Mumbai University

(Solution Set)

B.Sc.IT: Semester – VI

Practical Questions Paper

[CBSGS – 75:25 Pattern]

Data Warehouse

INDEX

Installing Oracle database software

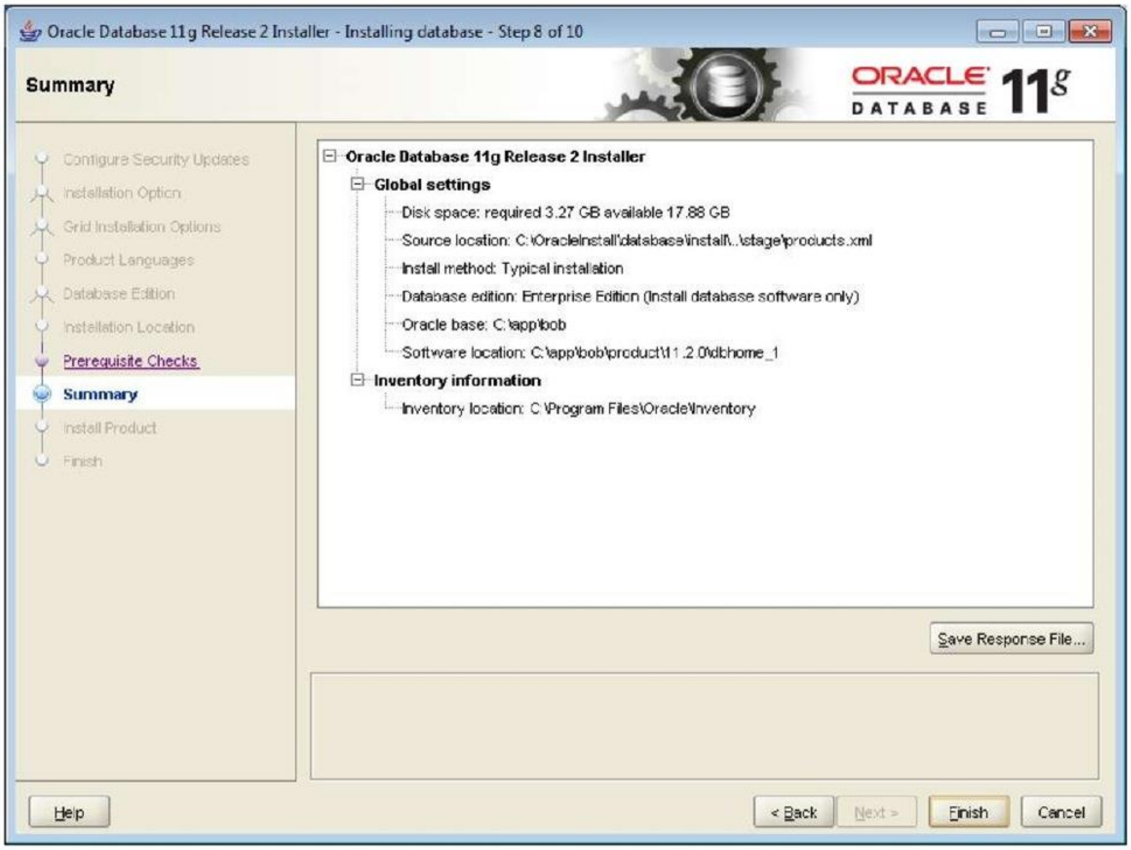
So far we've decided what system we're going to host the database on, downloaded the appropriate install file for that system, and unzipped the install files into a folder to begin the installation. We'll navigate to that folder and run the setup.exe file located there. This will launch the Oracle Universal Installer program to begin the installation. Those of us with experience installing the Oracle Database from prior versions will immediately notice the installer for 11gR2 has a slightly different look. It is more like a setup program for the database than the Universal Installer we're used to from previous versions, including the first edition of this topic.

We are installing the full database, which now automatically includes the Warehouse Builder client and database components. If we had an older version of the database (10g R2 for example) that did not include the Warehouse Builder software, or if we wanted to run the client on a different workstation than where the database software is installed, then there is the option to install the Warehouse Builder by itself.
A separately downloadable install for the standalone option is available at **http://www.oracle.com/technology/software/products/ warehouse/index.html.**
Skip ahead to the section titled installing the OWB standalone software if just the Warehouse Builder software is needed.
1. The first thing the installer is going to ask us is our email address for use in being notified of critical system updates that are available. This is something new Oracle has started doing with their installs to get people thinking about critical vulnerabilities and keeping their databases properly patched. It's similar to Microsoft's Windows Update feature that keeps users notified of available patches for the Windows operating system. That's a good feature but we would need a support agreement with Oracle and a My Oracle Support login to really make use of it so we're going to skip this and move on to step 2. Be aware that it will pop up a warning dialog asking us if we really want to remain uninformed about security related issues. We'll answer yes and move along.
2. The second step asks us what installation option we'd like. We can choose from one of three, create and configure a database, install the software only, or upgrade an existing database. We'll choose the second option. The Create and Install a Database option will make some assumptions about the database that we don't want depending on what options are selected so it's easier if we just install the database separately after the software is installed.
3. For the third step, we'll choose Single Instance for the Installation Type. The other option is for installing a database as part of a RAC installation (Real Application Cluster) of clustered databases.
4. In step 4 we'll choose the database language, or languages, we want to install.
5. Step 5 is where we choose the edition of the database to install, Enterprise, Standard, Standard Edition One, or Personal Edition. We'll choose the Enterprise Edition so we have access to all of the advanced features we need for the Warehouse Builder.

6. Step 6 will ask us for path names for ORACLE_BASE and for the Software Location (or Oracle home location). They will have suggested paths filled in for us. It is a good idea to leave the path names as they are and only change the drive designation if we'd like to install to a different hard drive. The install program will suggest a drive for the installation, but we might have a different preference. Oracle recommends a convention for naming folders and files that they call the Optimal Flexible Architecture (OFA). This is described in topic B of the Oracle Database Installation Guide for Microsoft Windows, which can be found at the following URL: **http://download.oracle.com/docs/cd/E118 8 2_01/install.112/e10843/of a.htm#CBBEDHEB.** It is a good idea to follow their recommendations for standardization so that others who have to work with the database files will know where to find them, and to save us from problems with possible conflicts with other Oracle products we may have installed. If we keep the default folder locations intact and only change the drive letter, we will adhere to the standard.

7. Step 7 of the install is where it will conduct the prerequisite checks to ensure our system is capable of running the database. If everything succeeds we'll move right to step 8, bypassing step 7 results. We could hit the back button on Step to move back to see the results if we wanted. If anything failed, it would have displayed the results for us.

8. Step 8 is the summary screen. It will display the Global Settings as in the previous version but includes a new Inventory section which replaces the old Product Languages, Space Requirements, and New Installations sections.



Oracle Database 11g Release 2 Installer - Installing database - Step 8 of 10

Summary

- Oracle Database 11g Release 2 Installer
  - Global settings
    - Disk space: required 3.27 GB available 17.88 GB
    - Source location: C:\OracleInstall\database\install\..\stage\products.xml
    - Install method: Typical installation
    - Database edition: Enterprise Edition (Install database software only)
    - Oracle base: C:\app\bob
    - Software location: C:\app\bob\product\11.2.0\dbhome_1
  - Inventory information
    - Inventory location: C:\Program Files\Oracle\Inventory

9. The actual installation happens in step 9. A progress bar proceeds to the right as the installation happens and steps for Prepare, Copy Files, and Setup Files are checked off as they are done.
10. Step 10 is the conclusion and finishes up with a success message:
Your database configuration files have been installed in C:\app\ while other components selected for installation have been installed in C:\app\product\11.2.0\dbhome_1. Be cautious not to accidentally delete these configuration files.

1. We will click Close to end the installation.

Basic versus advance install

The installation method we're following here is the quickest and easiest, but makes many decisions for us that more advanced options will ask us about like creating a database and Desktop vs Server installs. For the purpose of working through the examples in this topic, we will be OK with the basic installation. But if we were installing for a production environment, we would want to read through the Oracle Database Installation Guide (http://www.oracle. com/technology/documentation/database.html; click on View Library to view the documentation online or click on Download to download the documentation) to familiarize ourselves with the various situations that would require us to use the more advanced installation options. This would ensure that we don't end up with a database installation that will not support our needs.

Location of install results

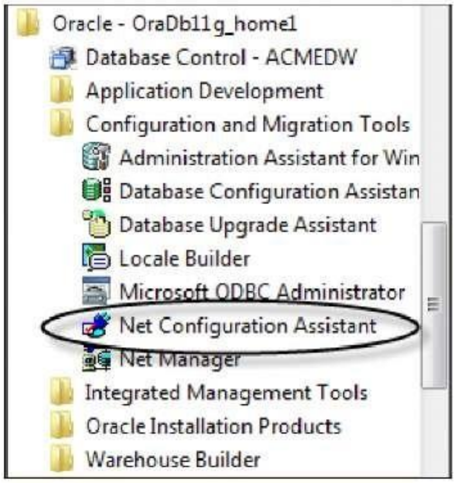
A good idea is to pay particular attention to the inventory location on the Step 8 summary screen, which tells us where we can find a log of the installation. The logs that the installer keeps are stored in the Oracle folder on the system drive in the following subfolder: C:\ Program Files\Oracle\Inventory\logs. The files are named with the following convention: install ActionsYYYY-MM-DD_HH-MI-SSPM where YYYY is the year, MM the month, DD the day, HH the hour, MI the minutes, SS the seconds of the time the installation was performed, and PM is either AM or PM. The files will have a .log extension. This information may come in useful later to see just what products were installed. The folder also will contain any errors encountered during the installation in files with a file extension of .err and any output generated by the installer in files with a file extension of .out.

Now that the software is installed, it's time to proceed with creating a database. But there is one step we have to do first—we need to configure the listener.

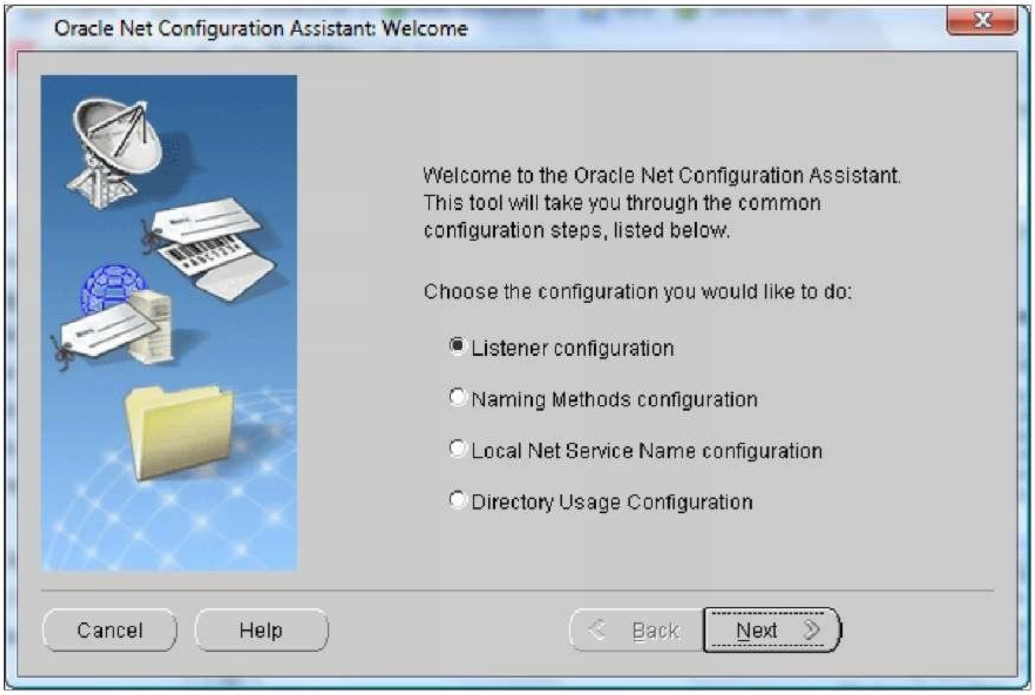
Configuring the listener

The listener is the utility that runs constantly in the background on the database server, listening for client connection requests to the database and handling them. It can be installed either before or after the creation of a database, but there is one option during the database creation that requires the listener to be configured—so we'll configure it now, before we create the database.

Run Net Configuration Assistant to configure the listener. It is available under the Oracle menu on the Windows Start menu as shown in the following image:


Oracle - OraDb11g_home1
Database Control - ACMEDW
Application Development
Configuration and Migration Tools
Administration Assistant for Win
Database Configuration Assistan
Database Upgrade Assistant
Locale Builder
Microsoft ODBC Administrator
Net Configuration Assistant
Net Manager
Integrated Management Tools
Oracle Installation Products
Warehouse Builder

The welcome screen will offer us four tasks that we can perform with this assistant. We'll select the first one to configure the listener, as shown here:


Oracle Net Configuration Assistant: Welcome

Welcome to the Oracle Net Configuration Assistant. This tool will take you through the common configuration steps, listed below.

Choose the configuration you would like to do:

● Listener configuration
○ Naming Methods configuration
○ Local Net Service Name configuration
○ Directory Usage Configuration

The next screen will ask you what we want to do with the listener. The four options are as follows:

- Add
- Reconfigure
- Delete
- Rename

Only the Add option will be available since we are installing Oracle for the first time. The remainder of the options will be grayed out and will be unavailable for selection. If they are not, then there is a listener already configured and we can proceed to the next section—creating the database. For those of us installing for the first time on our machines, we need to proceed with the configuration. The next screen will ask us what we want to name the listener. It will have LISTENER entered by default and that’s a fine name, which states exactly what it is, so let’s leave it at that and proceed.

The next screen is the protocol selection screen. It will have TCP already selected for us, which is what most installations will require. This is the standard communications protocol in use on the Internet and in most local networks. Leave that selected and proceed to the next screen to select the port number to use. The default port number is 1521, which is standard for communicating with Oracle databases and is the one most familiar to anyone who has ever worked with an Oracle database. So, change it only if you want to annoy the Oracle people in your organization who have all memorized the default Oracle port of 1521.

To change or not change the default listener port

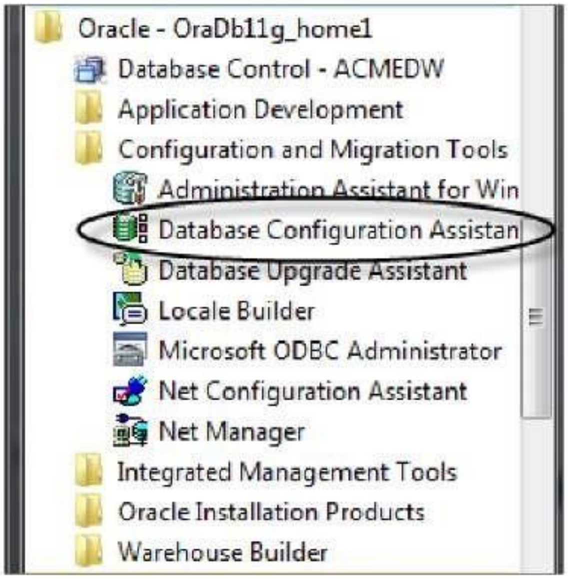
Putting aside the annoyance, the Oracle people might have to suffer as there are valid security reasons why we might want to change that port number. Since it is so common, the people accustomed to working with the Oracle database aren’t the only people who know that port number. Hackers looking to break into an Oracle database are going to go straight for that port number, so if we change it to something obscure, the database will be harder to find on the network for the people with malicious intent. If it does get changed, be sure to make a note of the assigned number.
There also may be firewall issues that allow only certain port numbers to be open through the firewall, which means communication on any of the other port numbers would be blocked. 1521 might be allowed by default since it is common for the Oracle database. It would be a good idea to check with the network support personnel to get their recommendation.

That is the last step. It will ask us if we want to configure another listener. Since we only need one, we’ll answer "no" and finish out the screens by clicking on the Finish button back on the main screen.

Creating the database

So far we have the Oracle software installed and a listener configured, but we have not created a database.

We will install a new database using Database Configuration Assistant, which Oracle provides to walk us step-by-step through the process of creating a database. It is launched from the Windows Start menu as shown in the following image:


Oracle - OraDb11g_home1
Database Control - ACMEDW
Application Development
Configuration and Migration Tools
Administration Assistant for Win
Database Configuration Assistan
Database Upgrade Assistant
Locale Builder
Microsoft ODBC Administrator
Net Configuration Assistant
Net Manager
Integrated Management Tools
Oracle Installation Products
Warehouse Builder

Running this application may require patience as we have to wait for the application to load after it's selected. Depending on the system it is running on, it can take over a minute to display, during which time there is no indication that anything is happening. It may be tempting to just select it again from the Start menu because it appears it didn't work the first time, but don't as that will just end up running two instances of the program. It will appear soon. The following are steps in the creation process:

1. The first step is to specify what action to take. Since we do not have a database created, we'll select the Create a Database option in step 1. If there was a database already created, the options for configuring a database or deleting a database would be selectable. Templates can be managed with the Database Configuration Assistant application, which are files containing preset options for various database configurations. Pre-supplied templates are provided with the application, and the application has the ability to custom-build templates.

In previous versions of the database, Automatic Storage Management could be configured as well however as of 11gR2, ASM has its own configuration assistant now, ASMCA. It is Oracle's feature for databases for automatically managing the layout and storage of database files on the system. These are both topics for a more advance topic on the Oracle Database. We will be creating a database using an existing template.

2. This step will offer the following three options for a database template to select:
° General Purpose or Transaction Processing ° Custom Database ° Data Warehouse
3. We are going to choose the Data Warehouse option for our purposes. If we already had a database installed that we wanted to use for learning OWB, but that's not configured as a data warehouse, it's not a problem. We can still run OWB hosted on it and create the data warehouse schema (database user and tables), which we'll be creating as we proceed through the topic. This would be fine for learning purposes, but for production-ready data warehouses a database configured specifically as a data warehouse should be used.
4. This step of the database creation will ask for a database name. The name of the database must be one to eight characters in length. Any more than that will generate an error when trying to proceed to the next screen. This is an Oracle database limitation. The database name can also include the network domain name of the domain of the host it is running on, to further uniquely identify it. Follow the name with a period and then the domain, which itself can include additional periods.
If this database is being created for business use, a good naming scheme would reflect the purpose of the database. Since we're creating this database for the data warehouse of ACME Toys and Gizmos Company, we'll choose a name that reflects this—ACME for the company name and DW for data warehouse, resulting in a database name of ACMEDW. It is important to remember this name as it will be a part of any future connections to the database.

As the database name is typed in, the SID (or Oracle System Identifier) is automatically filled in to match it. If the domain is added to the database name, the SID will stop pre-populating after the first period is entered. The end result is that the SID becomes the same as the first part of the database name.

5. This step of the database creation process asks whether we want to configure Enterprise Manager. The box is checked by default and left as it is. This is a web-based utility Oracle provides for controlling a database, and as it is very useful to have, we will want to enable it. There are two options for controlling a database: registering with Grid Control or local management. Grid Control is Oracle's centralized feature for controlling a grid, a network of loosely coupled modular hardware and software components that can be joined and rejoined together on demand to meet business needs. That is what the "g" in Oracle Database 11g stands for. If your network is not configured in a grid architecture, or you are installing on a standalone machine, then choose the local management option. It will automatically detect a Grid Control agent that is running locally, and if it doesn't find one, the Grid Control option will be grayed out anyway. In that case, you will only be able to select local management.
6. New in the 11gR2 version of the DBCA is the additional tab on this screen for the Automatic Maintenance option. This step used to be all by itself as step 12 of the install process. We'll deselect that option and move on, since we don't need that additional functionality. Automatic Maintenance

Tasks are tasks that run in predefined maintenance windows of time to perform various preconfigured maintenance operations on the database. Since the database for this topic is only for learning purposes, it is not critical that these maintenance tasks be done automatically.

7. Automatic maintenance is designed to run during preset maintenance windows, which are usually in the middle of the night. So if the database system is shut down every day, there wouldn't be a good window to run the tasks on regularly anyway. If installing in a production environment with servers that will be running 24 hours a day every day, then consider setting up the automatic maintenance to occur. Oracle provides three pre-configured maintenance tasks to choose from—collecting statistics for the query optimizer (for improving performance of SQL queries), Automatic Segment Advisor for analyzing storage space for areas that can possibly be reclaimed for use, and the Automatic SQL Tuning Advisor for automatically analyzing SQL statements for performance improvements.

If an error is encountered at some point during the database creation that indicates a listener is not configured, it simply means we started the DBCA before configuring a listener. To solve that, there is no need to exit out of the database install window, just go back and perform the listener install steps and come back here where the screen will allow us to proceed.

8. On this screen (step 5) we can set the database passwords on the system accounts using a different one for each account, or by choosing one password for all four. We're going to set a single password on all four, but for added security in a production environment, it is a good idea to make a different password for each. Click on the option to Use the Same Administrative Password for All Accounts and enter a password. This is very important to remember as these are key system accounts used for database administrative control. Oracle has decided as of this release to attempt to make us choose better, more secure passwords by popping up a warning dialog if the password we've chosen does not meet with their standards of complexity. It is only a warning however and we can answer Yes and continue.

9. This step is a combination now of the old steps 6 and 7 of the installation. It is about storage and specifying the locations where database files are to be created. We'll leave it at the default of File System for storage management. The other two options are for more advanced installations that have greater storage needs. The locations where database files are to be created can be left at the default for simplicity (which uses the locations specified in the template and follows the OFA standard for naming folders described above). A storage screen will come up where we'll be able to change the actual file locations if we want, for all but the Oracle-Managed Files option.

The Oracle-Managed Files option is provided by the database so that we can let Oracle automatically name and locate our data files. A folder location is specified on the step 6 screen, which will become the default location for any files created using this option. This is why we won't be able to change any file locations later on during the installation if this option is chosen. However, files can still be created with explicit names and locations after the database is running.

10. The next screen is for configuring recovery options. We're up to step 8 now. If we were installing a production database, we would want to make sure to use the Flash Recovery option and to enable Archiving. Flash Recovery is a feature Oracle has implemented in its database to provide a location that is managed by the database. It stores backups and files needed to recover a database in the event of disk failure. With Flash Recovery Area specified, we can recover data that would otherwise be lost in a system failure.
Enabling archiving turns on the archive log mode of the database, which causes it to archive the redo logs (files containing information that is used by the database to recover transactions in the event of a failure.) Having redo logs archived means you can recover your database up to the time of the failure, and not just up to the time of the last backup.

These recovery options will consume more disk space, but will provide a recovery option in the event of a failure. Each individual will have to make the call for their particular situation whether that is needed or not.
We'll specify Flash Recovery and for simplicity, we will just leave the default for size and location. We will not enable archiving at this point. These options can always be modified after the database is running, so this is not the last chance to set them.

11. This step is where we can have the installation program create some sample schemas in the database for our reference, and specify any custom scripts to run. The text on the screen can be read to decide whether they are needed or not. We don't need either of these for this topic, so it doesn't matter which option we choose.
12. The next screen is for Initialization Parameters. These are the settings that are put in place to define various options for the database such as Memory options. There are over 200 different parameters and to go through all of them would take much more time and space than we have here. There is no need for that at this point as there are about 28 parameters that Oracle says are basic parameters that every database installation should set. We're just going to leave the defaults set on this screen, which will set the basic parameters for us based on the amount of memory and disk space detected on our machine. We'll just move on from here. Once again, these can all be adjusted later after the database is created and running if we need to make changes.
13. The next step (step 10 of 11) is the Database Storage screen referred to earlier. Here the locations of the pre-built data and control files can be changed if needed. They should be left set to the default for simplicity since this won't be a production database. For a production environment, we would want to consider storing data files on separate partitions for performance reasons, and to minimize the impact of disk failures on the running database if something goes wrong. If all the data files are on one drive and it goes bad, then the whole database is down.

14. The final step has the following three options, and any or all can be selected for creating the database:

° Create the database directly

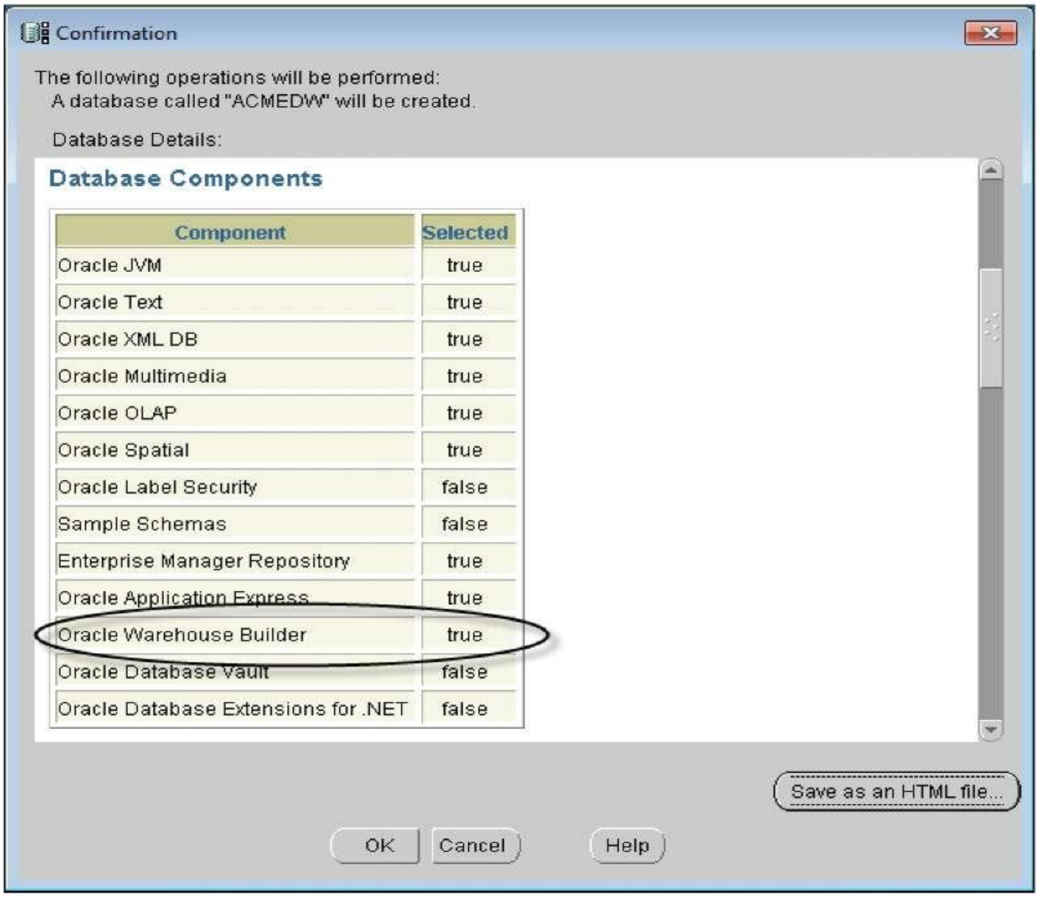
° Save the creation options as a template for later use

° Save database creation scripts that can be used later to create the database

We'll leave the first checkbox checked to go ahead and create the database.

The Next button is grayed out since this is the last screen. So click on the Finish button to begin creating the database using the selections we've just chosen. It will display a summary screen showing what options it will be using to install with. We can save this as an HTML file if we'd like to keep a record of it for future reference.

All that information will be available in the database by querying system tables later, but it's nice to have it all summarized in one file. We can scroll down that window and verify the various options that will be installed, including Oracle Warehouse Builder, which will have a true in the selected column as shown here:



The following operations will be performed:
A database called "ACMEDW" will be created.

Database Details:

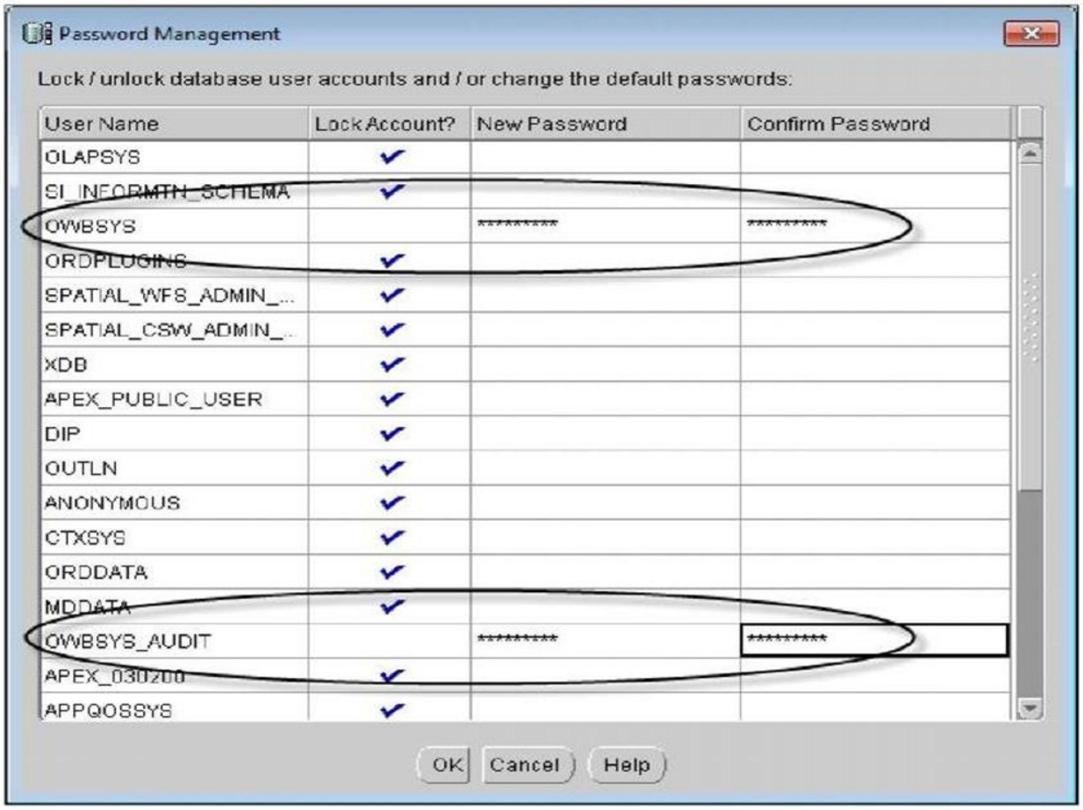
Database Components

Component	Selected
Oracle JVM	true
Oracle Text	true
Oracle XML DB	true
Oracle Multimedia	true
Oracle OLAP	true
Oracle Spatial	true
Oracle Label Security	false
Sample Schemas	false
Enterprise Manager Repository	true
Oracle Application Express	true
Oracle Warehouse Builder	true
Oracle Database Vault	false
Oracle Database Extensions for .NET	false

We will be presented with the progress screen next that will show us the progress as it creates the database.

When the install progress screen gets to 100% and all the items are checked off, we will be presented with a screen summarizing the database configuration details. Take a screen capture of this screen or write down the details because it's good to know information on how the database is configured. Especially, we'll need the database name in later installation steps. We may see the progress screen at 100% doing nothing with apparently no other display visible. Just look around the desktop underneath other windows for the Database Configuration Screen. It's important for the next step.

On the final Database Configuration Screen, there is a button in the lower right corner labeled Password Management. We need to click on this button to unlock the schema created for OWB use. Oracle configures its databases with most of the pre-installed schemas locked, and so users cannot access them. It is necessary to unlock them specifically, and assign our own passwords to them if we need to use them. Two of them are the OWBSYS schema and the OWBSYS_AUDIT schema. These are the schemas that the installation program automatically installs to support the Warehouse Builder. They are required for running the Warehouse Builder. Click on the Password Management button and on the resulting Password Management screen, we'll scroll down until we see the OWBSYS and OWBSYS_AUDIT schemas. We'll click on the check box to uncheck it on each one (indicating we want them unlocked) and then type in a password and confirm it as shown in the following image:



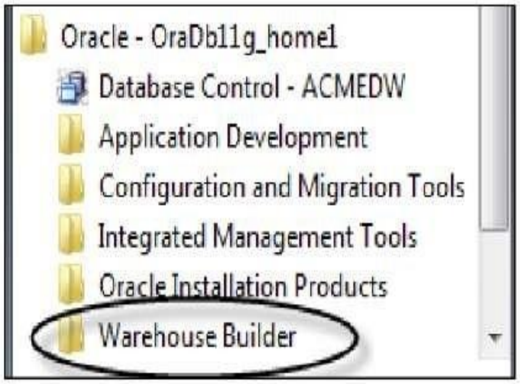
Lock / unlock database user accounts and / or change the default passwords:

User Name	Lock Account?	New Password	Confirm Password
OLAPSYS	✓		
SI_INFORMTN_SCHEMA	✓		
OWBSYS		*********	*********
ORDPLUGINS	✓		
SPATIAL_WFS_ADMIN_...	✓		
SPATIAL_CSW_ADMIN_...	✓		
XDB	✓		
APEX_PUBLIC_USER	✓		
DIP	✓		
OUTLN	✓		
ANONYMOUS	✓		
CTXSYS	✓		
ORDDATA	✓		
MDDATA	✓		
OWBSYS_AUDIT		*********	*********
APEX_030200	✓		
APPQOSSYS	✓		

Click on the OK button to apply these changes and close the window. On the Database Configuration Screen, click on the Exit button to exit out of the Database Configuration Assistant. That's it. We're done installing our first database and it's ready to use. Next, we'll discuss installing the OWB client if we want to run the client on another computer, or if we already have a 10gR2 database installed that we want to use with the Warehouse Builder.

Installing the OWB standalone software

If we are going to run the OWB client on the same computer as we just installed the Oracle database on, we don't need any more installations. That is the configuration used in this topic. The OWB client software is now installed by default with the main database installation. We can verify that by checking the Start menu entry for Oracle. We will see a submenu entry for Warehouse Builder as shown in the following image:


Oracle - OraDb11g_home1
Database Control - ACMEDW
Application Development
Configuration and Migration Tools
Integrated Management Tools
Oracle Installation Products
Warehouse Builder

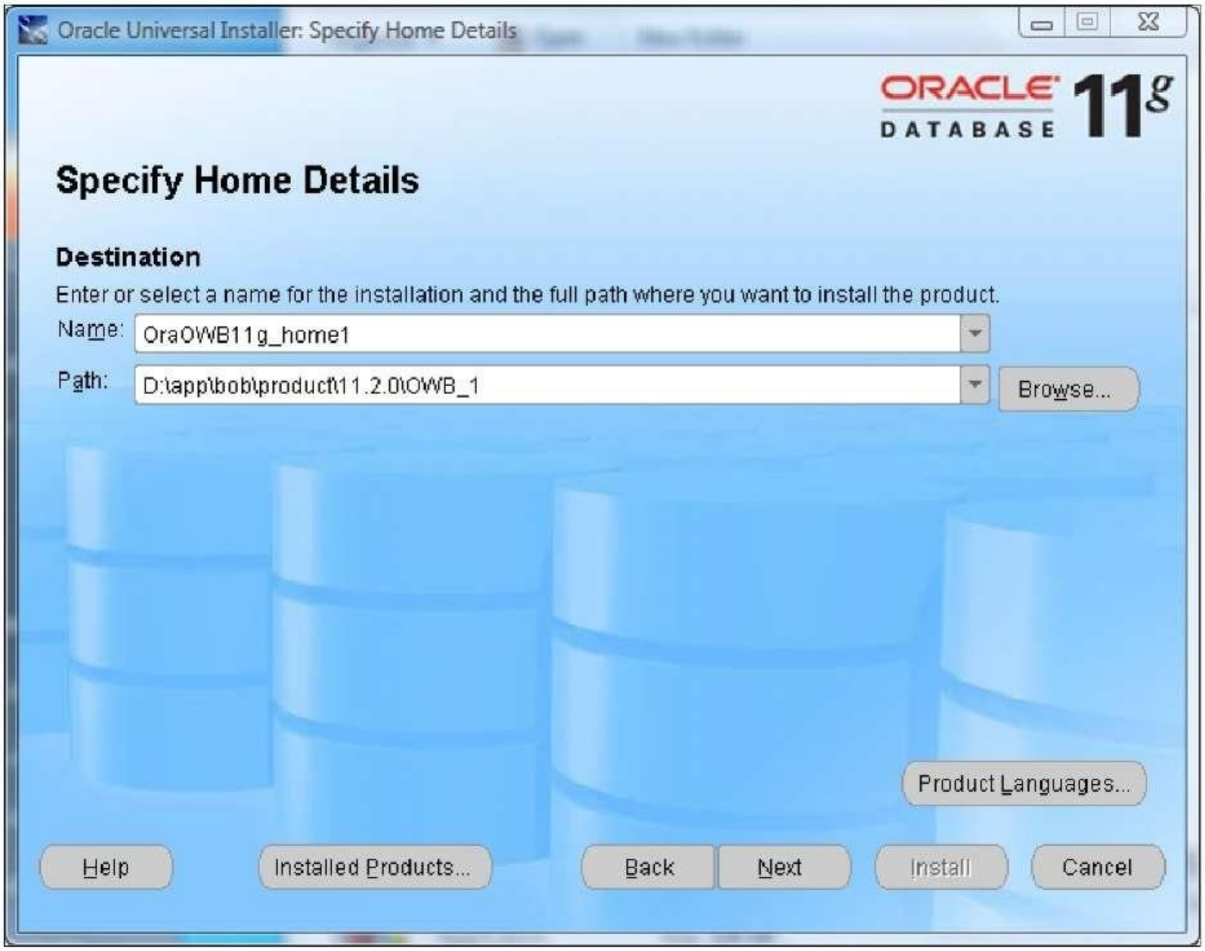
If we want to run the OWB client on another computer on the network, or if we have an older version of the database already installed (10g Rel 2) and want to be able to use the Warehouse Builder software with it, we'll need to continue here with the installation of the OWB client software. For all others, we can proceed to the next section on OWB — OWB components and architecture.

For the task of installing the standalone client, we'll need to download the OWB client install file. So we will go back to the Oracle site on the Internet. The download page is at the following URL at the time of writing: http://www.oracle.com/ technology/software/products/warehouse/index.html. If that is not working, go to the main Oracle site and search for the Business Intelligence | Data Warehousing page where there is a link for the download of the OWB client.

Once again we'll have to accept the license agreement before the download links will become active. So we'll accept it and download the install file to the client computer on which we'll be installing the software. The Windows ZIP file is about 930MB in size so we need to make sure we have enough room on our hard drive to store the file. We'll need at least double that amount of space because the install files will take up that much space when unzipped.

When we have downloaded the ZIP file and unzipped it to our hard drive, run setup.exe in the top-level folder to run the Oracle Universal Installer. It should look familiar. Oracle is definitely correct in calling their installer "Universal". Every Oracle database product uses that installer, so we will become very familiar with it if we have to install any more Oracle products. It is universal also in the fact that it runs on every platform that Oracle supports, and so the same interface is used no matter where we install it. The installation steps are as follows:

1. The first step it goes through is asking us for the Oracle home details. It's similar to what it asked at the beginning of the database installation as shown in the following image:



Oracle Universal Installer: Specify Home Details

Specify Home Details

Destination

Enter or select a name for the installation and the full path where you want to install the product.

Name: OraOWB11g_home1

Path: D:\app\bob\product\11.2.0\OWB_1

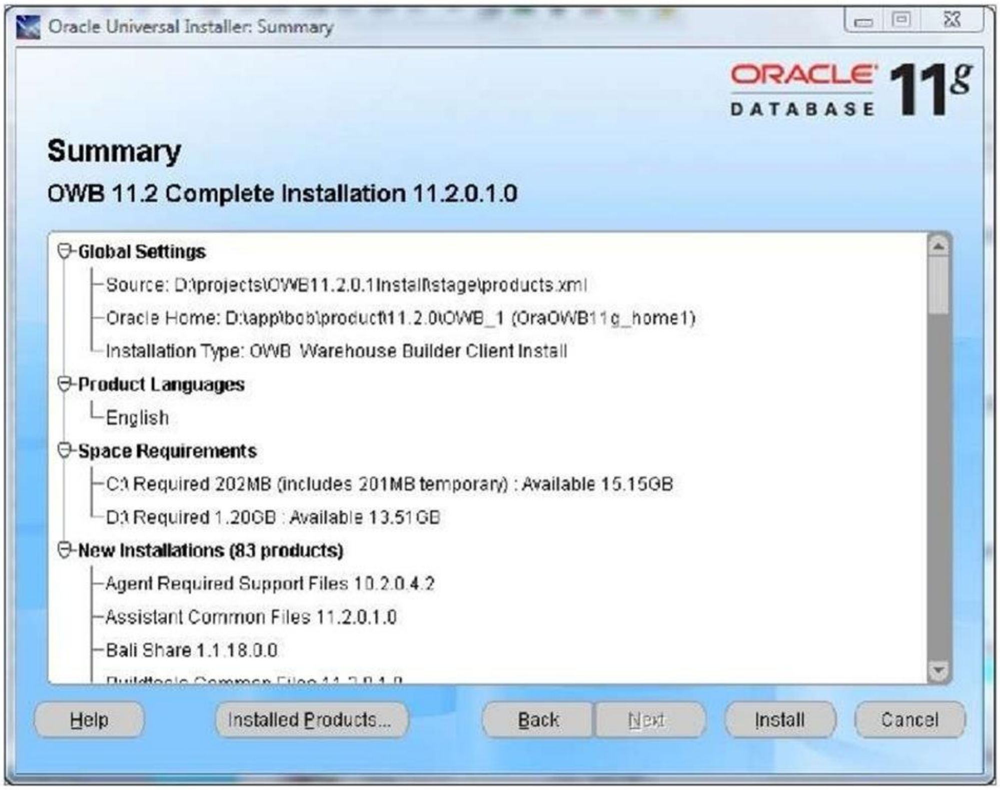
The installer will again suggest OraDb11g_home1 or something similar, but we'll change it to OraOWB11g_home1 since it's just the OWB installation and not the full database.

When installing the standalone OWB software, remember that it cannot be installed into the same ORACLE home as the database. It must reside in its own Oracle home folder. So if we have a database that's already installed on the same machine, we'll have to make sure the ORACLE_ HOME we specify is different. The installer will warn us if we try to specify the same one and won't let us continue until it is different.

We need to verify the installation location for the home location also. The suggested name that it provides conforms to the OFA standard just as the database installation did, so we'll want to just change the drive letter if needed. However, the bottom-most folder name can be changed if needed without violating the OFA standard. If it has a default of db_1, we can change it to OWB_1 just to be clear that it's the OWB client.

2. The second step is that email prompt again for being notified about security issues that we saw earlier in the database software install. We can just continue and answer yes to the popup warning.

3. The next and final step is the summary screen. The OWB client installation is not as complex as a full database installation, so it does not need all the additional information it asked for during the database installation. The summary screen should look similar to the following:



Oracle Universal Installer: Summary

ORACLE DATABASE 11g

Summary

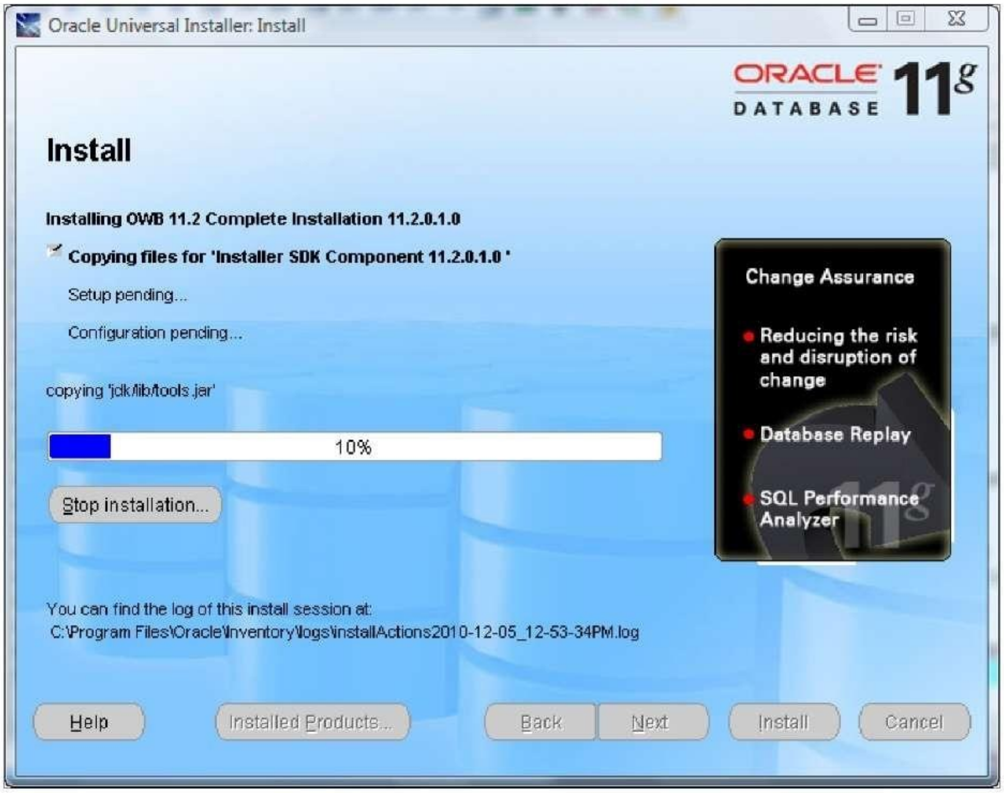
OWB 11.2 Complete Installation 11.2.0.1.0

- Global Settings
  - Source: D:\projects\OWB11.2.0.1Install\stage\products.xml
  - Oracle Home: D:\app\bob\product\11.2.0\OWB_1 (OraOWB11g_home1)
  - Installation Type: OWB Warehouse Builder Client Install
- Product Languages
  - English
- Space Requirements
  - C:\ Required 202MB (includes 201MB temporary) : Available 15.15GB
  - D:\ Required 1.20GB : Available 13.51GB
- New Installations (83 products)
  - Agent Required Support Files 10.2.0.4.2
  - Assistant Common Files 11.2.0.1.0
  - Bali Share 1.1.18.0.0
  - Buildtools Common Files 11.2.0.1.0

This summary gives us an idea of the disk space it will need, as well as the products that will be installed. If we scroll down the list, we'll see a number of other Oracle utilities and applications that it will install. We will also see items that are installed on the server as a part of the database install, but that will now be available to us on our client workstation. SQL*Plus appears there, which is the command line utility for accessing an Oracle database directly using SQL (Structured Query

Language, the language used for accessing information stored in databases) among a host of other features.

Upon proceeding, the next screen will begin the installation and present us with the progress screen with a sliding bar moving to the right to indicate how far it has progressed. This is similar to what it did for the full database installation. An example of that screen is included next for reference:



Oracle Universal Installer: Install

Install

Installing OWB 11.2 Complete Installation 11.2.0.1.0

Copying files for 'Installer SDK Component 11.2.0.1.0'

Setup pending...

Configuration pending...

copying 'jdk/lib/tools.jar'

10%

You can find the log of this install session at:
C:\Program Files\Oracle\Inventory\logs\installActions2010-12-05_12-53-34PM.log

Install results

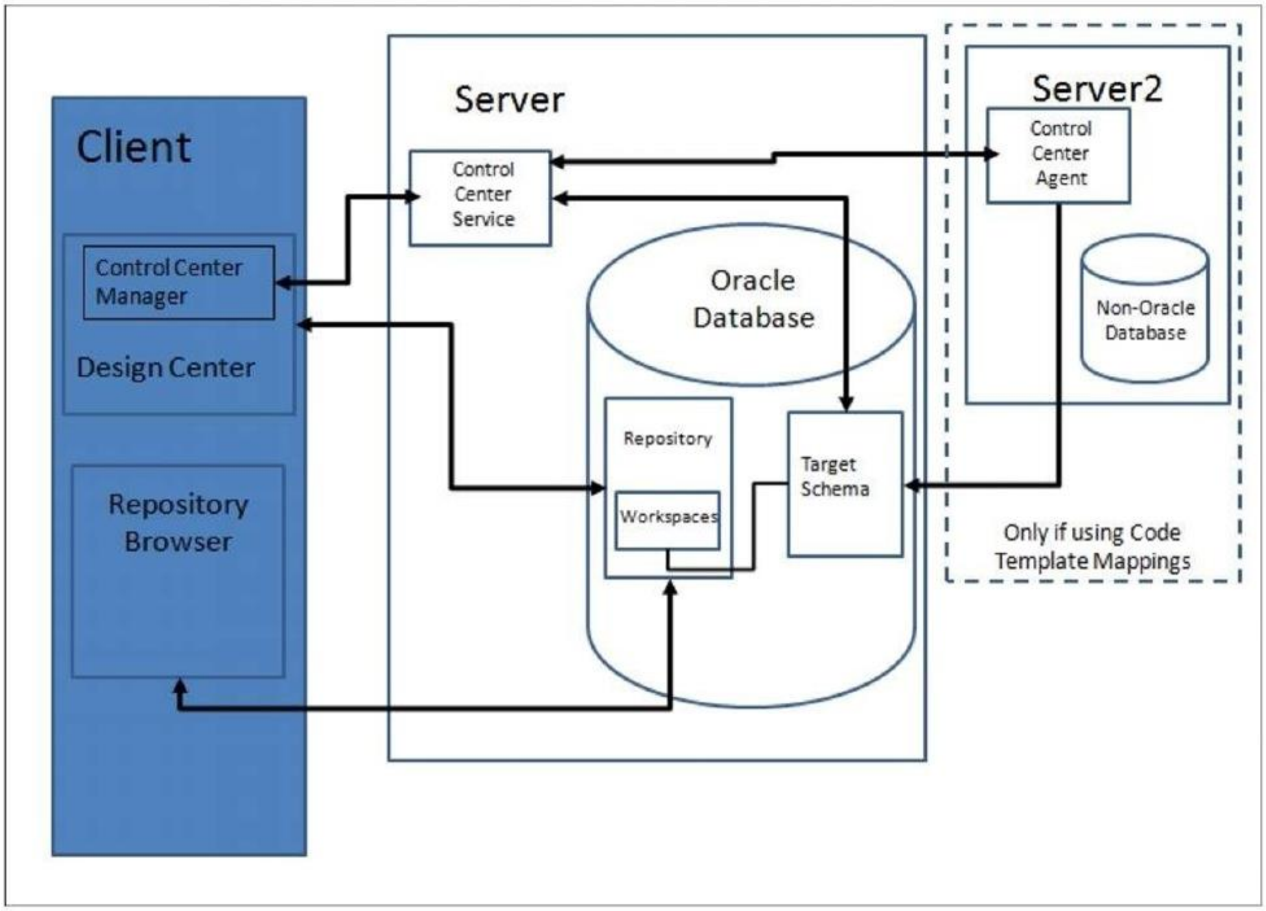
The log files with the results of the installation are stored in the same location as they are for a full database install. The universal installer will use that same folder for all its installs.

When the installation is complete, we will be presented with the final success screen and an Exit button. And as if to remind us about the universal nature of the installer, it will pop up a confirmation box asking if we really want to exit, even though for this installation there is nothing else that would be available to do on that final screen if we said no.

OWB components and architecture

Now that we've installed the database and OWB client, let's talk about the various components that have just been installed that are a part of the OWB and the architecture of OWB in the database. Then we'll perform one final task that is required before using OWB to create our data warehouse.

Oracle Warehouse Builder is composed on the client of the Design Center (including the Control Center Manager) and the Repository Browser. The server components are the Control Center Service, the Repository (including Workspaces), and the Target Schema. New for release 11gR2 is the Control Center Agent which is used by the new Code Template Mappings to communicate with a non-Oracle database. We'll be covering that in more detail in topic 10. A diagram illustrating the various components and their interactions follows:



Client and server

The previous diagram depicts a client and server, but these are really just logical notions to indicate the purpose of the individual components and are not necessarily physically separate machines. The client components are installed with the database as we've seen previously and,

therefore, can run on the same machine as the database. This configuration is assumed throughout the topic.
The Design Center is the main client graphical interface for designing our data warehouse. This is where we will spend a good deal of time to define our sources and targets, and describe the extract, transform, and load (ETL) processes we use to load the target from the sources. The ETL procedures are what we will define to carry out the extraction of the data from our sources, any transformations needed on it and subsequent loading into the data warehouse. What we will create in the Design Center is a logical design only, not a physical implementation. This logical design will be stored behind the scenes in a Workspace in the Repository on the server. The user interacts with the Design Center, which stores all its work in a Repository Workspace.

We will use the Control Center Manager for managing the creation of that physical implementation by deploying the designs we've created into the Target Schema. The process of deployment is OWB's method for creating physical objects from the logical definitions created using the Design Center. We'll then use the Control Center Manager to execute the design by running the code associated with the ETL that we've designed. The Control Center Manager interacts behind the scenes with the Control Center Service, which runs on the server as shown in the previous image. The user directly interacts with the Control Center Manager and the Design Center only.
The Target Schema is where OWB will deploy the objects to, and where the execution of the ETL processes that load our data warehouse will take place. It is the actual data warehouse schema that gets built. It contains the objects that were designed in the Design Center, as well as the ETL code to load those objects. The Target Schema is not an actual Warehouse Builder software component, but is a part of the Oracle Database. However, it will contain Warehouse Builder components such as synonyms that will allow the ETL mappings to access objects in the Repository.
The Repository is the schema that hosts the design metadata definitions we create for our sources, targets, and ETL processes. Metadata is basically data about data. We will be defining sources, targets, and ETL processes using the Design Center and the information about what we have defined (the metadata) is stored in the Repository.

The Repository is a Warehouse Builder software component for which a separate schema is created when the database is installed—OWBSYS. This is the schema we talked about unlocking during the installation discussion previously as one of the final steps in the database creation process. This will be created automatically by the 11g install, but is installed separately using scripts if we want to host the Repository on an Oracle 10g database. The explanations in this topic all assume that the Repository is hosted on an Oracle 11g database. The Oracle Warehouse Builder Installation and Administration Guide found at the following URL:
http://download.oracle. com/docs/cd/E118 82_01/owb.112/e1713 0/toc.htm discusses the procedure for installing the Repository schema on an Oracle 10g release 2 database if needed. It also

contains more detailed information about the various other configurations that can be installed along with more detailed diagrams of the various options.
The Repository will contain one or more Workspaces as shown in the previous diagram. A Workspace is where we will do our work to create the data warehouse. There can be more than one workspace defined in the Repository. A common example of how multiple workspaces can be employed is to use different workspaces corresponding to sets of users working on related projects. We could have one workspace for development, one for testing, and one for production. The development team could be working in the development environment separately from the test team that would be working in the test environment. For our purposes at the ACME Toys and Gizmos Company, we will be working out of one workspace.

This concept of the workspace is new in the 11g releases of OWB. The Repository is created in the OWBSYS schema during the database installation. So setting up the Repository information and workspaces no longer requires SYSDBA privileges for the user to install the Repository. SYSDBA is an advanced administrative privilege that is assigned to a user in an Oracle database. This allows the user to perform tasks affecting the database and other database users that ordinary user accounts cannot do (or for that matter, other administrative accounts without SYSDBA). For security reasons, we want to restrict user accounts with SYSDBA privilege to a minimum. So it is good that we don’t have to use that privilege when we install the Repository.
One final OWB component to consider is the Repository Browser on the client. It is a web browser interface for retrieving information from the Repository. It will allow us to view the metadata, create reports, and audit runtime operations. It is the only other component besides the Design Center and the Control Center Manager that the user interacts with directly.

We will have a chance to visit each one of these areas in much more detail as we progress through the design and build of our data warehouse. However, first there is one more installation step we have to take before we can begin using the Warehouse Builder. The Repository must be configured for use and a workspace must be defined.

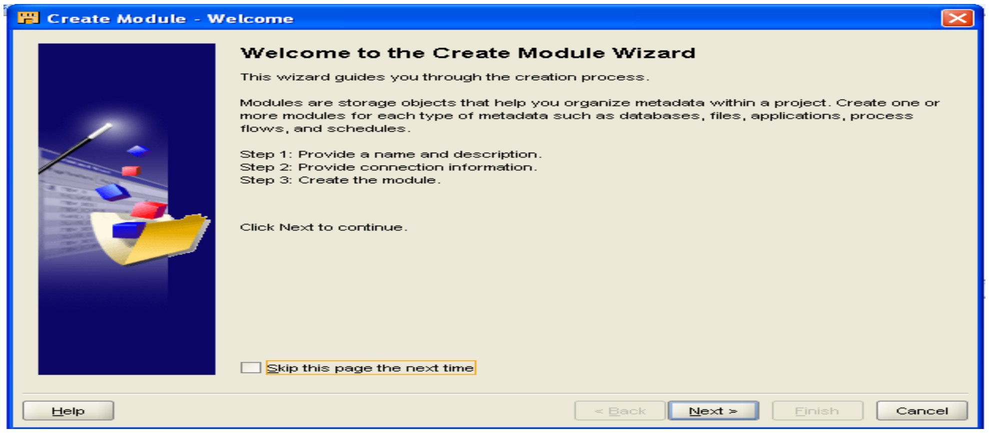
PRACTICAL NO:01

Aim: Importing source data structures in Oracle

1.Creating an Oracle Database module

To create an Oracle Database module: Databases |Oracle node |Project Explorer

1.Click on the **Next** button on **Welcome** screen



Create Module - Welcome

Welcome to the Create Module Wizard

This wizard guides you through the creation process.

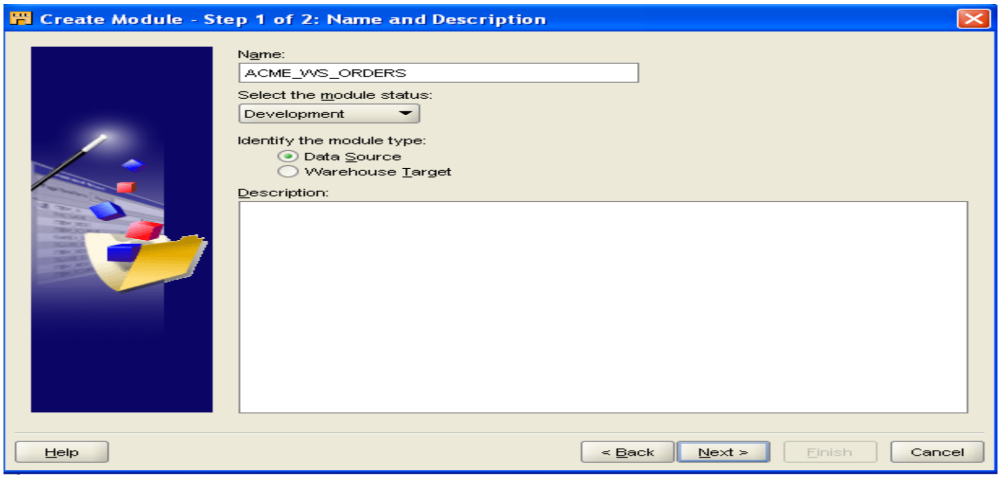
Modules are storage objects that help you organize metadata within a project. Create one or more modules for each type of metadata such as databases, files, applications, process flows, and schedules.

Step 1: Provide a name and description.
Step 2: Provide connection information.
Step 3: Create the module.

Click Next to continue.

□ Skip this page the next time

2. Name the module as **ACME_WS_ORDERS** and select the **Data source**



Create Module - Step 1 of 2: Name and Description

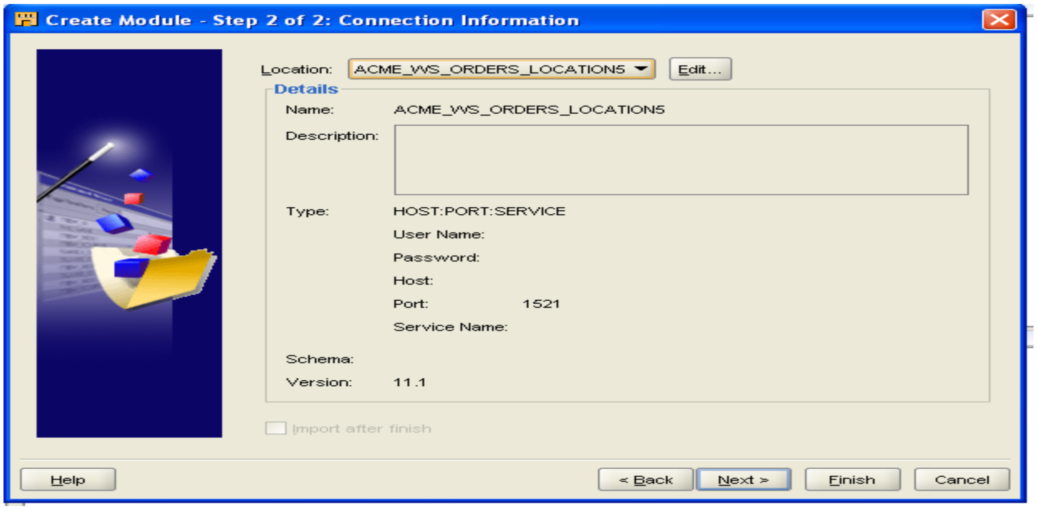
Name:
ACME_WS_ORDERS

Select the module status:
Development

Identify the module type:
● Data Source
○ Warehouse Target

Description:

3.Go to **Edit…**button change the location as **ACME_WS_ORDERS_LOCATION**



Create Module - Step 2 of 2: Connection Information

Location: ACME_WS_ORDERS_LOCATIONS

Details

Name: ACME_WS_ORDERS_LOCATIONS

Description:

Type: HOST:PORT:SERVICE

User Name:

Password:

Host:

Port: 1521

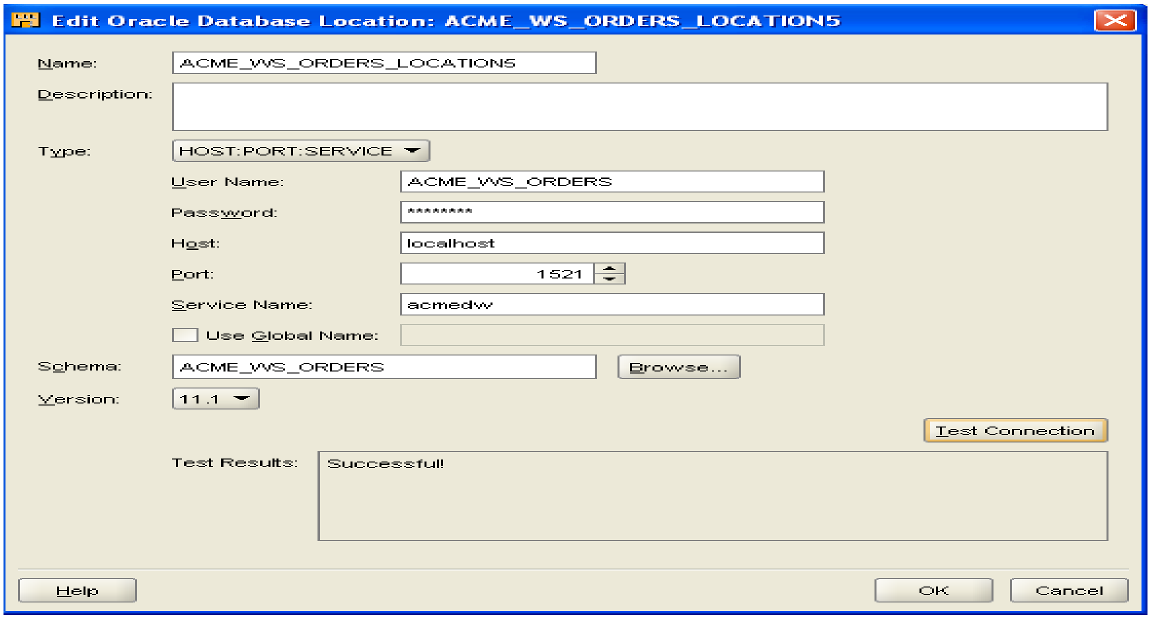
Service Name:

Schema:

Version: 11.1

□ Import after finish

4.fill the details



Edit Oracle Database Location: ACME_WS_ORDERS_LOCATION5

Name: ACME_WS_ORDERS_LOCATIONS

Description:

Type: HOST:PORT:SERVICE

User Name: ACME_WS_ORDERS

Password: ********

Host: localhost

Port: 1521

Service Name: acmedw

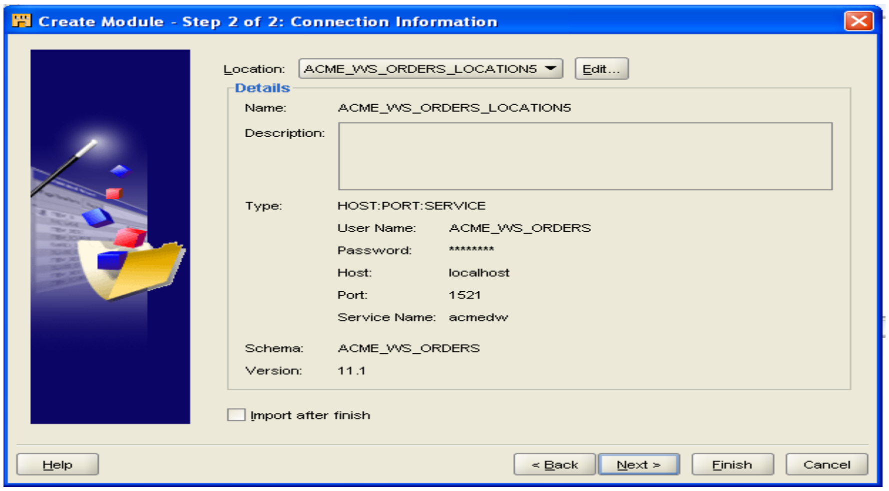
□ Use Global Name:

Schema: ACME_WS_ORDERS

Version: 11.1

Test Results: Successfull!

5. Uncheck **import after finish**



Create Module - Step 2 of 2: Connection Information

Location: ACME_WS_ORDERS_LOCATIONS

Details

Name: ACME_WS_ORDERS_LOCATIONS

Description:

Type: HOST:PORT:SERVICE

User Name: ACME_WS_ORDERS

Password: ********

Host: localhost

Port: 1521

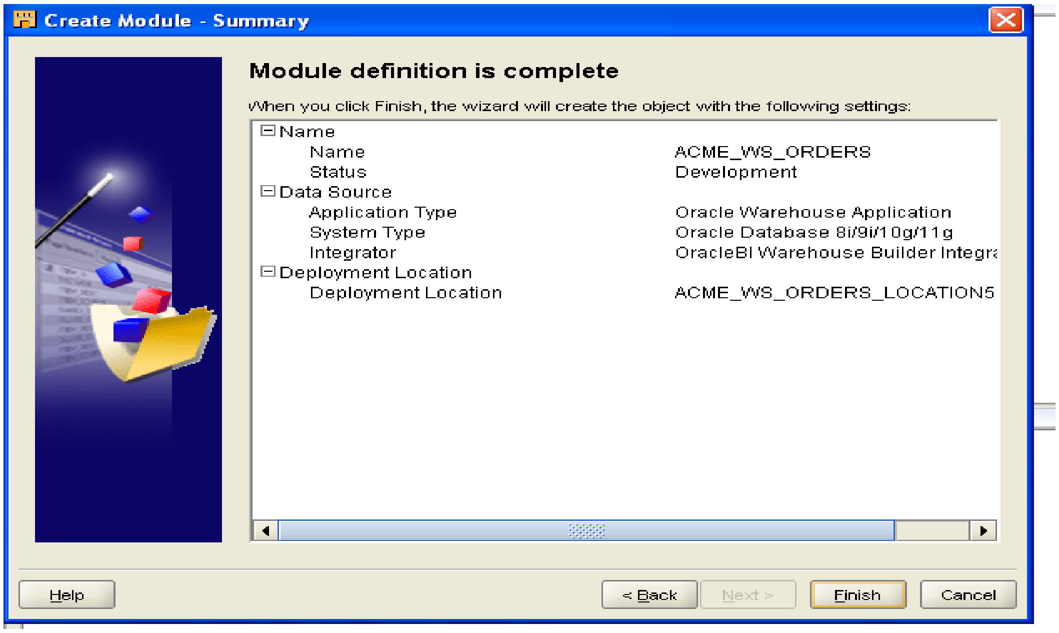
Service Name: acmedw

Schema: ACME_WS_ORDERS

Version: 11.1

□ Import after finish

6. click on **Finish**



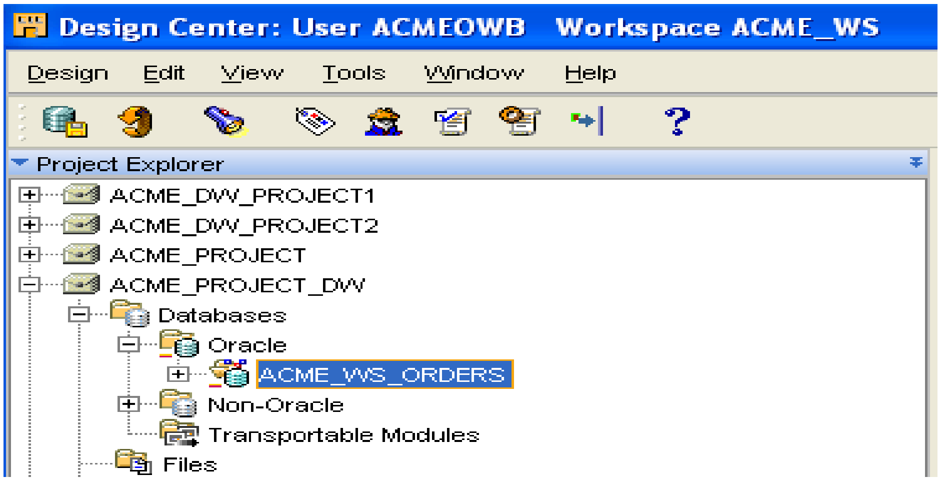
Create Module - Summary

Module definition is complete

When you click Finish, the wizard will create the object with the following settings:

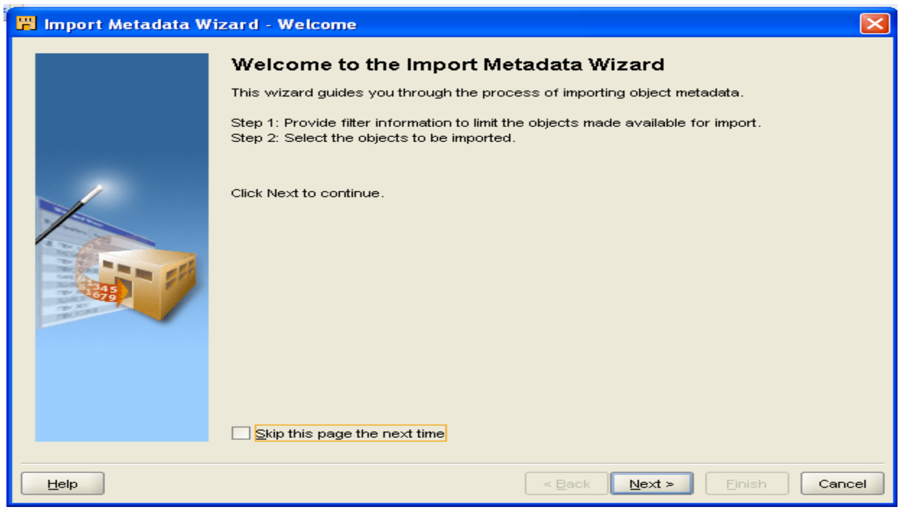
```
⊟ Name
     Name                              ACME_WS_ORDERS
     Status                            Development
⊟ Data Source
     Application Type                  Oracle Warehouse Application
     System Type                       Oracle Database 8i/9i/10g/11g
     Integrator                        OracleBI Warehouse Builder Integra
⊟ Deployment Location
     Deployment Location               ACME_WS_ORDERS_LOCATIONS
```

7.created **ACME_WS_ORDERS**



2.Importing source metadata from a database

1.**ACME_WS_ORDERS** |**Databases** |**Oracle** node |**Project Explorer** | **Import….** From pop-up menu Click on the next button on **Welcome** screen


Import Metadata Wizard - Welcome

Welcome to the Import Metadata Wizard

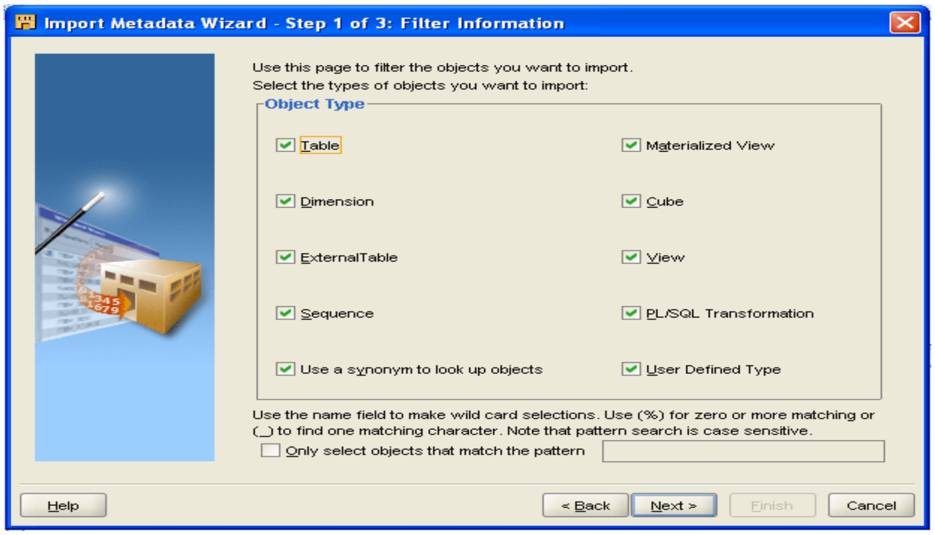
This wizard guides you through the process of importing object metadata.

Step 1: Provide filter information to limit the objects made available for import.
Step 2: Select the objects to be imported.

Click Next to continue.

□ Skip this page the next time
Help
< Back
Next >
Finish
Cancel

2. Presented with a screen labeled **Step1**



Import Metadata Wizard - Step 1 of 3: Filter Information

Use this page to filter the objects you want to import.
Select the types of objects you want to import:

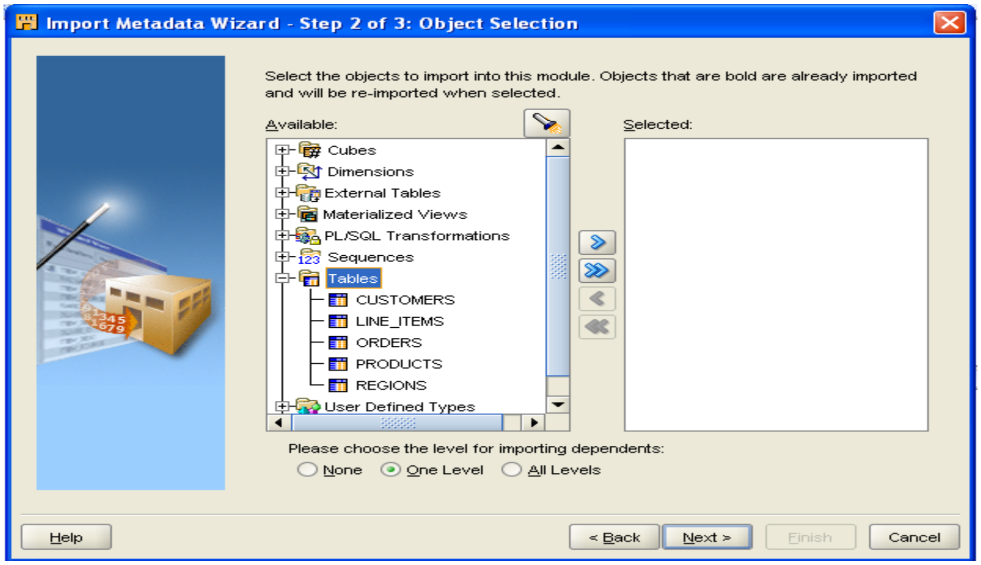
Object Type

☑ Table
☑ Materialized View
☑ Dimension
☑ Cube
☑ ExternalTable
☑ View
☑ Sequence
☑ PL/SQL Transformation
☑ Use a synonym to look up objects
☑ User Defined Type

Use the name field to make wild card selections. Use (%) for zero or more matching or (_) to find one matching character. Note that pattern search is case sensitive.

□ Only select objects that match the pattern

3. Click on next to move to **Step2 and** expand the table node



Import Metadata Wizard - Step 2 of 3: Object Selection

Select the objects to import into this module. Objects that are bold are already imported and will be re-imported when selected.

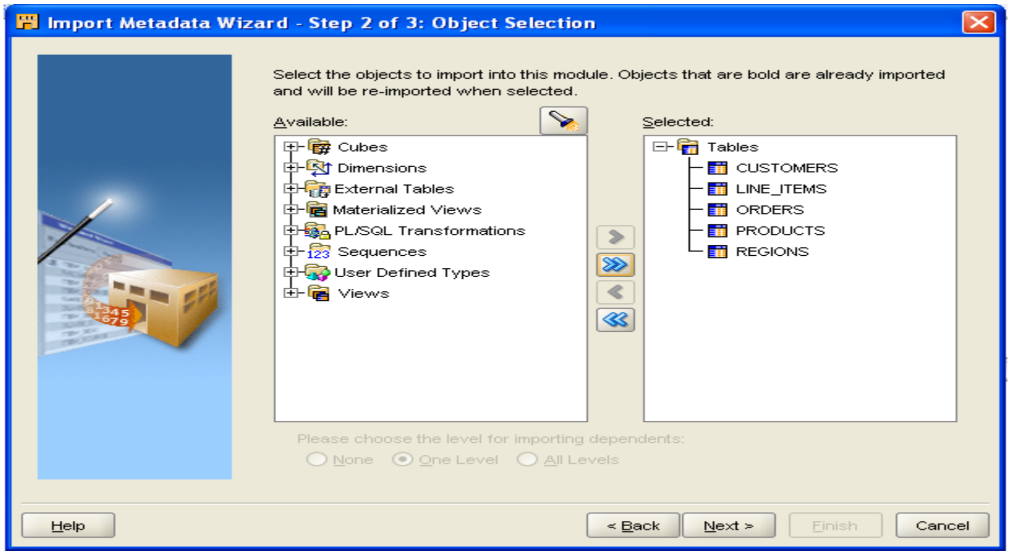
Available:

- Cubes
- Dimensions
- External Tables
- Materialized Views
- PL/SQL Transformations
- Sequences
- Tables
  - CUSTOMERS
  - LINE_ITEMS
  - ORDERS
  - PRODUCTS
  - REGIONS
- User Defined Types

Selected:

Please choose the level for importing dependents:
○ None ● One Level ○ All Levels

4. Tables name will display under the Table entry.click on tables, then click on the **single right arrow**(**>**) in the middle of the screen



Import Metadata Wizard - Step 2 of 3: Object Selection

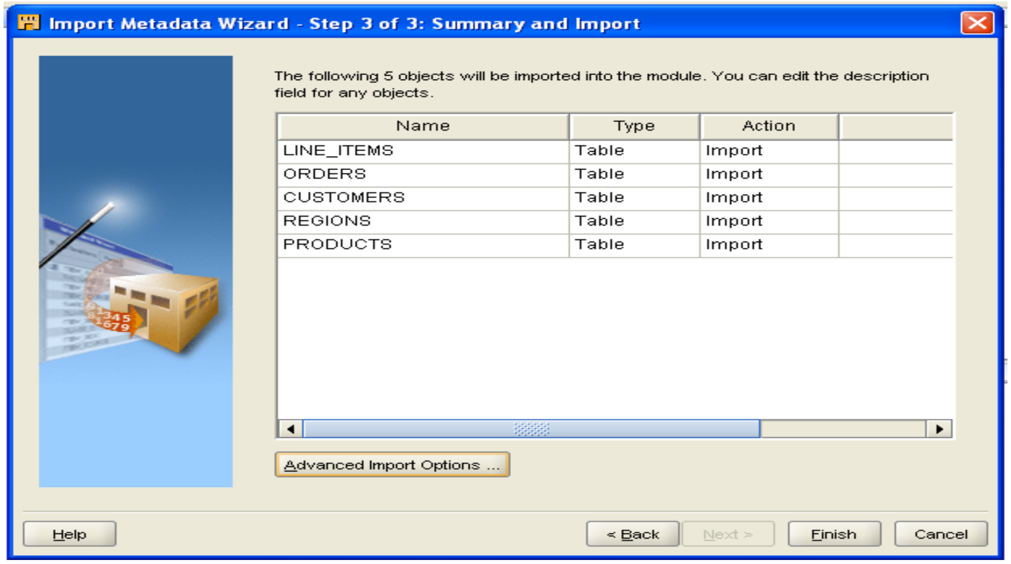
Select the objects to import into this module. Objects that are bold are already imported and will be re-imported when selected.

Available:
- Cubes
- Dimensions
- External Tables
- Materialized Views
- PL/SQL Transformations
- Sequences
- User Defined Types
- Views

Selected:
- Tables
  - CUSTOMERS
  - LINE_ITEMS
  - ORDERS
  - PRODUCTS
  - REGIONS

Please choose the level for importing dependents:
○ None ● One Level ○ All Levels

5. We'll click on the **Next** button | **summary and import page**| **Advance import option**…..button.

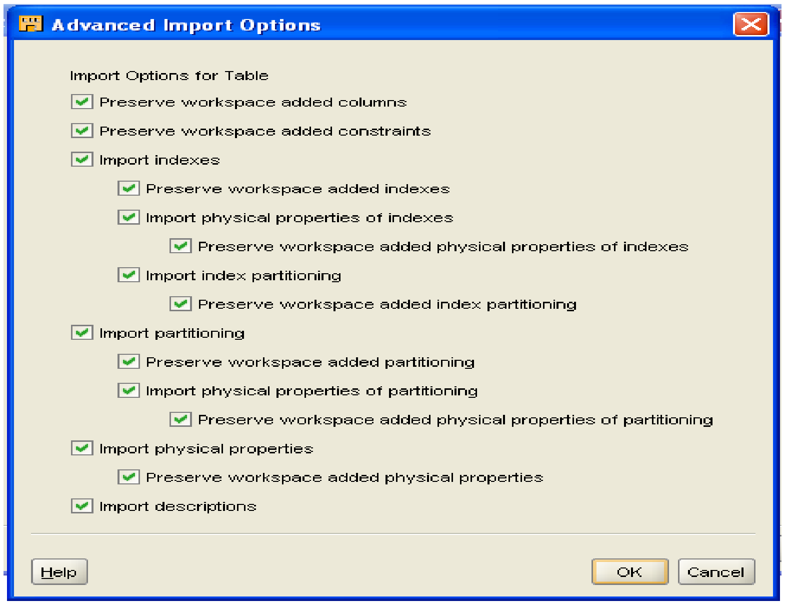


Import Metadata Wizard - Step 3 of 3: Summary and Import

The following 5 objects will be imported into the module. You can edit the description field for any objects.

Name	Type	Action
LINE_ITEMS	Table	Import
ORDERS	Table	Import
CUSTOMERS	Table	Import
REGIONS	Table	Import
PRODUCTS	Table	Import

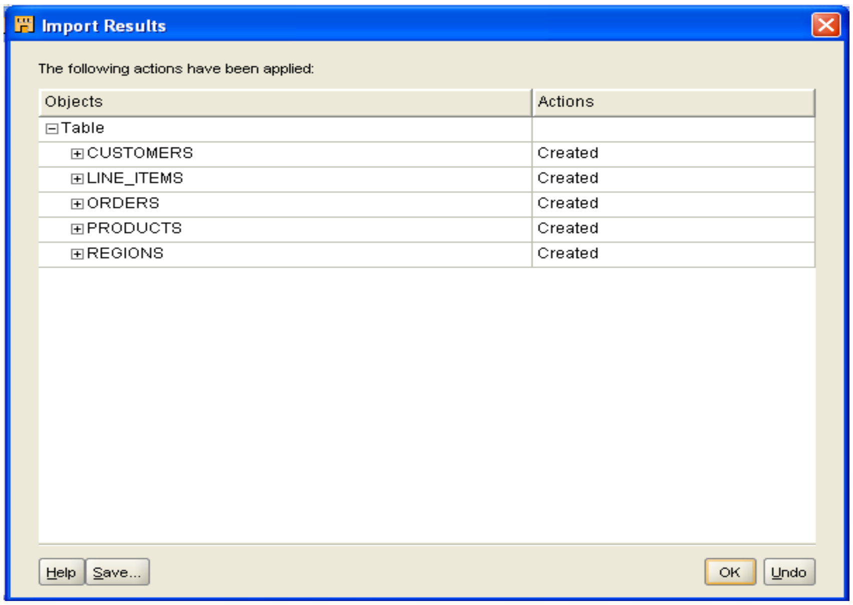
6. Following screen will appear.



Import Options for Table

- ☑ Preserve workspace added columns
- ☑ Preserve workspace added constraints
- ☑ Import indexes
  - ☑ Preserve workspace added indexes
  - ☑ Import physical properties of indexes
    - ☑ Preserve workspace added physical properties of indexes
  - ☑ Import index partitioning
    - ☑ Preserve workspace added index partitioning
- ☑ Import partitioning
  - ☑ Preserve workspace added partitioning
  - ☑ Import physical properties of partitioning
    - ☑ Preserve workspace added physical properties of partitioning
- ☑ Import physical properties
  - ☑ Preserve workspace added physical properties
- ☑ Import descriptions

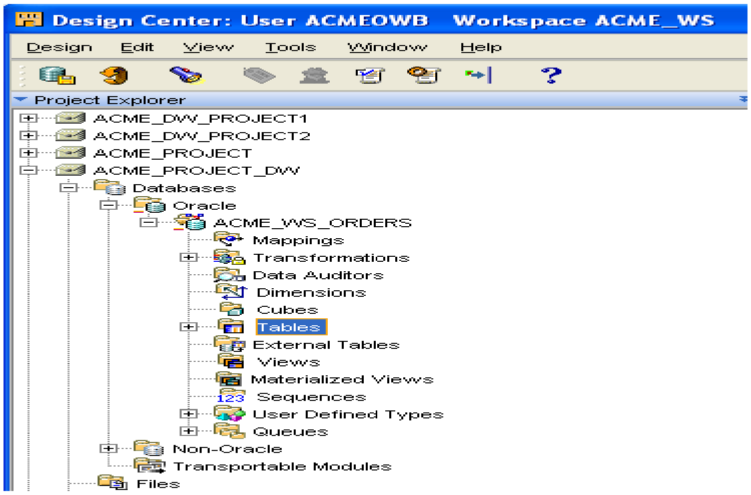
7.Click **OK**...buttton



The following actions have been applied:

Objects	Actions
⊟Table	
⊞CUSTOMERS	Created
⊞LINE_ITEMS	Created
⊞ORDERS	Created
⊞PRODUCTS	Created
⊞REGIONS	Created

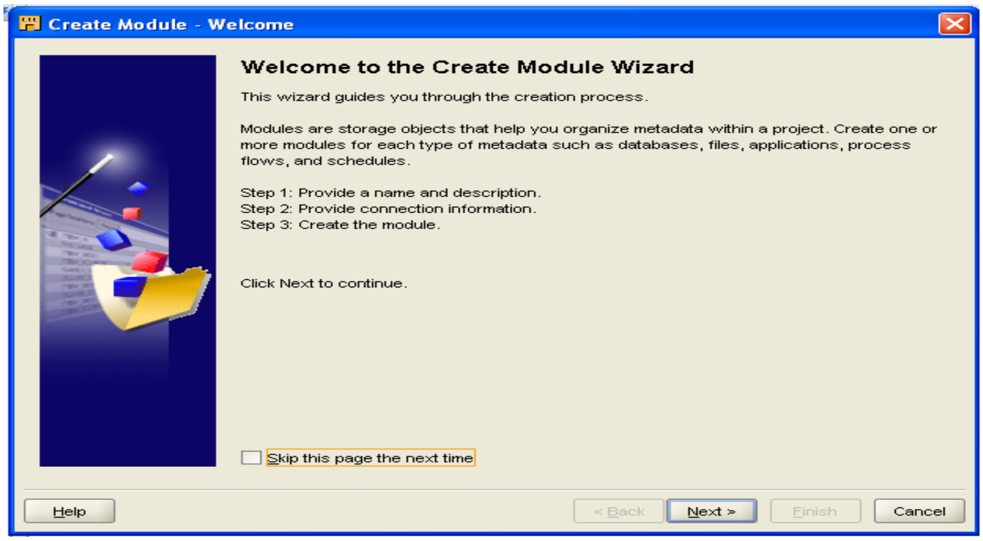
8. New table will be created



Defining source metadata manually with the Data Object Editor

Defining source metadata manully:Databases|Non_oracle|ODBC

1.Right click on the ODBE and select **New….** And Click on the **Next** button on **Welcome** screen



Welcome to the Create Module Wizard

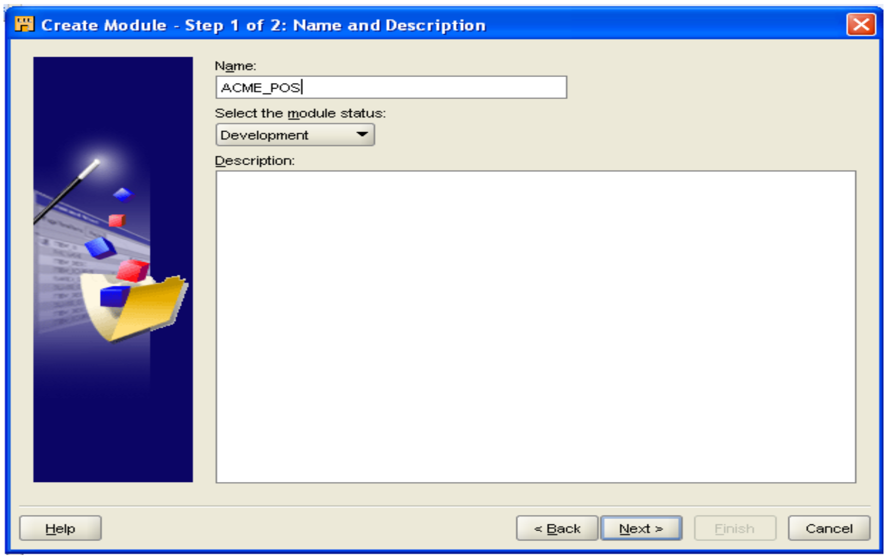
This wizard guides you through the creation process.

Modules are storage objects that help you organize metadata within a project. Create one or more modules for each type of metadata such as databases, files, applications, process flows, and schedules.

Step 1: Provide a name and description.
Step 2: Provide connection information.
Step 3: Create the module.

Click Next to continue.

2.Give the name of module as **ACME_POS** and select



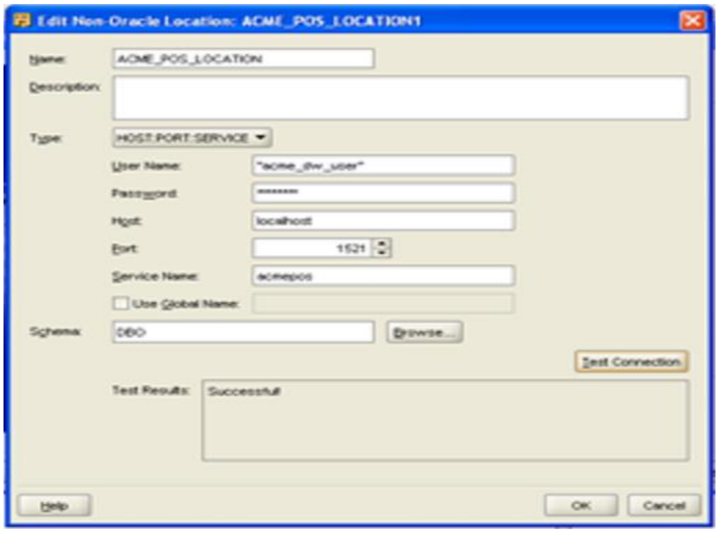
Create Module - Step 1 of 2: Name and Description

Name: ACME_POS

Select the module status: Development

Description:

3.fill the details



Edit Non-Oracle Location: ACME_POS_LOCATION1

Name: ACME_POS_LOCATION

Description:

Type: HOST:PORT:SERVICE

User Name: "acme_dw_user"

Password: *******

Host: localhost

Port: 1521

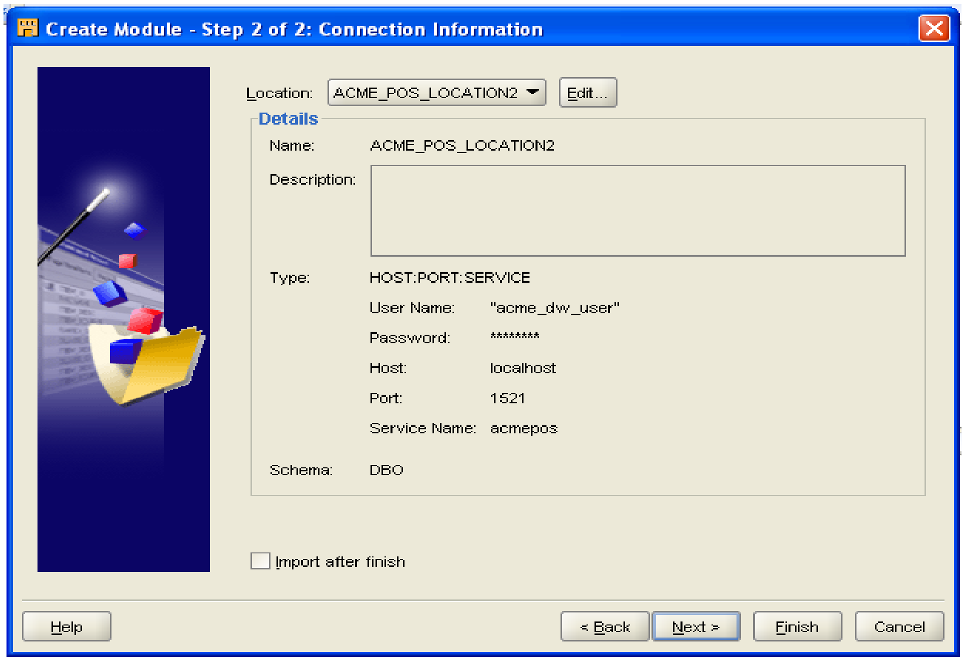
Service Name: acmepos

□ Use Global Name:

Schema: DBO

Test Results: Successfull

4.uncheck **import after finish**



Location: ACME_POS_LOCATION2 Edit...

Details

Name: ACME_POS_LOCATION2

Description:

Type: HOST:PORT:SERVICE

User Name: "acme_dw_user"

Password: ********

Host: localhost

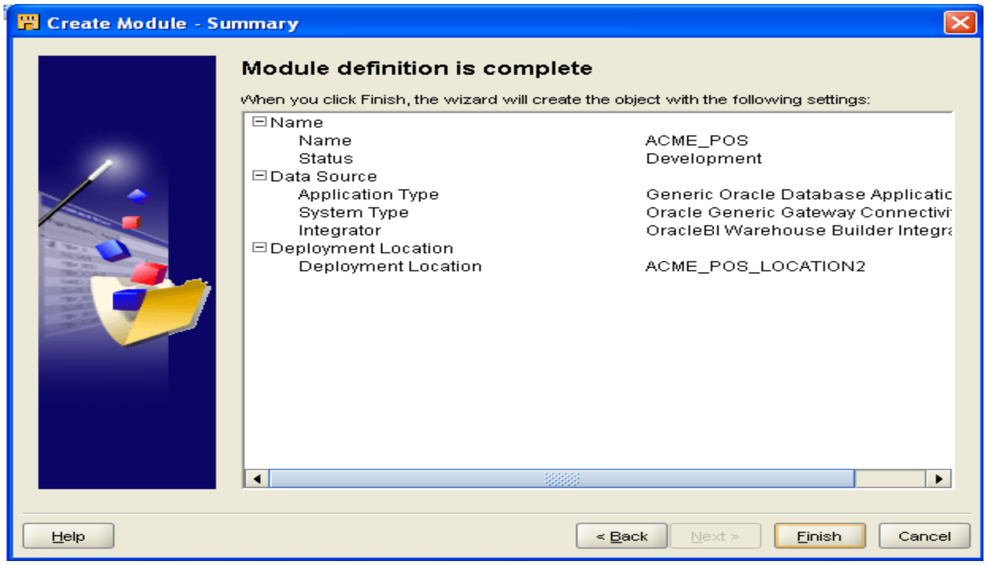
Port: 1521

Service Name: acmepos

Schema: DBO

□ Import after finish

5.click on **Finish**

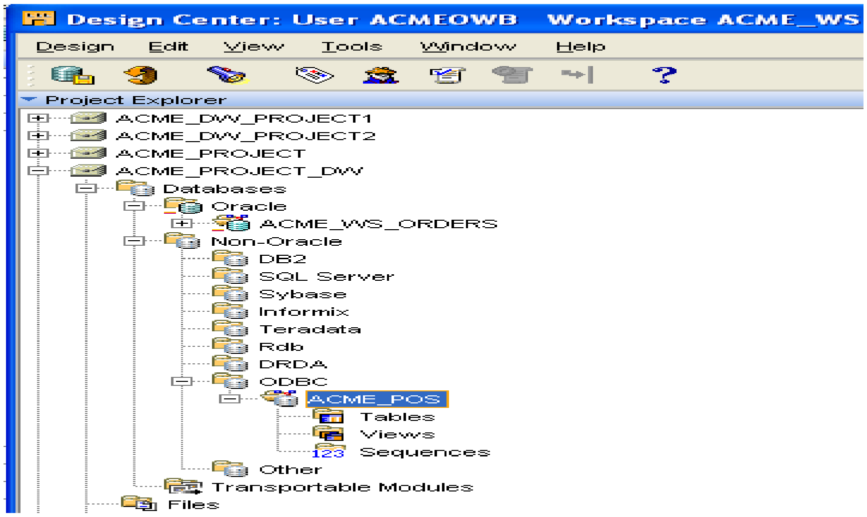


Module definition is complete

When you click Finish, the wizard will create the object with the following settings:

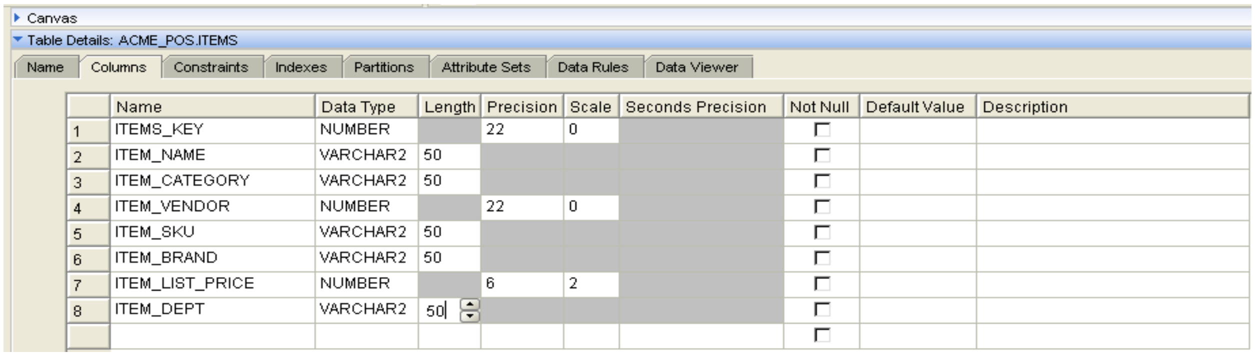
```
⊟ Name
      Name                               ACME_POS
      Status                             Development
⊟ Data Source
      Application Type                   Generic Oracle Database Applicatic
      System Type                        Oracle Generic Gateway Connectivi
      Integrator                         OracleBI Warehouse Builder Integra
⊟ Deployment Location
      Deployment Location                ACME_POS_LOCATION2
```

6.Created ACME_POS

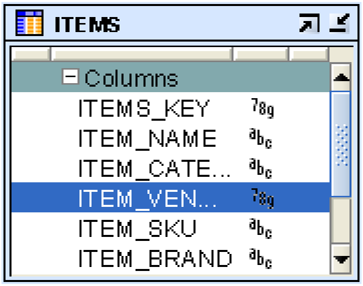


7.Add the following attributes in the table

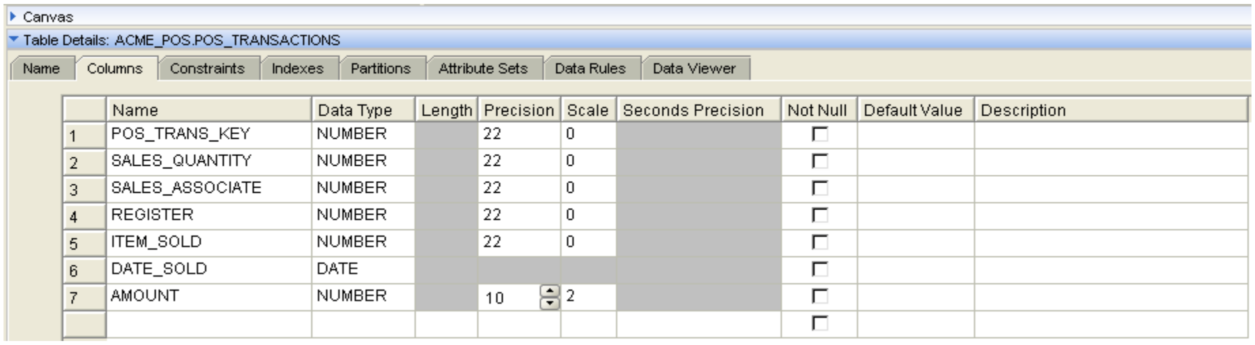
1.ITEMS



	Name	Data Type	Length	Precision	Scale	Seconds Precision	Not Null	Default Value	Description
1	ITEMS_KEY	NUMBER		22	0		☐		
2	ITEM_NAME	VARCHAR2	50				☐		
3	ITEM_CATEGORY	VARCHAR2	50				☐		
4	ITEM_VENDOR	NUMBER		22	0		☐		
5	ITEM_SKU	VARCHAR2	50				☐		
6	ITEM_BRAND	VARCHAR2	50				☐		
7	ITEM_LIST_PRICE	NUMBER		6	2		☐		
8	ITEM_DEPT	VARCHAR2	50				☐		
							☐		



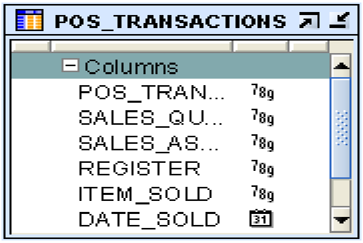
2.POS_TRANSACTIONS



Canvas

Table Details: ACME_POS.POS_TRANSACTIONS

	Name	Data Type	Length	Precision	Scale	Seconds Precision	Not Null	Default Value	Description
1	POS_TRANS_KEY	NUMBER		22	0		□		
2	SALES_QUANTITY	NUMBER		22	0		□		
3	SALES_ASSOCIATE	NUMBER		22	0		□		
4	REGISTER	NUMBER		22	0		□		
5	ITEM_SOLD	NUMBER		22	0		□		
6	DATE_SOLD	DATE					□		
7	AMOUNT	NUMBER		10	2		□		
							□		

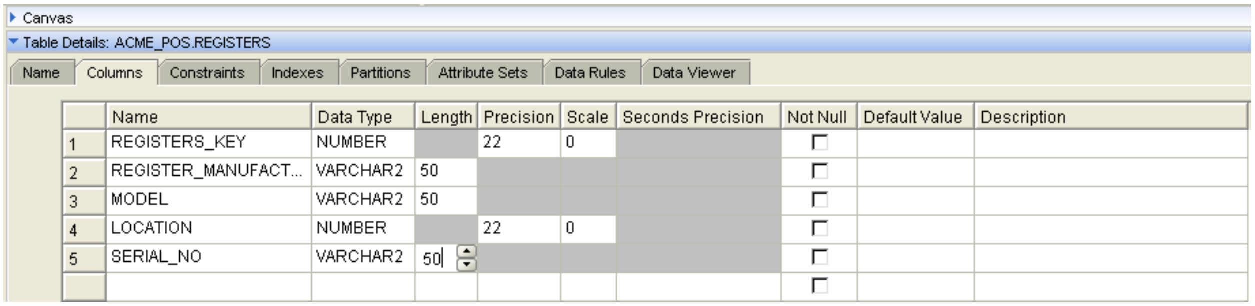


POS_TRANSACTIONS

Columns

POS_TRAN...	789
SALES_QU...	789
SALES_AS...	789
REGISTER	789
ITEM_SOLD	789
DATE_SOLD	31

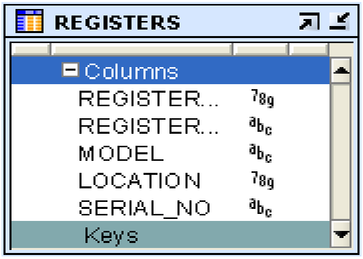
3.Registers



Canvas

Table Details: ACME_POS.REGISTERS

	Name	Data Type	Length	Precision	Scale	Seconds Precision	Not Null	Default Value	Description
1	REGISTERS_KEY	NUMBER		22	0		□		
2	REGISTER_MANUFACT...	VARCHAR2	50				□		
3	MODEL	VARCHAR2	50				□		
4	LOCATION	NUMBER		22	0		□		
5	SERIAL_NO	VARCHAR2	50				□		
							□		



REGISTERS

Columns

REGISTER...	789
REGISTER...	abc
MODEL	abc
LOCATION	789
SERIAL_NO	abc

Keys

4.STORES

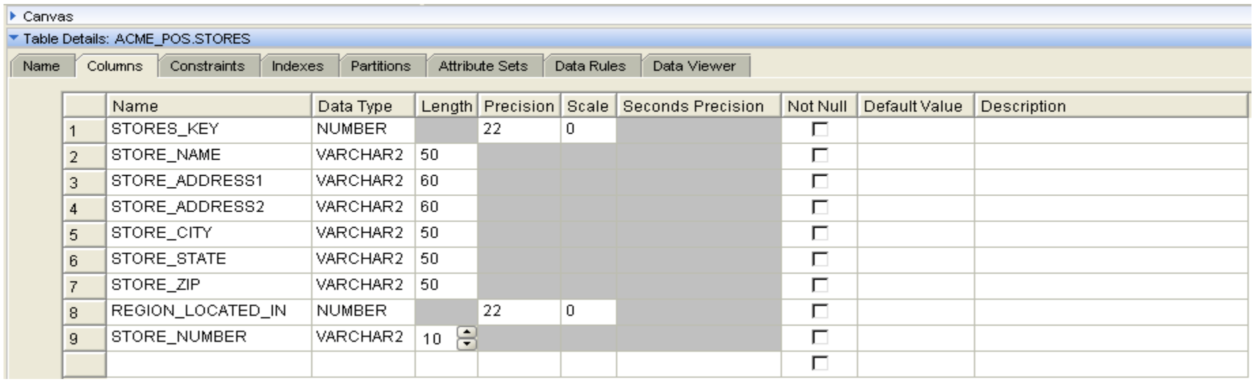
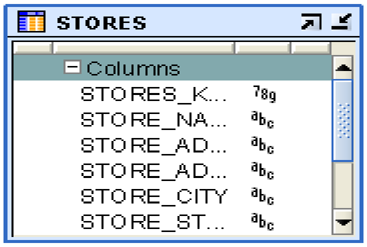


Table Details: ACME_POS.STORES

	Name	Data Type	Length	Precision	Scale	Seconds Precision	Not Null	Default Value	Description
1	STORES_KEY	NUMBER		22	0		□		
2	STORE_NAME	VARCHAR2	50				□		
3	STORE_ADDRESS1	VARCHAR2	60				□		
4	STORE_ADDRESS2	VARCHAR2	60				□		
5	STORE_CITY	VARCHAR2	50				□		
6	STORE_STATE	VARCHAR2	50				□		
7	STORE_ZIP	VARCHAR2	50				□		
8	REGION_LOCATED_IN	NUMBER		22	0		□		
9	STORE_NUMBER	VARCHAR2	10				□		
							□		



STORES

Columns

STORES_K...	789
STORE_NA...	abc
STORE_AD...	abc
STORE_AD...	abc
STORE_CITY	abc
STORE_ST...	abc

5.REGIONS

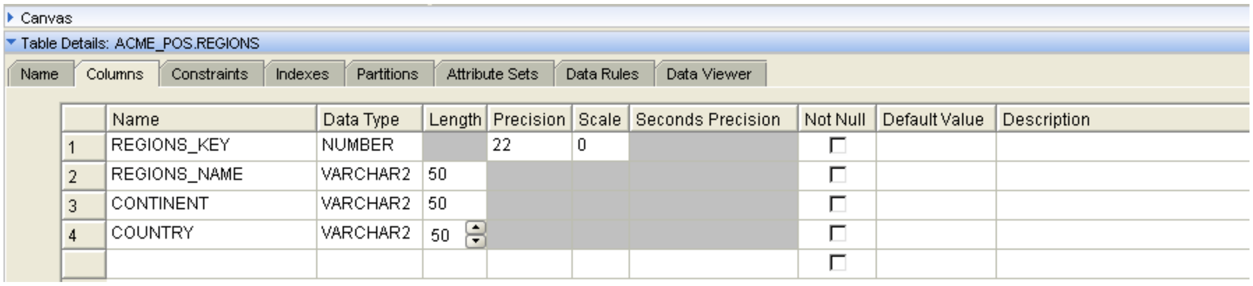
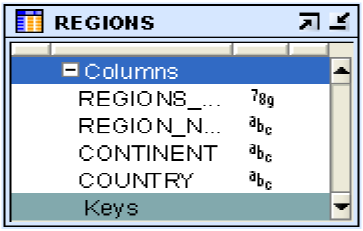


Table Details: ACME_POS.REGIONS

	Name	Data Type	Length	Precision	Scale	Seconds Precision	Not Null	Default Value	Description
1	REGIONS_KEY	NUMBER		22	0		□		
2	REGIONS_NAME	VARCHAR2	50				□		
3	CONTINENT	VARCHAR2	50				□		
4	COUNTRY	VARCHAR2	50				□		
							□		



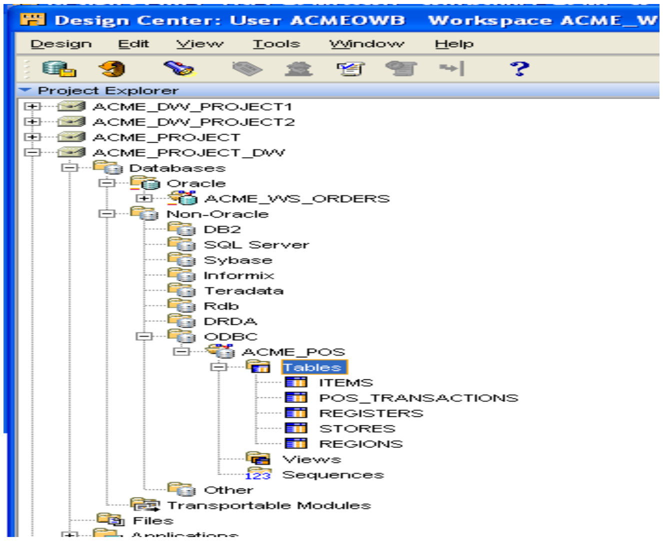
REGIONS

Columns

REGIONS_...	789
REGION_N...	abc
CONTINENT	abc
COUNTRY	abc

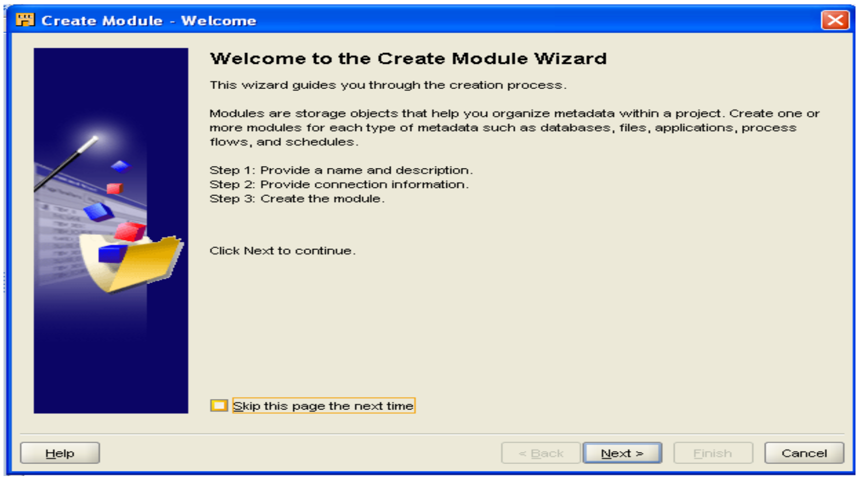
Keys

8.Tables will be created.



5)Importing source metadata from files

1.**Project Explorer** we will see that there is **files** node right below the **Database** node Click on the next button on **Welcome** screen



Welcome to the Create Module Wizard

This wizard guides you through the creation process.

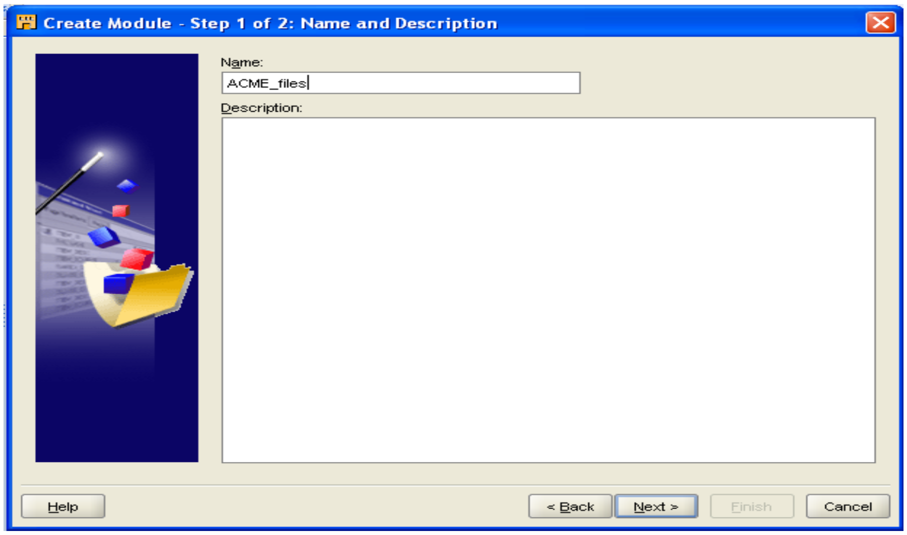
Modules are storage objects that help you organize metadata within a project. Create one or more modules for each type of metadata such as databases, files, applications, process flows, and schedules.

Step 1: Provide a name and description.
Step 2: Provide connection information.
Step 3: Create the module.

Click Next to continue.

□ Skip this page the next time

2.Give the name of module as ACME_FILES and select

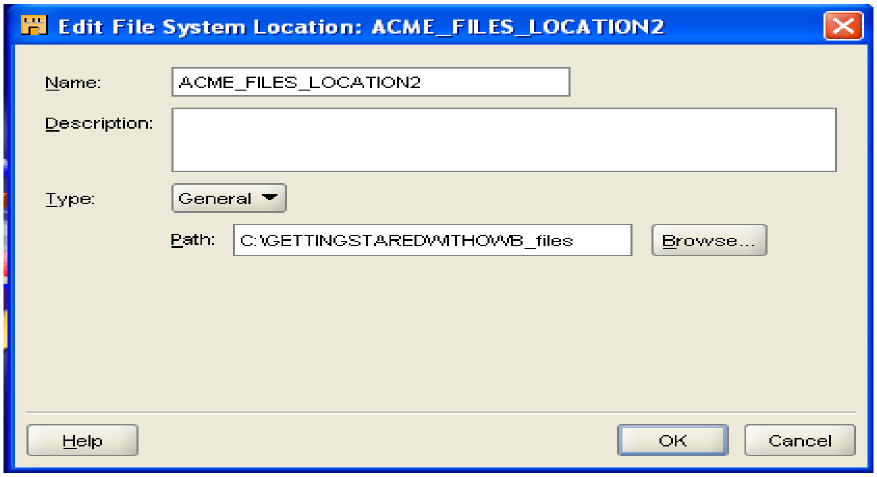


Create Module - Step 1 of 2: Name and Description

Name: ACME_files

Description:

3.Click on **edit** button search for the folder where we are store the counties file with the help of **Browse……** button



Edit File System Location: ACME_FILES_LOCATION2

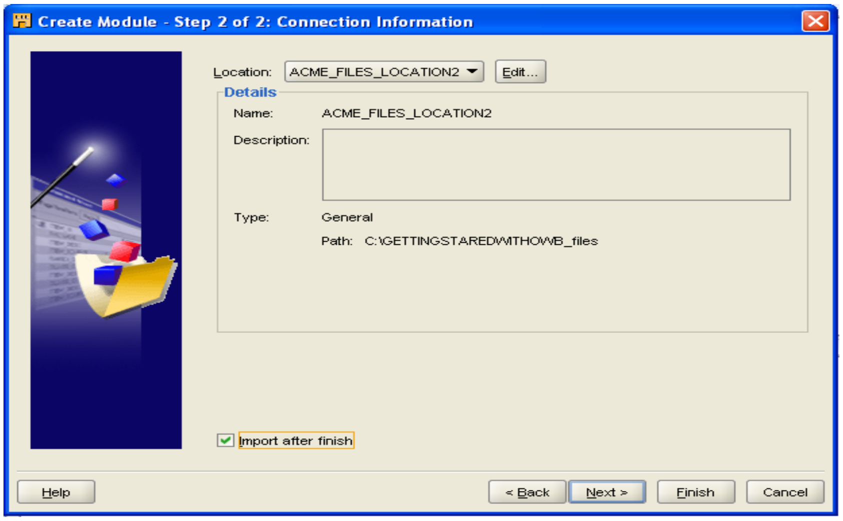
Name: ACME_FILES_LOCATION2

Description:

Type: General

Path: C:\GETTINGSTAREDWITHOWB_files

7.Check **import after finish**



Create Module - Step 2 of 2: Connection Information

Location: ACME_FILES_LOCATION2

Details

Name: ACME_FILES_LOCATION2

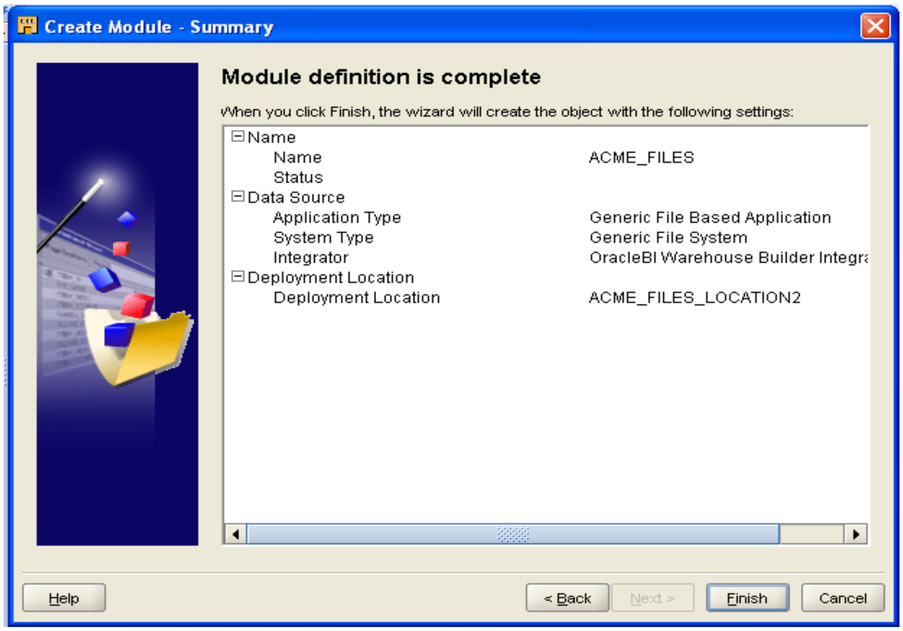
Description:

Type: General

Path: C:\GETTINGSTAREDWITHOWB_files

☑ Import after finish

8.Click on **Finish** button



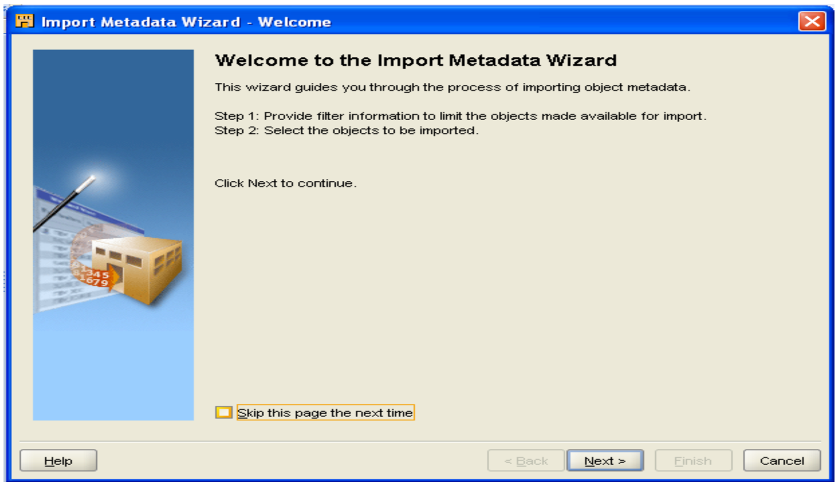
Create Module - Summary

Module definition is complete

When you click Finish, the wizard will create the object with the following settings:

```
⊟Name
    Name                      ACME_FILES
    Status
⊟Data Source
    Application Type          Generic File Based Application
    System Type               Generic File System
    Integrator                OracleBI Warehouse Builder Integra
⊟Deployment Location
    Deployment Location       ACME_FILES_LOCATION2
```

9.**Import Metadata Wizard** appears .Click on **next** button on **Welcome** screen



Import Metadata Wizard - Welcome

Welcome to the Import Metadata Wizard

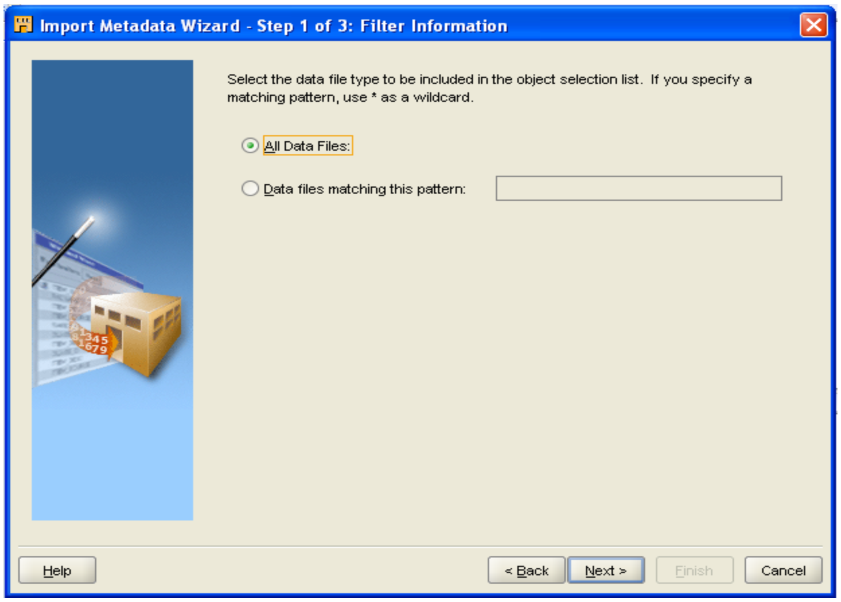
This wizard guides you through the process of importing object metadata.

Step 1: Provide filter information to limit the objects made available for import.
Step 2: Select the objects to be imported.

Click Next to continue.

□ Skip this page the next time

10.Select **All Data files** then click on **next** button



Import Metadata Wizard - Step 1 of 3: Filter Information

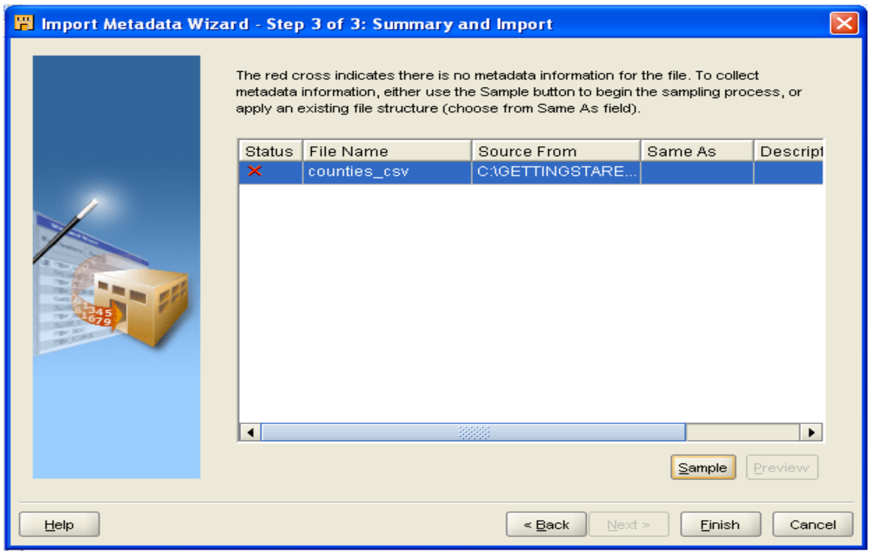
Select the data file type to be included in the object selection list. If you specify a matching pattern, use * as a wildcard.

● All Data Files:

○ Data files matching this pattern:

11.In Object Selection the counties file click on the **single right arrow**(**>**) in the middle of the screen

12.Click on **Sample** button

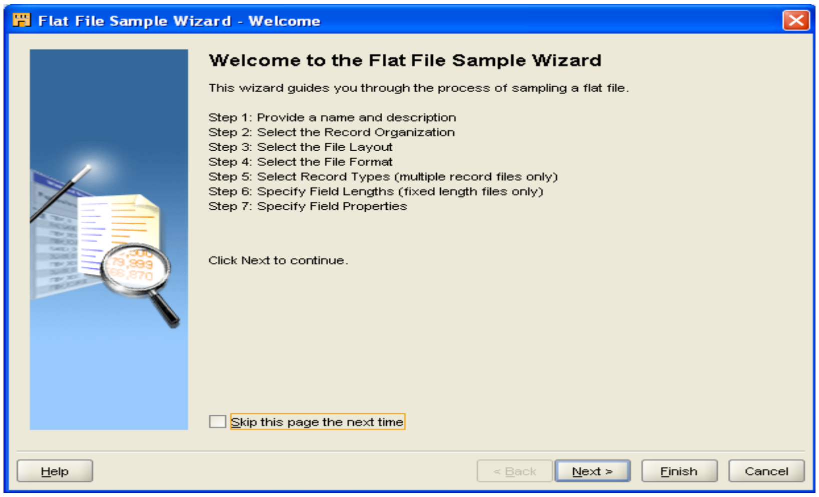


Import Metadata Wizard - Step 3 of 3: Summary and Import

The red cross indicates there is no metadata information for the file. To collect metadata information, either use the Sample button to begin the sampling process, or apply an existing file structure (choose from Same As field).

Status	File Name	Source From	Same As	Descript
✗	counties_csv	C:\GETTINGSTARE...		

13.click on **next** button



Welcome to the Flat File Sample Wizard

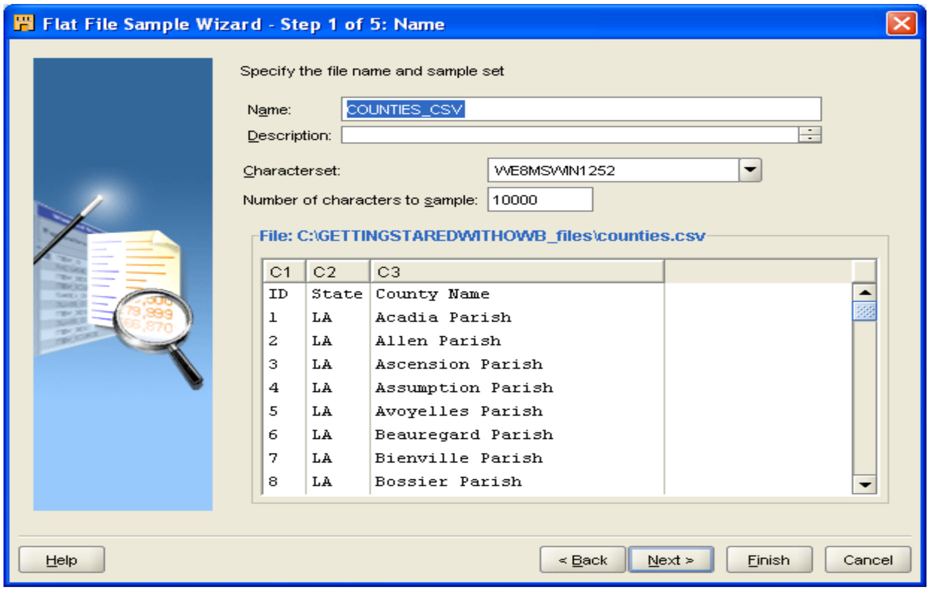
This wizard guides you through the process of sampling a flat file.

Step 1: Provide a name and description
Step 2: Select the Record Organization
Step 3: Select the File Layout
Step 4: Select the File Format
Step 5: Select Record Types (multiple record files only)
Step 6: Specify Field Lengths (fixed length files only)
Step 7: Specify Field Properties

Click Next to continue.

□ Skip this page the next time

14. click on **next** button



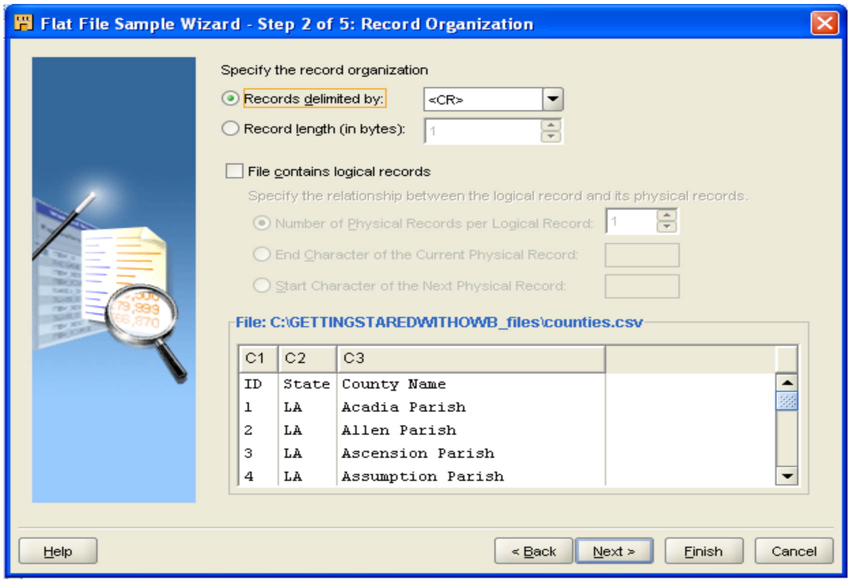
Specify the file name and sample set

Name: COUNTIES_CSV
Description:
Characterset: WE8MSWIN1252
Number of characters to sample: 10000

File: C:\GETTINGSTAREDWITHOWB_files\counties.csv

C1	C2	C3
ID	State	County Name
1	LA	Acadia Parish
2	LA	Allen Parish
3	LA	Ascension Parish
4	LA	Assumption Parish
5	LA	Avoyelles Parish
6	LA	Beauregard Parish
7	LA	Bienville Parish
8	LA	Bossier Parish

15.Select **Records delimited by** .click next



Flat File Sample Wizard - Step 2 of 5: Record Organization

Specify the record organization

● Records delimited by: <CR>

○ Record length (in bytes): 1

□ File contains logical records

Specify the relationship between the logical record and its physical records.

● Number of Physical Records per Logical Record: 1

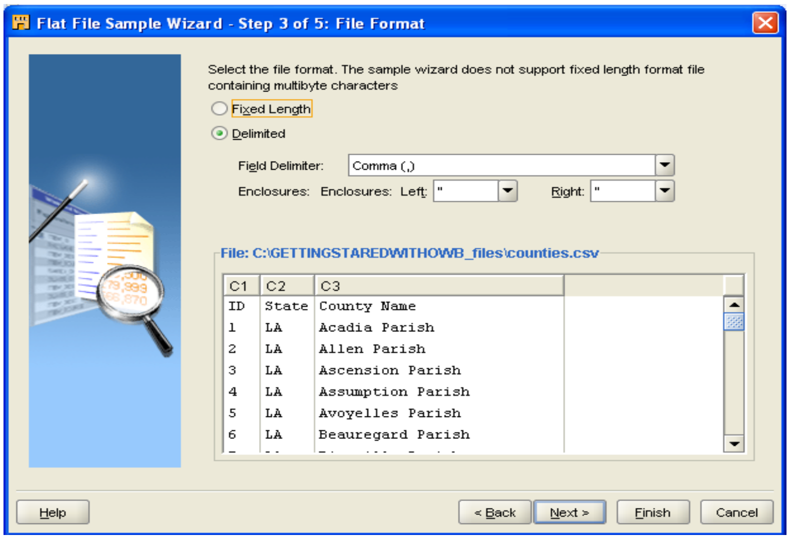
○ End Character of the Current Physical Record:

○ Start Character of the Next Physical Record:

File: C:\GETTINGSTAREDWITHOWB_files\counties.csv

C1	C2	C3
ID	State	County Name
1	LA	Acadia Parish
2	LA	Allen Parish
3	LA	Ascension Parish
4	LA	Assumption Parish

16.Select **delimited** then click on **next** button



Flat File Sample Wizard - Step 3 of 5: File Format

Select the file format. The sample wizard does not support fixed length format file containing multibyte characters

○ Fixed Length

● Delimited

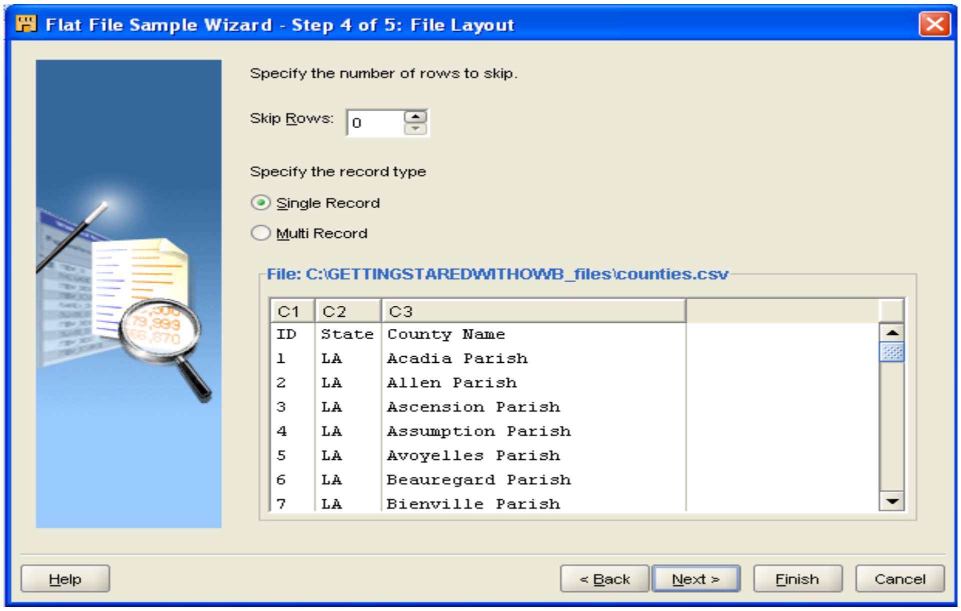
Field Delimiter: Comma (,)

Enclosures: Enclosures: Left: " Right: "

File: C:\GETTINGSTAREDWITHOWB_files\counties.csv

C1	C2	C3
ID	State	County Name
1	LA	Acadia Parish
2	LA	Allen Parish
3	LA	Ascension Parish
4	LA	Assumption Parish
5	LA	Avoyelles Parish
6	LA	Beauregard Parish

17.Select **Single Records** then click on **next** button



Flat File Sample Wizard - Step 4 of 5: File Layout

Specify the number of rows to skip.

Skip Rows: 0

Specify the record type

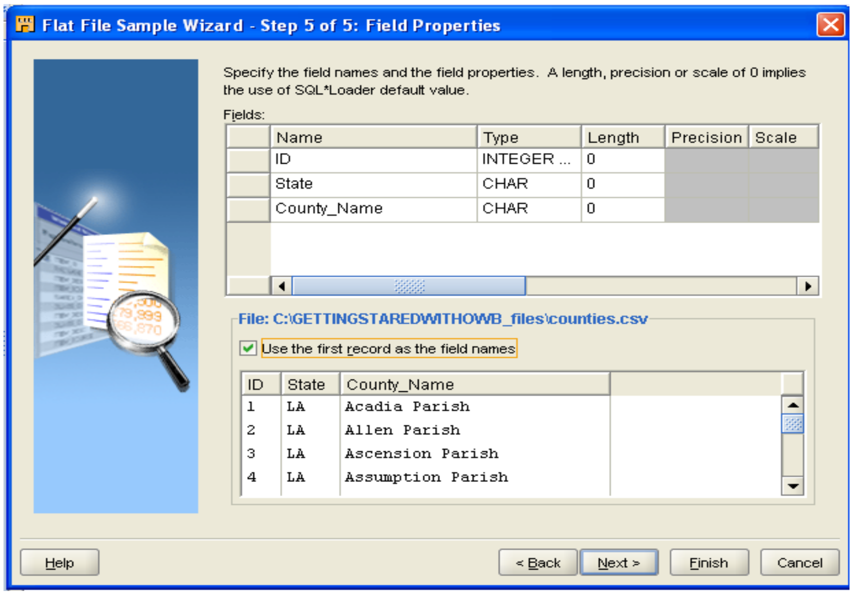
● Single Record

○ Multi Record

File: C:\GETTINGSTAREDWITHOWB_files\counties.csv

C1	C2	C3
ID	State	County Name
1	LA	Acadia Parish
2	LA	Allen Parish
3	LA	Ascension Parish
4	LA	Assumption Parish
5	LA	Avoyelles Parish
6	LA	Beauregard Parish
7	LA	Bienville Parish

18.check **Use the first record as the field names** then click on **next** button



Flat File Sample Wizard - Step 5 of 5: Field Properties

Specify the field names and the field properties. A length, precision or scale of 0 implies the use of SQL*Loader default value.

Fields:

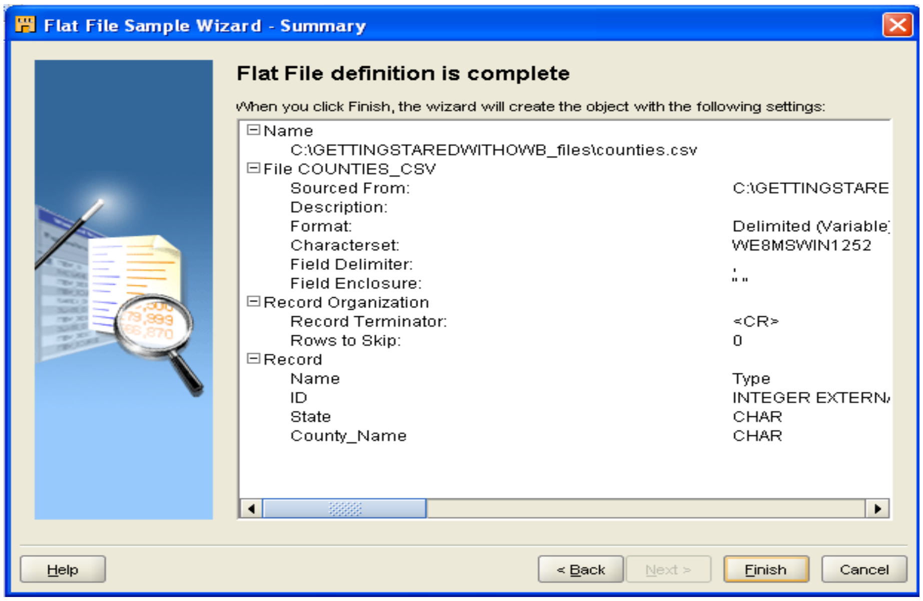
Name	Type	Length	Precision	Scale
ID	INTEGER ...	0		
State	CHAR	0		
County_Name	CHAR	0		

File: C:\GETTINGSTAREDWITHOWB_files\counties.csv

☑ Use the first record as the field names

ID	State	County_Name
1	LA	Acadia Parish
2	LA	Allen Parish
3	LA	Ascension Parish
4	LA	Assumption Parish

19.Click on **finish……**



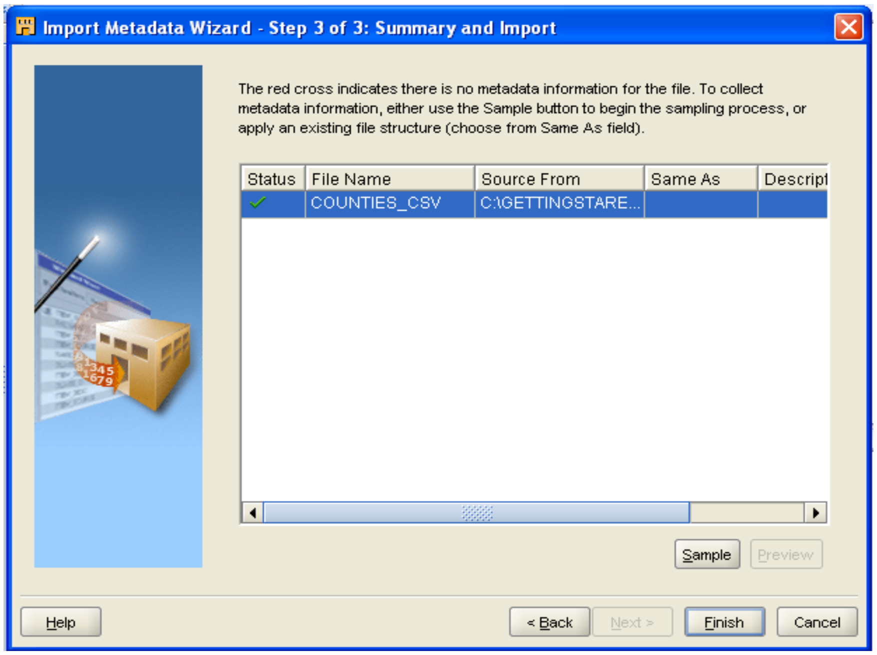
Flat File Sample Wizard - Summary

Flat File definition is complete

When you click Finish, the wizard will create the object with the following settings:

```
⊟Name
    C:\GETTINGSTAREDWITHOWB_files\counties.csv
⊟File COUNTIES_CSV
    Sourced From:                       C:\GETTINGSTARE
    Description:
    Format:                             Delimited (Variable)
    Characterset:                       WE8MSWIN1252
    Field Delimiter:                    ,
    Field Enclosure:                    " "
⊟Record Organization
    Record Terminator:                  <CR>
    Rows to Skip:                       0
⊟Record
    Name                                Type
    ID                                  INTEGER EXTERN/
    State                               CHAR
    County_Name                         CHAR
```

20.The **status** are in check format then click on **finish** button

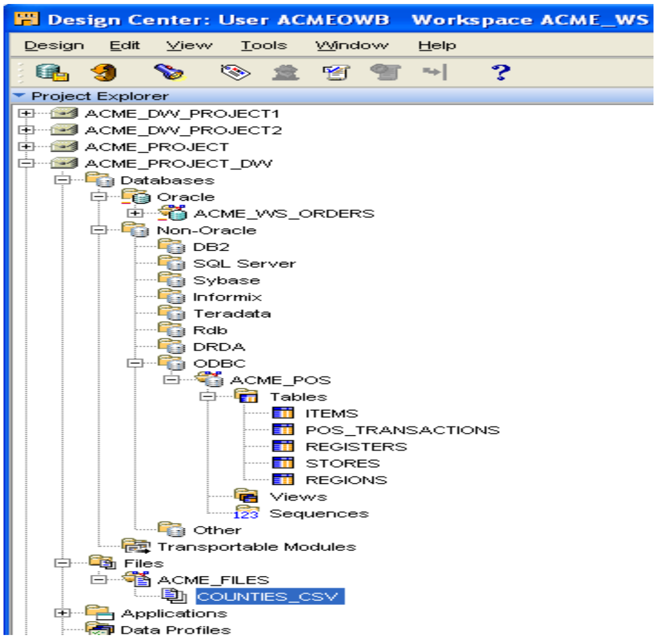


Import Metadata Wizard - Step 3 of 3: Summary and Import

The red cross indicates there is no metadata information for the file. To collect metadata information, either use the Sample button to begin the sampling process, or apply an existing file structure (choose from Same As field).

Status	File Name	Source From	Same As	Descript
✓	COUNTIES_CSV	C:\GETTINGSTARE...		

21.Counties_CSV file will be created.

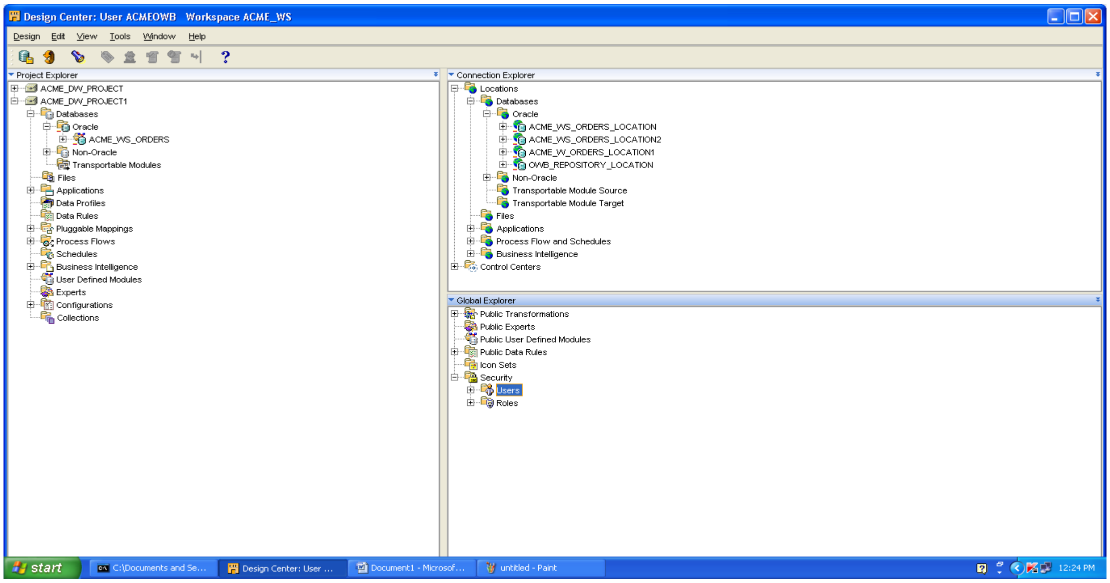


PRACTICAL NO. 02

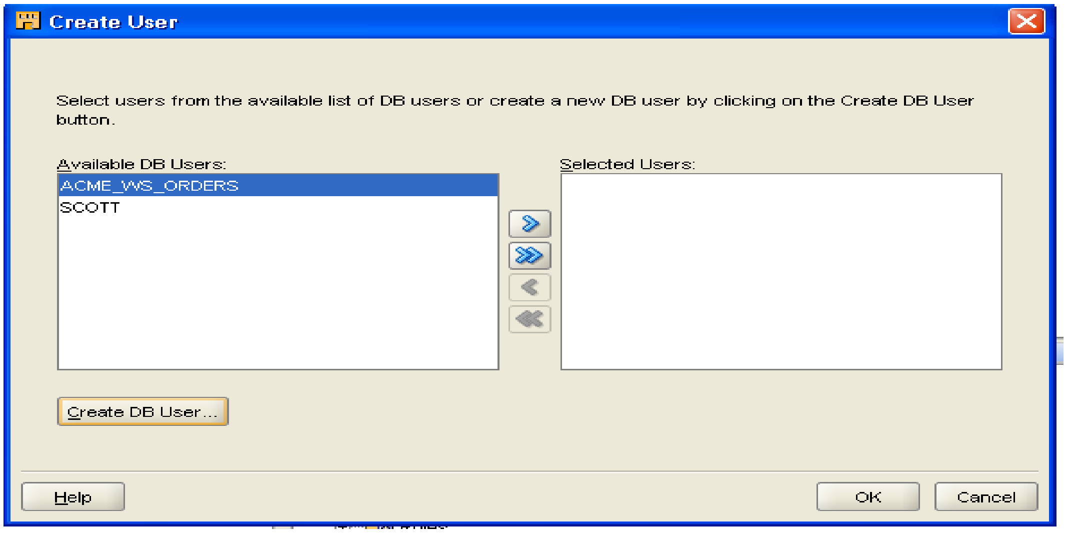
Aim: Design Target Data Structure in Oracle

Creating a Target User

1: Design Center | Global Explorer | Security |Users



2. Right Click on the Users click on **New** and then Click on **Create DB User..**



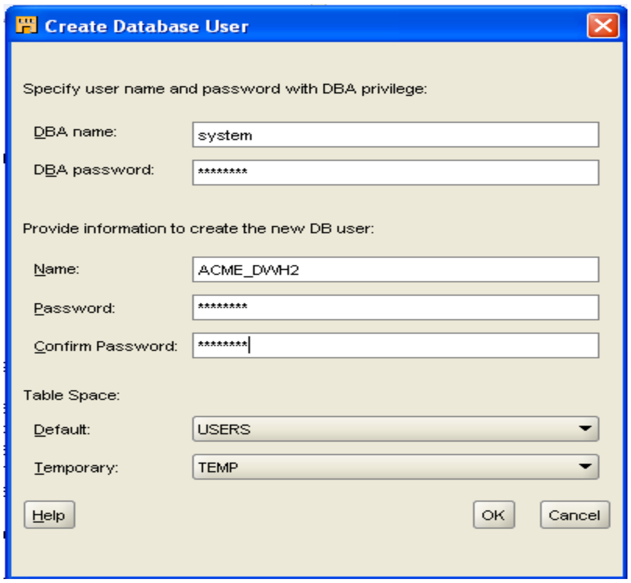
Select users from the available list of DB users or create a new DB user by clicking on the Create DB User button.

Available DB Users:

ACME_WS_ORDERS
SCOTT

Selected Users:

3. Following Screen will appear and click on **OK** button



Create Database User

Specify user name and password with DBA privilege:

DBA name: system

DBA password: ********

Provide information to create the new DB user:

Name: ACME_DWH2

Password: ********

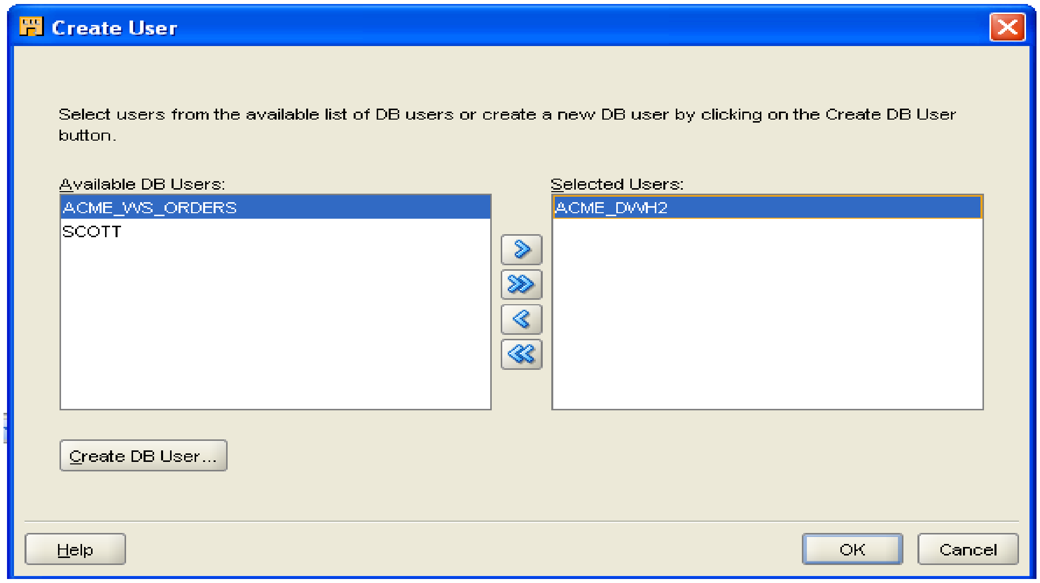
Confirm Password: ********

Table Space:

Default: USERS

Temporary: TEMP

4.Click on **OK**... button



Create User

Select users from the available list of DB users or create a new DB user by clicking on the Create DB User button.

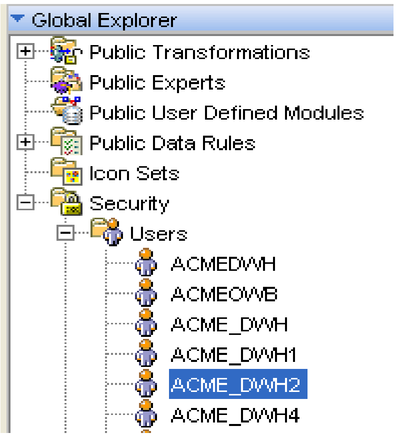
Available DB Users:

ACME_WS_ORDERS
SCOTT

Selected Users:

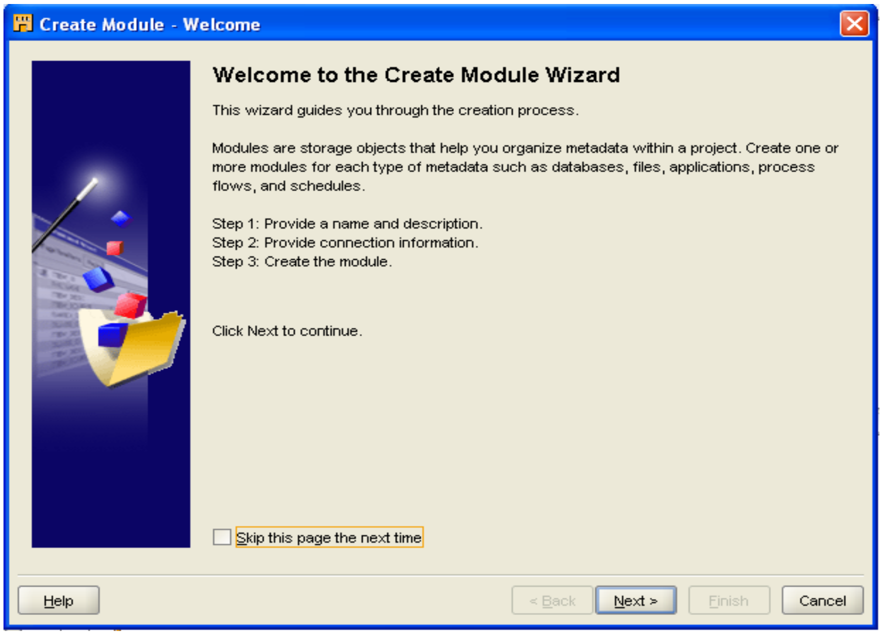
ACME_DWH2

5.**Target User** will be created



Create Target Module

1. Create the ACME_WS_ORDERS | Oracle Object | Database |Select New … | Create Module Wizard | ACME_DWH |Click on **Next** on Welcome Screen



Welcome to the Create Module Wizard

This wizard guides you through the creation process.

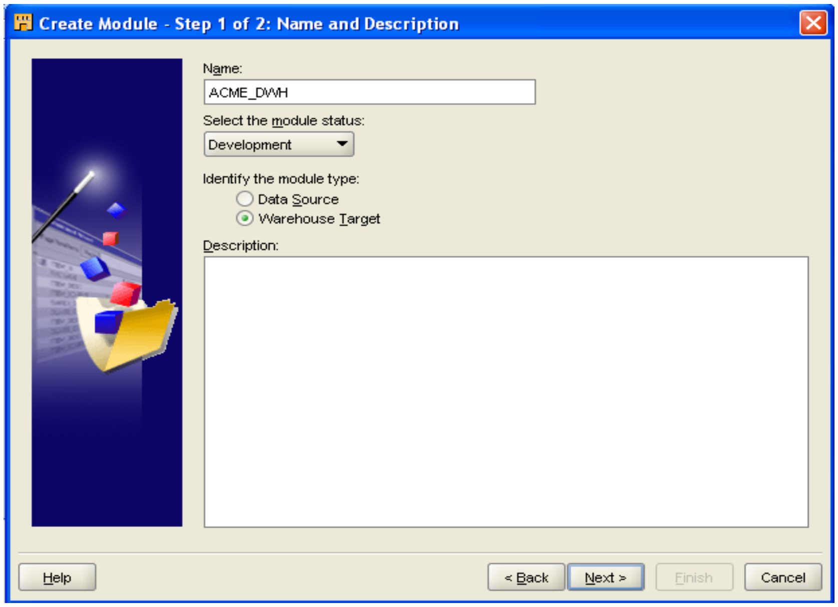
Modules are storage objects that help you organize metadata within a project. Create one or more modules for each type of metadata such as databases, files, applications, process flows, and schedules.

Step 1: Provide a name and description.
Step 2: Provide connection information.
Step 3: Create the module.

Click Next to continue.

☐ Skip this page the next time

2.Following Screen will Appear | Click on the **Next** Button



Create Module - Step 1 of 2: Name and Description

Name: ACME_DWH

Select the module status: Development

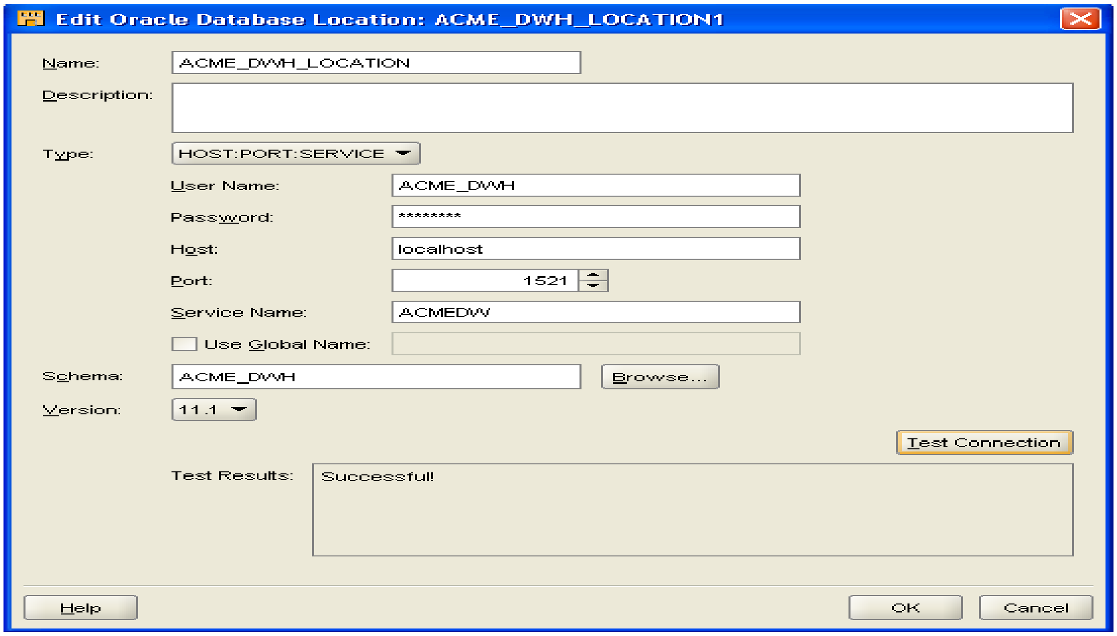
Identify the module type:

○ Data Source

● Warehouse Target

Description:

3. . Fill the Fields | Click on **Test Connection** | Click on **OK..** button



Edit Oracle Database Location: ACME_DWH_LOCATION1

Name: ACME_DWH_LOCATION

Description:

Type: HOST:PORT:SERVICE

User Name: ACME_DWH

Password: ********

Host: localhost

Port: 1521

Service Name: ACMEDW

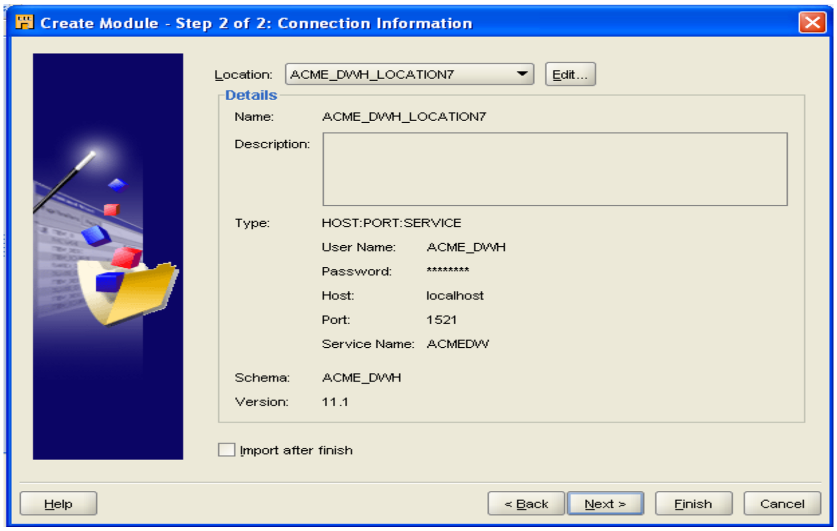
□ Use Global Name:

Schema: ACME_DWH

Version: 11.1

Test Results: Successfull!

4. Click on the **Next..** Button



Create Module - Step 2 of 2: Connection Information

Location: ACME_DWH_LOCATION7

Details

Name: ACME_DWH_LOCATION7

Description:

Type: HOST:PORT:SERVICE

User Name: ACME_DWH

Password: ********

Host: localhost

Port: 1521

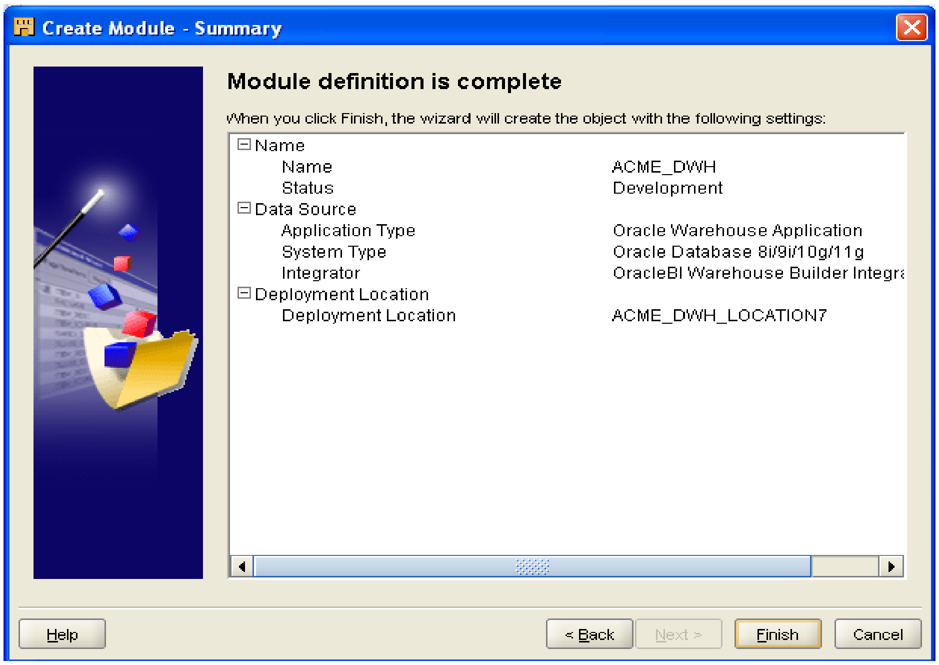
Service Name: ACMEDW

Schema: ACME_DWH

Version: 11.1

□ Import after finish

5.Click on **finish** button



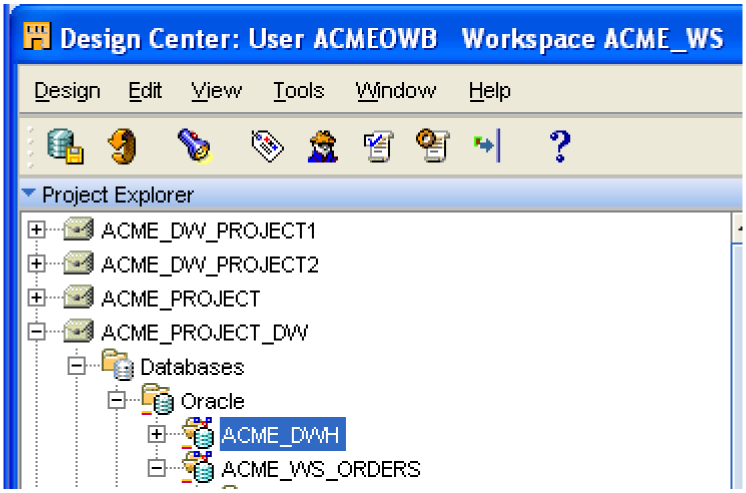
Create Module - Summary

Module definition is complete

When you click Finish, the wizard will create the object with the following settings:

```
⊟ Name
    Name                      ACME_DWH
    Status                    Development
⊟ Data Source
    Application Type          Oracle Warehouse Application
    System Type               Oracle Database 8i/9i/10g/11g
    Integrator                OracleBI Warehouse Builder Integra
⊟ Deployment Location
    Deployment Location       ACME_DWH_LOCATION7
```

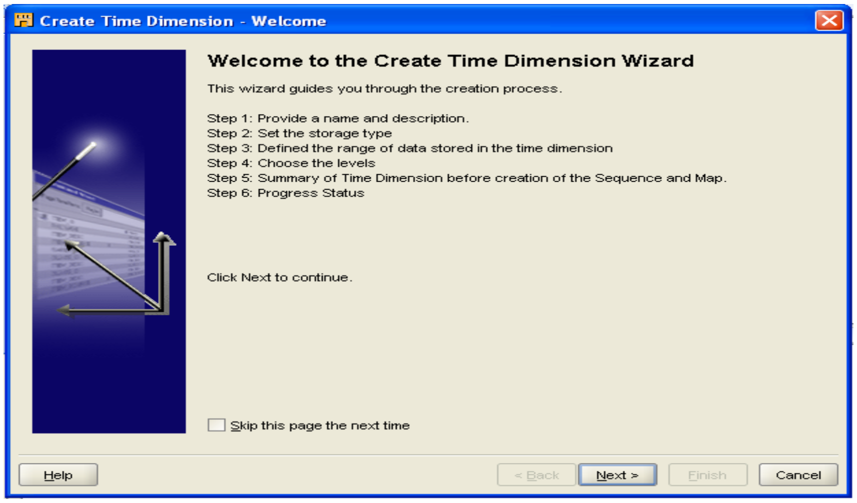
6.**Target Module** will be created



PRACTICAL NO: 3

AIM: Create target structure in OWB

1. In **Project Explorer**| **ACME_DW_PROJECT**|**Databases** Node|**Oracle**| ACME_DWH then right click on the Dimensions Node|**New**|**Using Time Wizard…** Then click on **NEXT…** button



Welcome to the Create Time Dimension Wizard

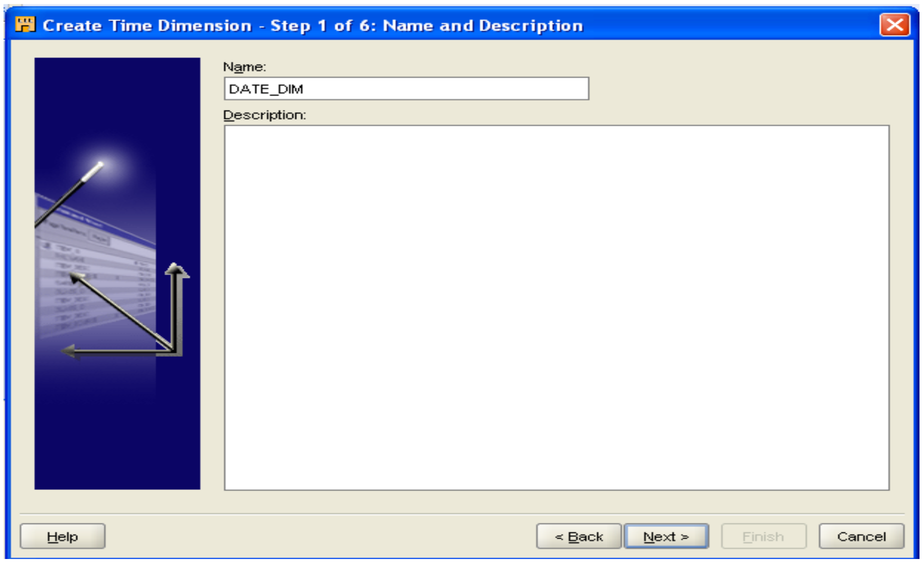
This wizard guides you through the creation process.

Step 1: Provide a name and description.
Step 2: Set the storage type
Step 3: Defined the range of data stored in the time dimension
Step 4: Choose the levels
Step 5: Summary of Time Dimension before creation of the Sequence and Map.
Step 6: Progress Status

Click Next to continue.

□ Skip this page the next time

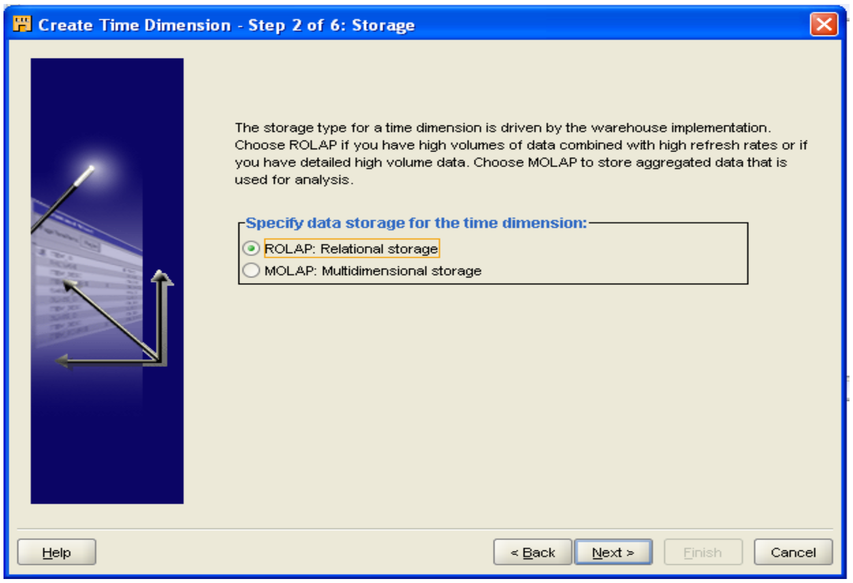
2.Give Name as DATE_DIM. Then click NEXT.



Name: DATE_DIM

Description:

3.Select “ROLAP” and click on NEXT button



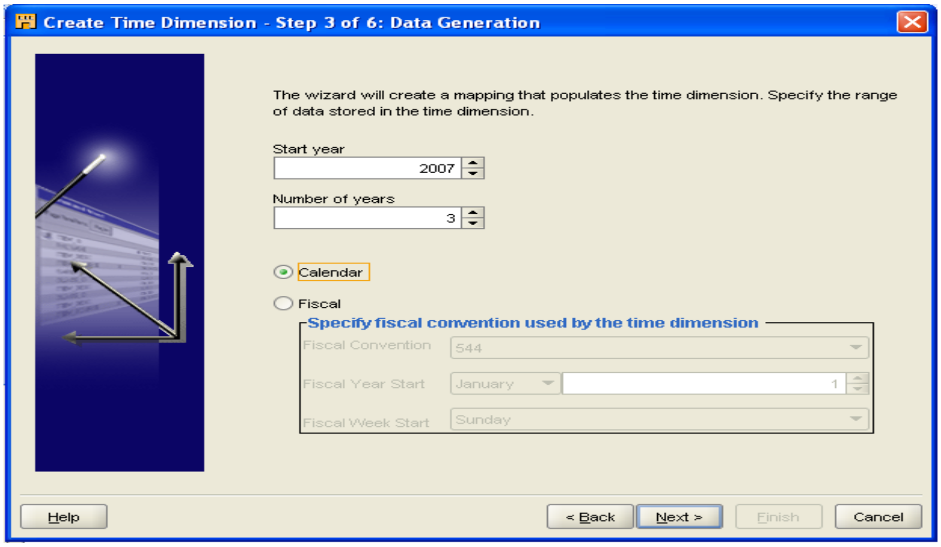
Create Time Dimension - Step 2 of 6: Storage

The storage type for a time dimension is driven by the warehouse implementation. Choose ROLAP if you have high volumes of data combined with high refresh rates or if you have detailed high volume data. Choose MOLAP to store aggregated data that is used for analysis.

Specify data storage for the time dimension:

● ROLAP: Relational storage
○ MOLAP: Multidimensional storage

4.Keep the default selection and select NEXT



Create Time Dimension - Step 3 of 6: Data Generation

The wizard will create a mapping that populates the time dimension. Specify the range of data stored in the time dimension.

Start year: 2007

Number of years: 3

● Calendar
○ Fiscal

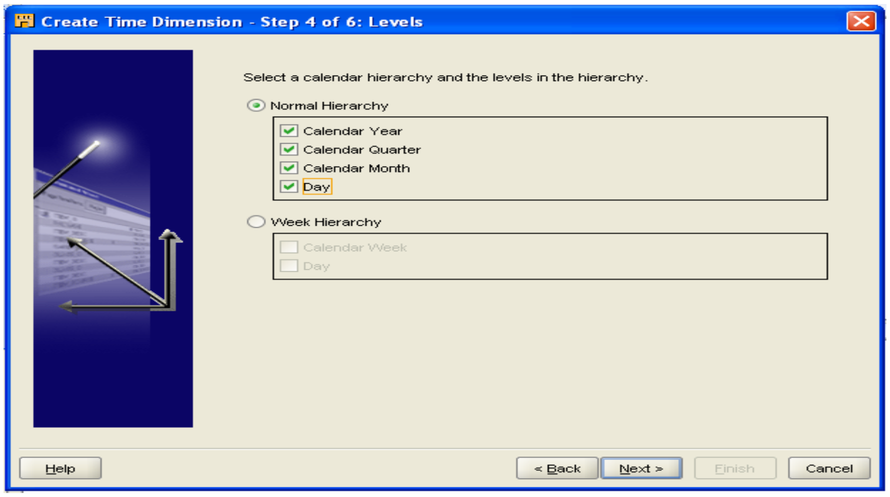
Specify fiscal convention used by the time dimension

Fiscal Convention: 544

Fiscal Year Start: January 1

Fiscal Week Start: Sunday

5.All the Checkbox will appear as uncheck. Select all the check boxes of Normal Hierarchy



Create Time Dimension - Step 4 of 6: Levels

Select a calendar hierarchy and the levels in the hierarchy.

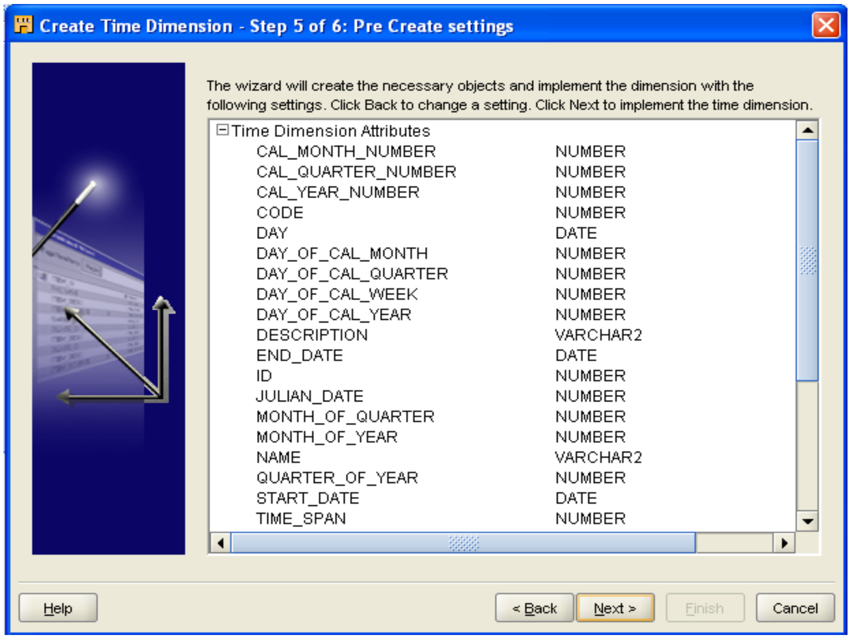
● Normal Hierarchy

- ☑ Calendar Year
- ☑ Calendar Quarter
- ☑ Calendar Month
- ☑ Day

○ Week Hierarchy

- □ Calendar Week
- □ Day

6.Summary screen will appear. Select NEXT.

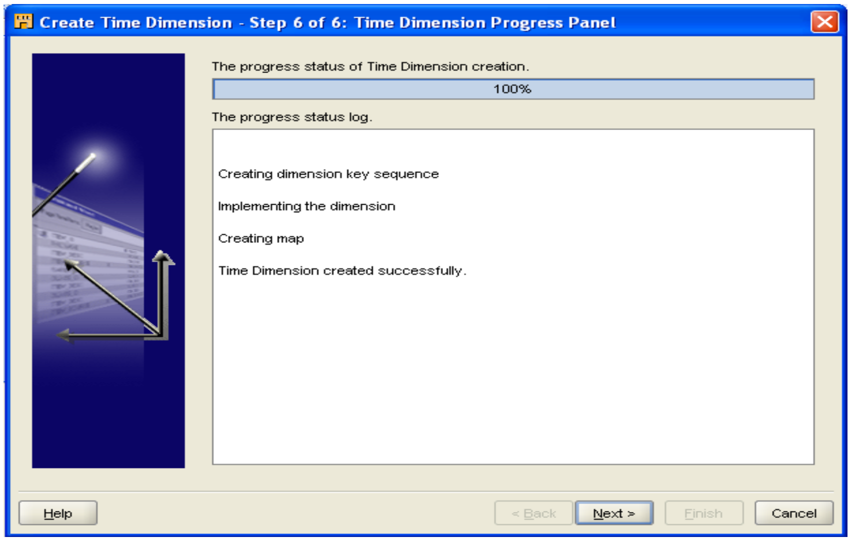


Create Time Dimension - Step 5 of 6: Pre Create settings

The wizard will create the necessary objects and implement the dimension with the following settings. Click Back to change a setting. Click Next to implement the time dimension.

```
⊟Time Dimension Attributes
      CAL_MONTH_NUMBER          NUMBER
      CAL_QUARTER_NUMBER        NUMBER
      CAL_YEAR_NUMBER           NUMBER
      CODE                      NUMBER
      DAY                       DATE
      DAY_OF_CAL_MONTH          NUMBER
      DAY_OF_CAL_QUARTER        NUMBER
      DAY_OF_CAL_WEEK           NUMBER
      DAY_OF_CAL_YEAR           NUMBER
      DESCRIPTION               VARCHAR2
      END_DATE                  DATE
      ID                        NUMBER
      JULIAN_DATE               NUMBER
      MONTH_OF_QUARTER          NUMBER
      MONTH_OF_YEAR             NUMBER
      NAME                      VARCHAR2
      QUARTER_OF_YEAR           NUMBER
      START_DATE                DATE
      TIME_SPAN                 NUMBER
```

7.Click NEXT


Create Time Dimension - Step 6 of 6: Time Dimension Progress Panel
100%
Help
< Back
Next >
Finish
Cancel

The progress status of Time Dimension creation.

The progress status log.

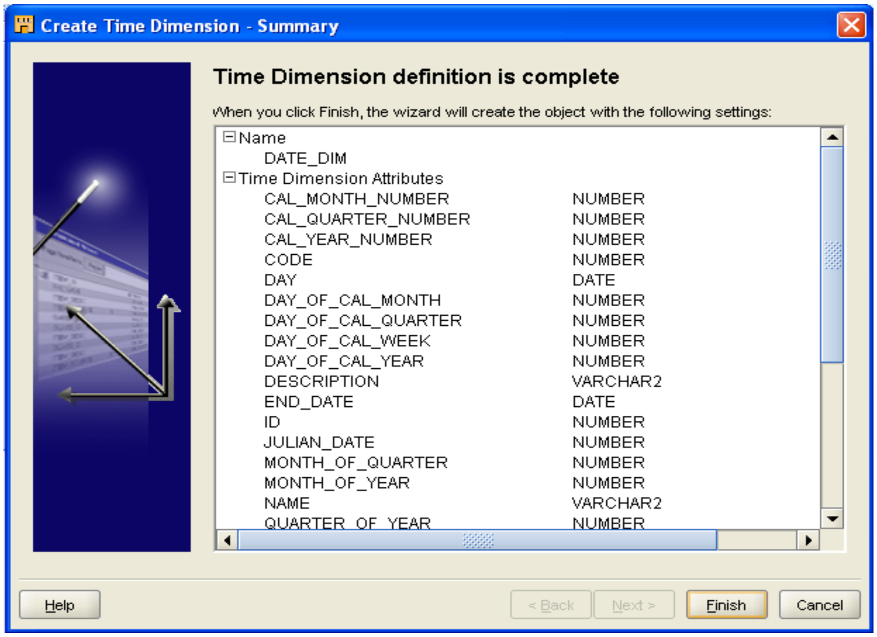
Creating dimension key sequence

Implementing the dimension

Creating map

Time Dimension created successfully.

8.Click on FINISH



Time Dimension definition is complete

When you click Finish, the wizard will create the object with the following settings:

⊟Name

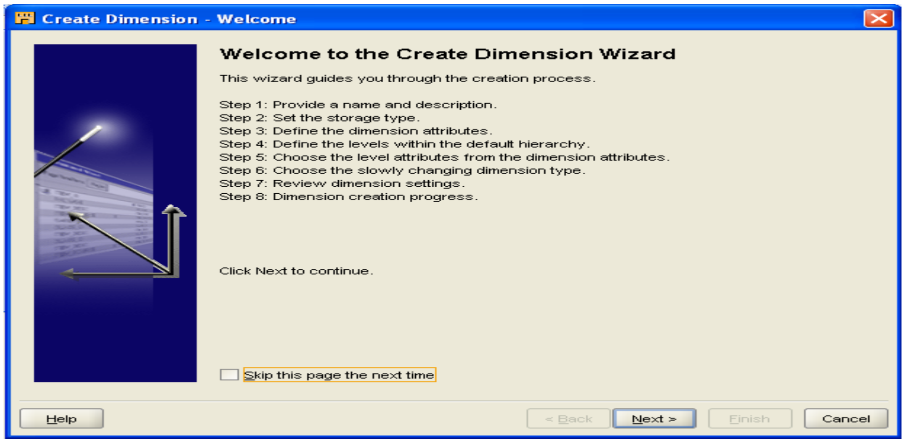
DATE_DIM

⊟Time Dimension Attributes

CAL_MONTH_NUMBER	NUMBER
CAL_QUARTER_NUMBER	NUMBER
CAL_YEAR_NUMBER	NUMBER
CODE	NUMBER
DAY	DATE
DAY_OF_CAL_MONTH	NUMBER
DAY_OF_CAL_QUARTER	NUMBER
DAY_OF_CAL_WEEK	NUMBER
DAY_OF_CAL_YEAR	NUMBER
DESCRIPTION	VARCHAR2
END_DATE	DATE
ID	NUMBER
JULIAN_DATE	NUMBER
MONTH_OF_QUARTER	NUMBER
MONTH_OF_YEAR	NUMBER
NAME	VARCHAR2
QUARTER_OF_YEAR	NUMBER

Product Dimension:

1.In **Project Explorer**| **ACME_DW_PROJECT**|**Databases** Node| **Oracle| ACME_DWH** then right click on the Dimensions Node|**New**|**Using Wizard…** Then click on **NEXT**



Create Dimension - Welcome

Welcome to the Create Dimension Wizard

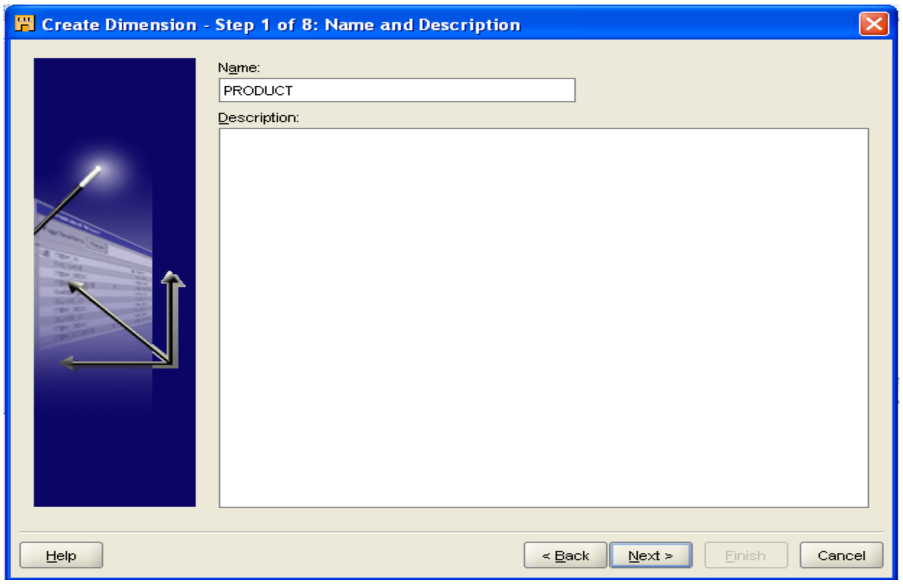
This wizard guides you through the creation process.

Step 1: Provide a name and description.
Step 2: Set the storage type.
Step 3: Define the dimension attributes.
Step 4: Define the levels within the default hierarchy.
Step 5: Choose the level attributes from the dimension attributes.
Step 6: Choose the slowly changing dimension type.
Step 7: Review dimension settings.
Step 8: Dimension creation progress.

Click Next to continue.

□ Skip this page the next time

2.Give Name as Product. Then click NEXT.

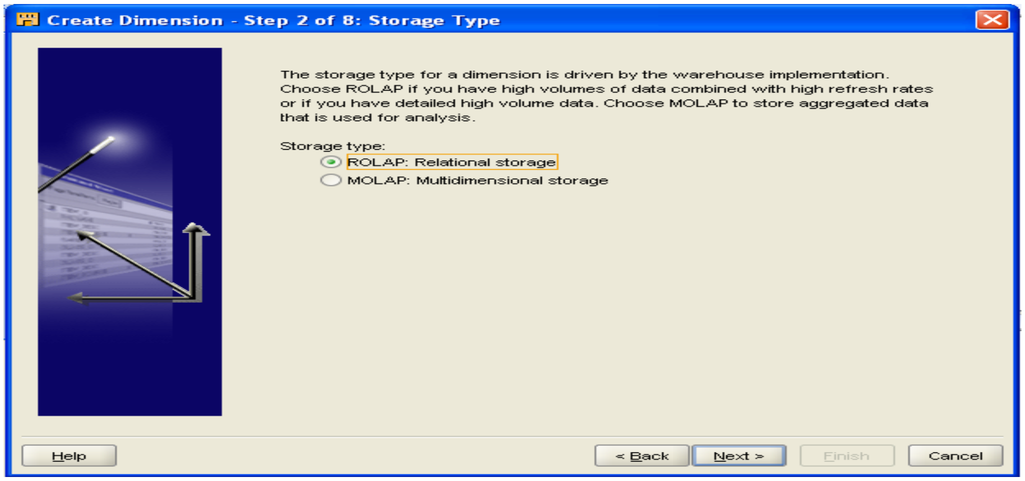


Create Dimension - Step 1 of 8: Name and Description

Name: PRODUCT

Description:

3.Select ROLAP as a storage type and click NEXT


Create Dimension - Step 2 of 8: Storage Type

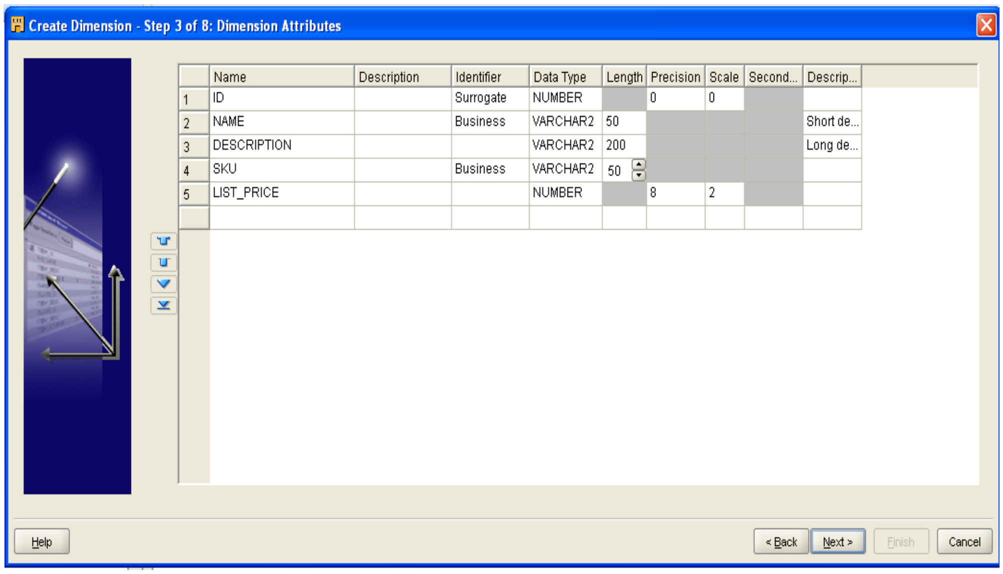
The storage type for a dimension is driven by the warehouse implementation. Choose ROLAP if you have high volumes of data combined with high refresh rates or if you have detailed high volume data. Choose MOLAP to store aggregated data that is used for analysis.

Storage type:

● ROLAP: Relational storage

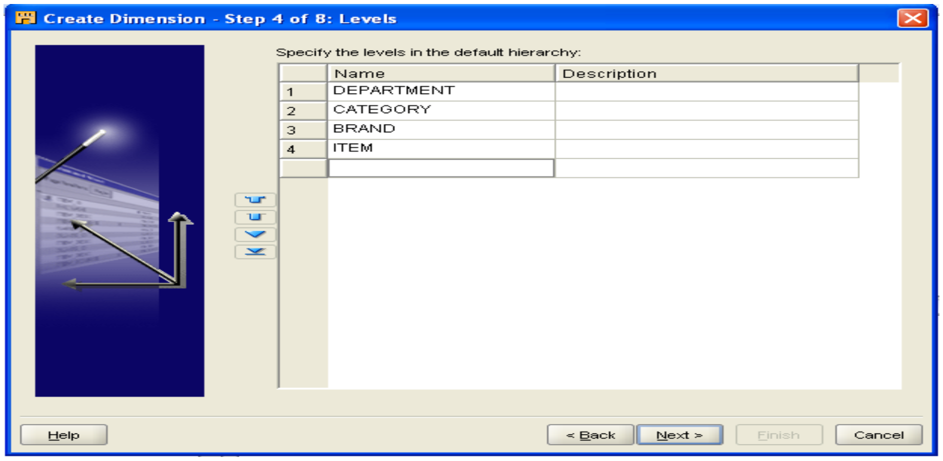
○ MOLAP: Multidimensional storage

4.Add Row SKU with its attributes such as identifier : Business, Data type : VARCHAR2,Length : 50 and LIST_PRICE with its attributes such as Data type : NUMBER, Precision : 8,Scale : 2 and click NEXT.


Create Dimension - Step 3 of 8: Dimension Attributes

	Name	Description	Identifier	Data Type	Length	Precision	Scale	Second...	Descrip...
1	ID		Surrogate	NUMBER		0	0		
2	NAME		Business	VARCHAR2	50				Short de...
3	DESCRIPTION			VARCHAR2	200				Long de...
4	SKU		Business	VARCHAR2	50				
5	LIST_PRICE			NUMBER		8	2		

5.Add row named as DEPARTMENT, CATEGORY, BRAND and ITEM and click NEXT.

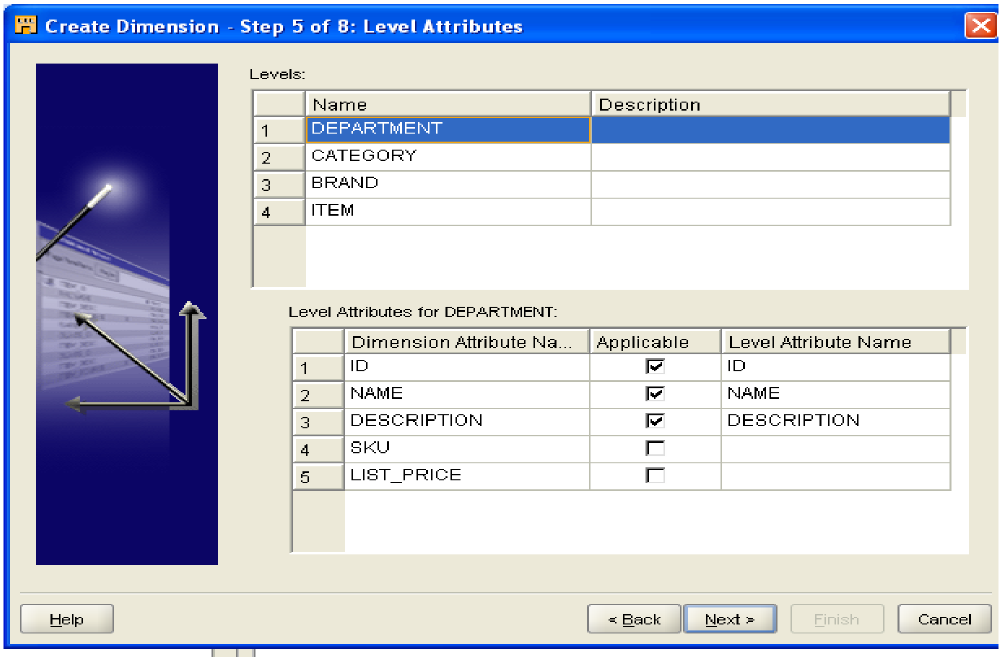


Create Dimension - Step 4 of 8: Levels

Specify the levels in the default hierarchy:

	Name	Description
1	DEPARTMENT	
2	CATEGORY	
3	BRAND	
4	ITEM	

6. Uncheck SKU and LIST_PRICE column of DEPARTMENT and click NEXT.



Create Dimension - Step 5 of 8: Level Attributes

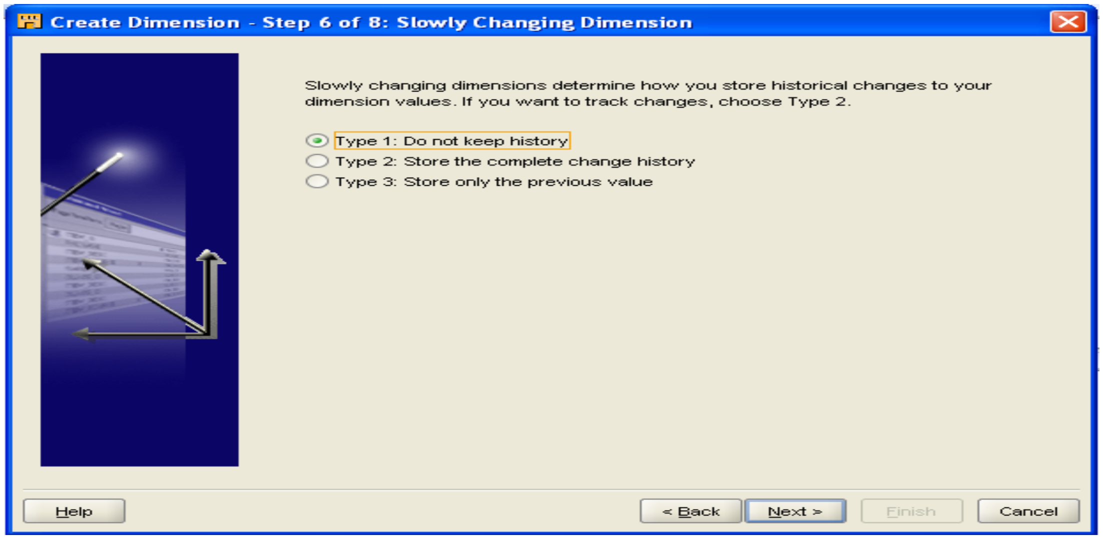
Levels:

	Name	Description
1	DEPARTMENT	
2	CATEGORY	
3	BRAND	
4	ITEM	

Level Attributes for DEPARTMENT:

	Dimension Attribute Na...	Applicable	Level Attribute Name
1	ID	☑	ID
2	NAME	☑	NAME
3	DESCRIPTION	☑	DESCRIPTION
4	SKU	□	
5	LIST_PRICE	□	

7. Choose “Type 1 : Do not keep history” and click NEXT.

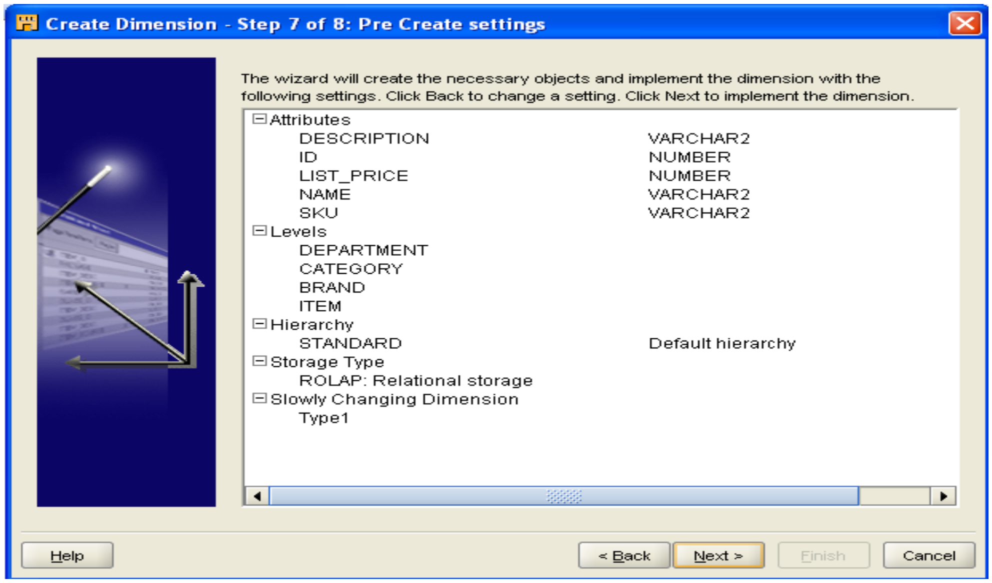


Create Dimension - Step 6 of 8: Slowly Changing Dimension

Slowly changing dimensions determine how you store historical changes to your dimension values. If you want to track changes, choose Type 2.

● Type 1: Do not keep history
○ Type 2: Store the complete change history
○ Type 3: Store only the previous value

8. This is a summary screen. Click NEXT

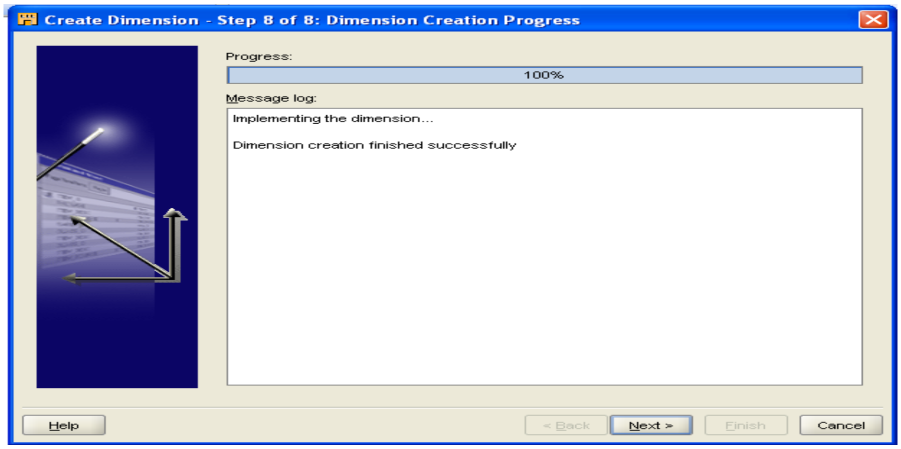


Create Dimension - Step 7 of 8: Pre Create settings

The wizard will create the necessary objects and implement the dimension with the following settings. Click Back to change a setting. Click Next to implement the dimension.

```
⊟Attributes
    DESCRIPTION                    VARCHAR2
    ID                             NUMBER
    LIST_PRICE                     NUMBER
    NAME                           VARCHAR2
    SKU                            VARCHAR2
⊟Levels
    DEPARTMENT
    CATEGORY
    BRAND
    ITEM
⊟Hierarchy
    STANDARD                       Default hierarchy
⊟Storage Type
    ROLAP: Relational storage
⊟Slowly Changing Dimension
    Type1
```

9.Click NEXT.



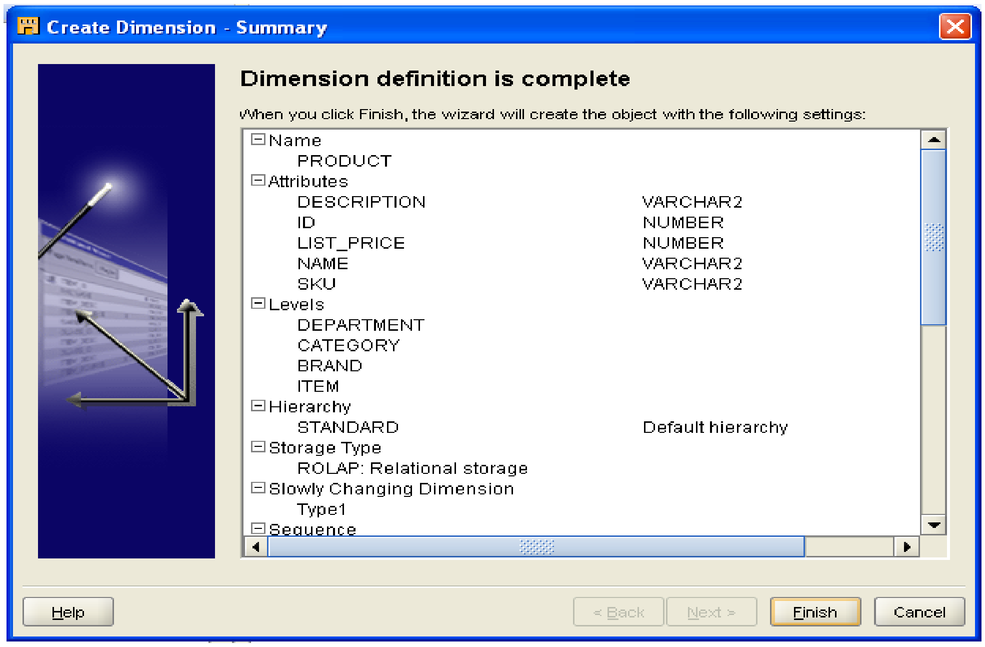
Create Dimension - Step 8 of 8: Dimension Creation Progress

Progress: 100%

Message log:

```
Implementing the dimension...

Dimension creation finished successfully
```

10. Click FINISH.



Create Dimension - Summary

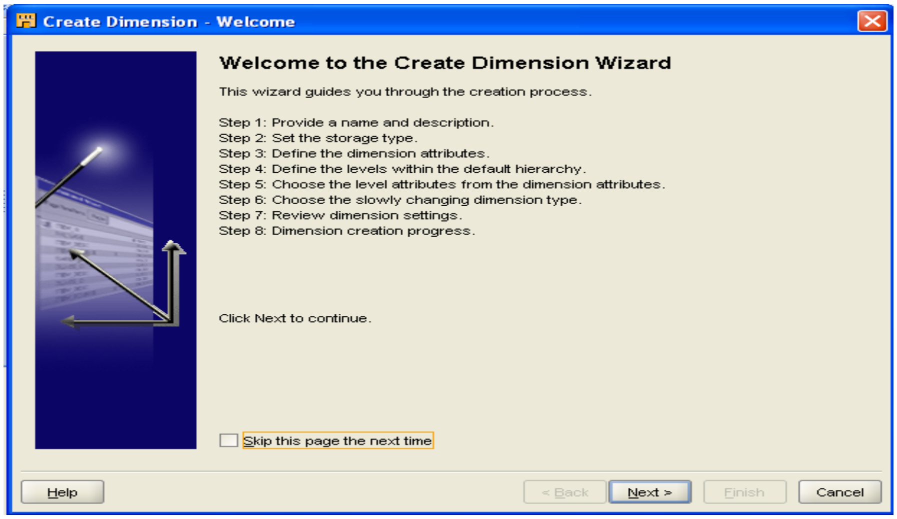
Dimension definition is complete

When you click Finish, the wizard will create the object with the following settings:

```
⊟Name
    PRODUCT
⊟Attributes
    DESCRIPTION          VARCHAR2
    ID                   NUMBER
    LIST_PRICE           NUMBER
    NAME                 VARCHAR2
    SKU                  VARCHAR2
⊟Levels
    DEPARTMENT
    CATEGORY
    BRAND
    ITEM
⊟Hierarchy
    STANDARD             Default hierarchy
⊟Storage Type
    ROLAP: Relational storage
⊟Slowly Changing Dimension
    Type1
⊟Sequence
```

STORE Dimension :

1.In Project Explorer| ACME_DW_PROJECT|Databases Node|Oracle| ACME_DWH then right click on the Dimensions Node|New|Using Wizard… Then click on NEXT



Create Dimension - Welcome

Welcome to the Create Dimension Wizard

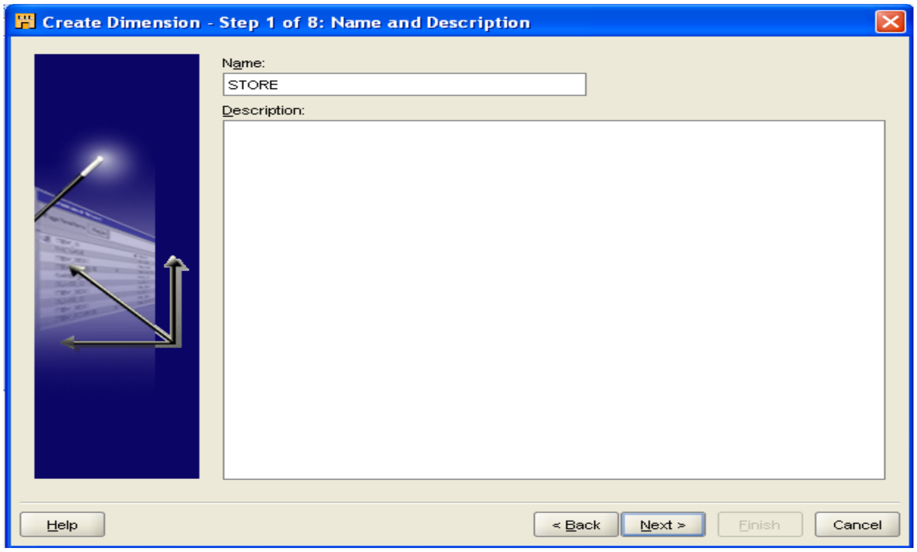
This wizard guides you through the creation process.

Step 1: Provide a name and description.
Step 2: Set the storage type.
Step 3: Define the dimension attributes.
Step 4: Define the levels within the default hierarchy.
Step 5: Choose the level attributes from the dimension attributes.
Step 6: Choose the slowly changing dimension type.
Step 7: Review dimension settings.
Step 8: Dimension creation progress.

Click Next to continue.

□ Skip this page the next time

2.Give the name as “Store” and click on NEXT.

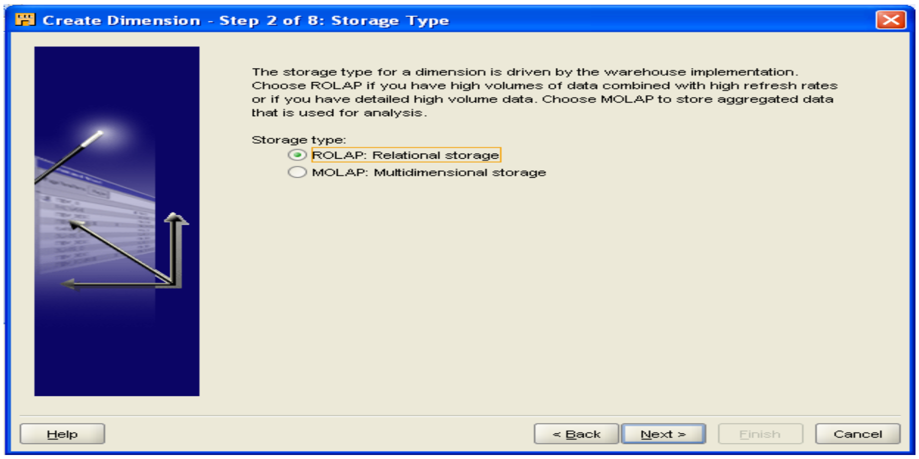


Create Dimension - Step 1 of 8: Name and Description

Name: STORE

Description:

3.Select storage option as ROLAP and click NEXT



Create Dimension - Step 2 of 8: Storage Type

The storage type for a dimension is driven by the warehouse implementation. Choose ROLAP if you have high volumes of data combined with high refresh rates or if you have detailed high volume data. Choose MOLAP to store aggregated data that is used for analysis.

Storage type:

● ROLAP: Relational storage

○ MOLAP: Multidimensional storage

4.Add Store_Number whose Identifier : Business, Data Type: VARCHAR2, Length :10, Address1 whose Data Type: VARCHAR2, Length :60,

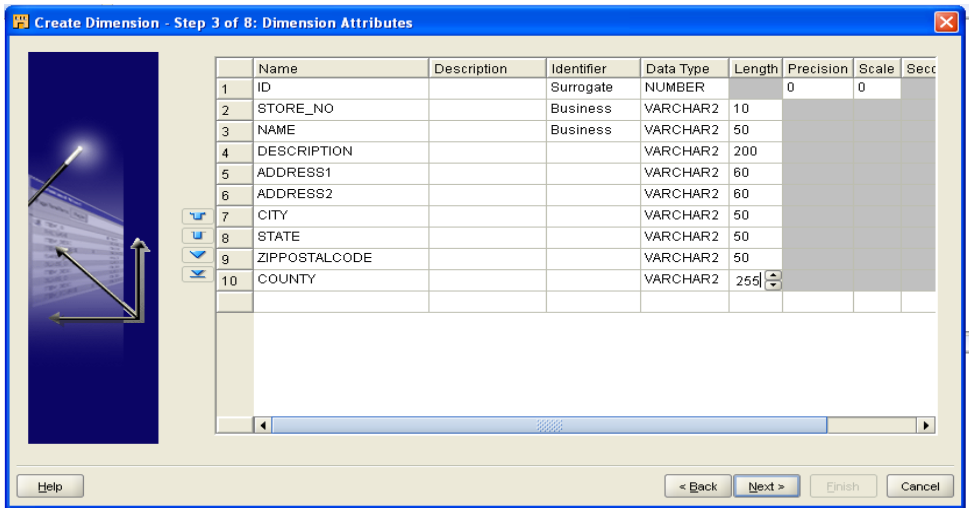
Address2 whose Data Type: VARCHAR2, Length :60,

City whose Data Type: VARCHAR2, Length :50,

State whose Data Type: VARCHAR2, Length :50,

ZipPostalCode whose Data Type: VARCHAR2, Length :50,

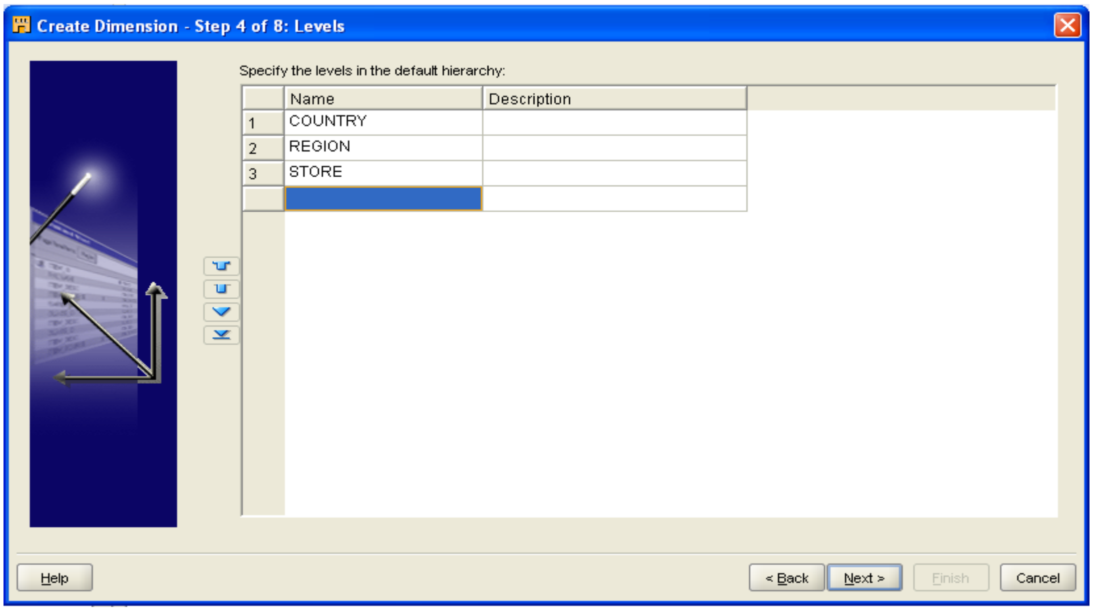
County whose Data Type: VARCHAR2, Length :255.



Create Dimension - Step 3 of 8: Dimension Attributes

	Name	Description	Identifier	Data Type	Length	Precision	Scale	Secc
1	ID		Surrogate	NUMBER		0	0	
2	STORE_NO		Business	VARCHAR2	10			
3	NAME		Business	VARCHAR2	50			
4	DESCRIPTION			VARCHAR2	200			
5	ADDRESS1			VARCHAR2	60			
6	ADDRESS2			VARCHAR2	60			
7	CITY			VARCHAR2	50			
8	STATE			VARCHAR2	50			
9	ZIPPOSTALCODE			VARCHAR2	50			
10	COUNTY			VARCHAR2	255			

5.Add 3 fields named as COUNTRY, REGION and STORE. And click NEXT.

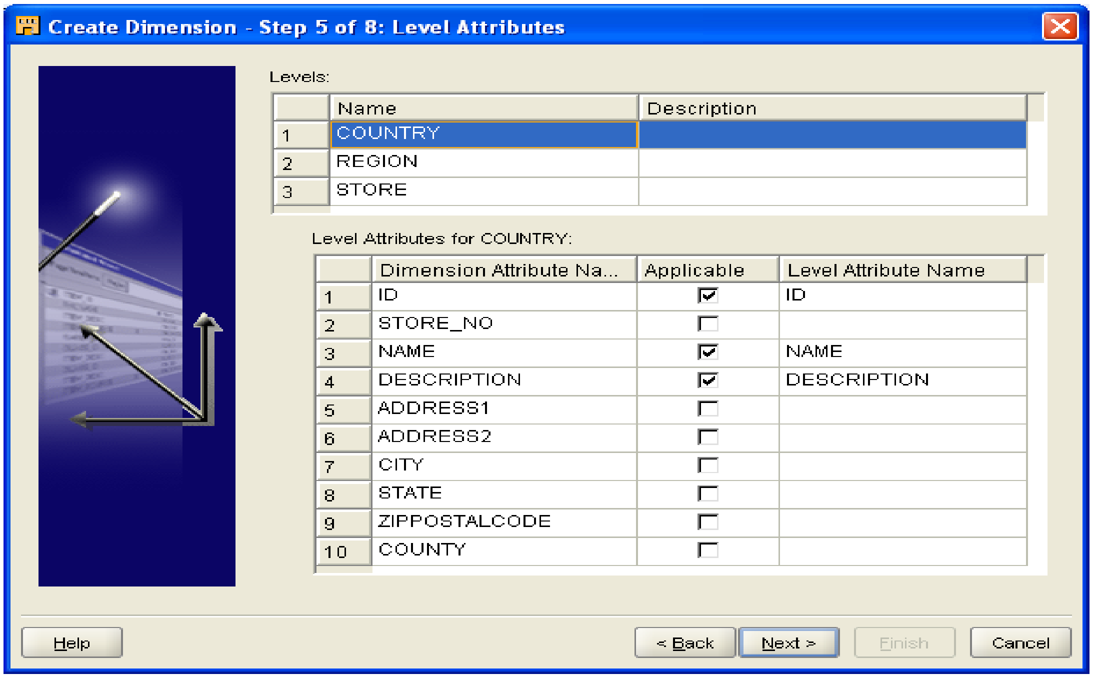


Create Dimension - Step 4 of 8: Levels

Specify the levels in the default hierarchy:

	Name	Description
1	COUNTRY	
2	REGION	
3	STORE	

6. Uncheck the Description check box for Store level.



Create Dimension - Step 5 of 8: Level Attributes

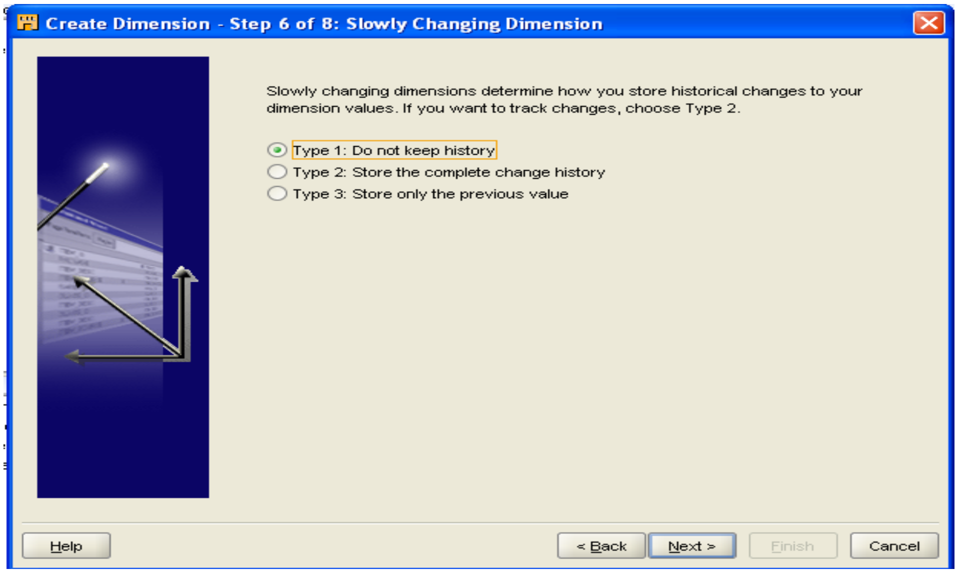
Levels:

	Name	Description
1	COUNTRY	
2	REGION	
3	STORE	

Level Attributes for COUNTRY:

	Dimension Attribute Na...	Applicable	Level Attribute Name
1	ID	☑	ID
2	STORE_NO	□	
3	NAME	☑	NAME
4	DESCRIPTION	☑	DESCRIPTION
5	ADDRESS1	□	
6	ADDRESS2	□	
7	CITY	□	
8	STATE	□	
9	ZIPPOSTALCODE	□	
10	COUNTY	□	

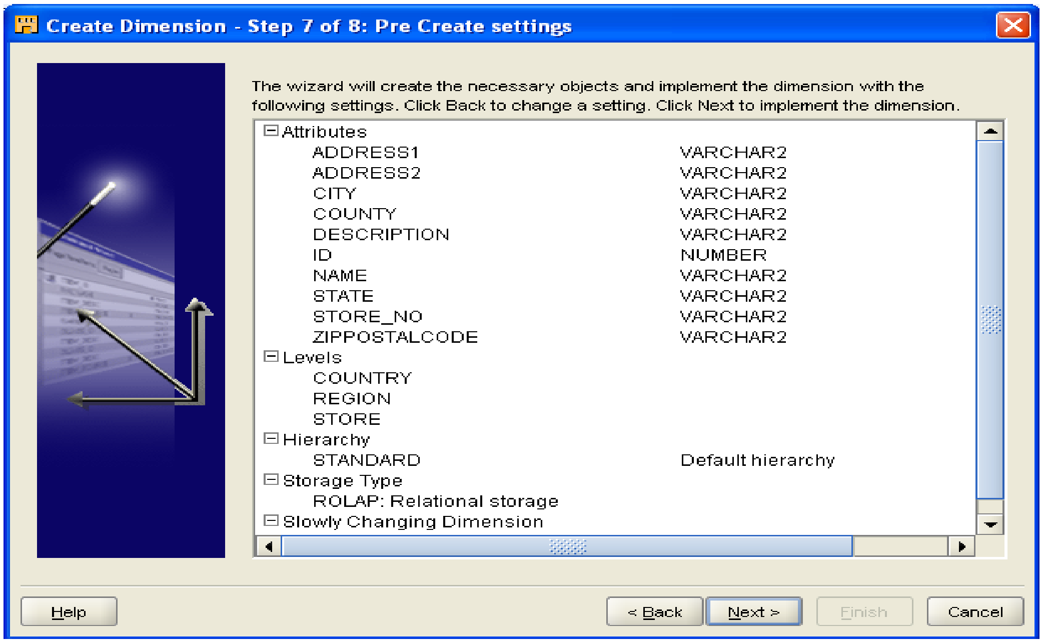
7. Select "Type 1 : Do not keep history" and select NEXT.



Slowly changing dimensions determine how you store historical changes to your dimension values. If you want to track changes, choose Type 2.

● Type 1: Do not keep history
○ Type 2: Store the complete change history
○ Type 3: Store only the previous value

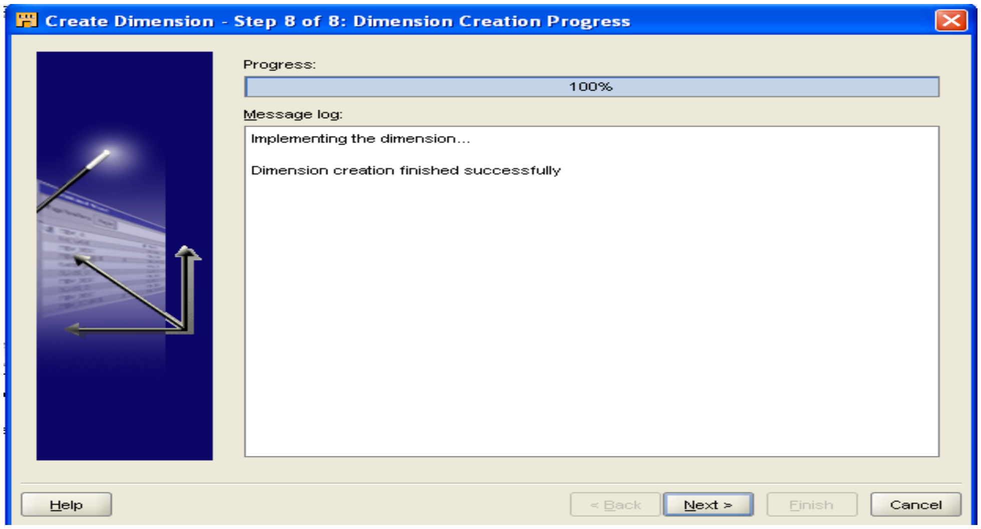
8. The summary screen will appear. Click NEXT



The wizard will create the necessary objects and implement the dimension with the following settings. Click Back to change a setting. Click Next to implement the dimension.

```
⊟ Attributes
      ADDRESS1                     VARCHAR2
      ADDRESS2                     VARCHAR2
      CITY                         VARCHAR2
      COUNTY                       VARCHAR2
      DESCRIPTION                  VARCHAR2
      ID                           NUMBER
      NAME                         VARCHAR2
      STATE                        VARCHAR2
      STORE_NO                     VARCHAR2
      ZIPPOSTALCODE                VARCHAR2
⊟ Levels
      COUNTRY
      REGION
      STORE
⊟ Hierarchy
      STANDARD                     Default hierarchy
⊟ Storage Type
      ROLAP: Relational storage
⊟ Slowly Changing Dimension
```

9. Click NEXT.



Progress:

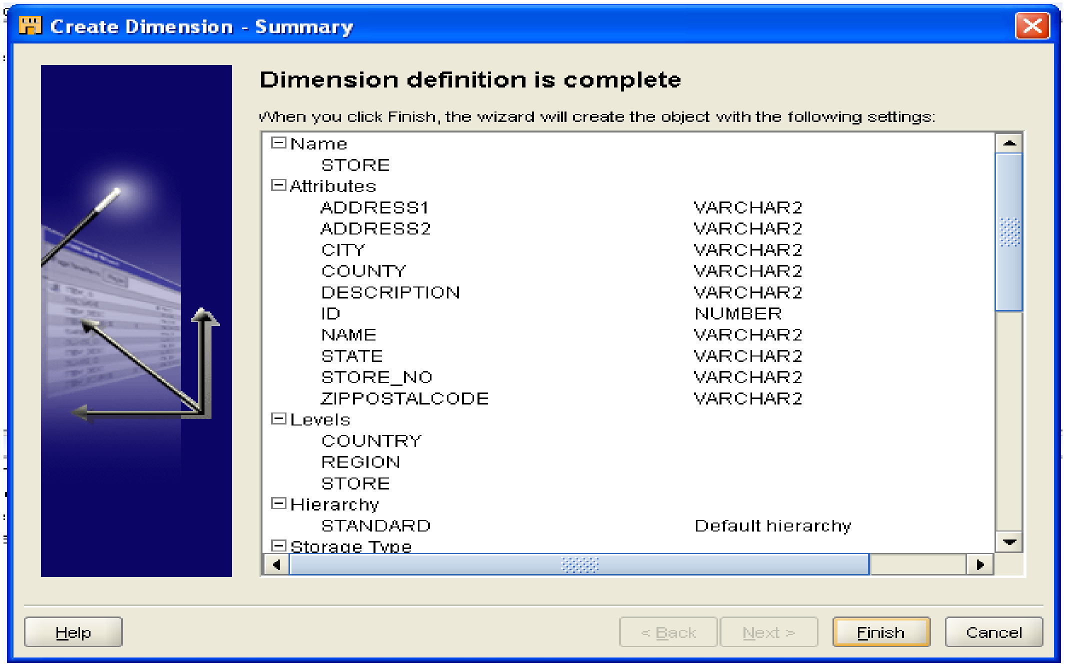
100%

Message log:

Implementing the dimension...

Dimension creation finished successfully

10. Select FINISH.

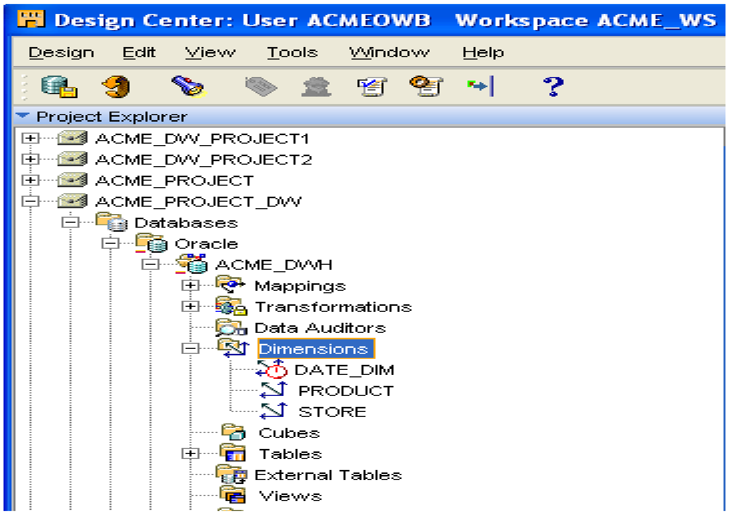


Dimension definition is complete

When you click Finish, the wizard will create the object with the following settings:

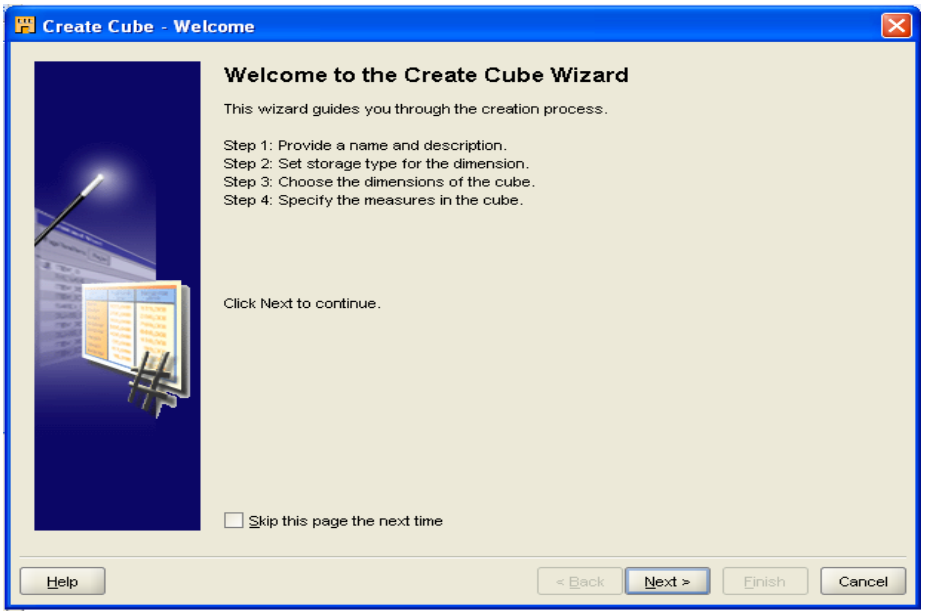
```
⊟Name
    STORE
⊟Attributes
    ADDRESS1                VARCHAR2
    ADDRESS2                VARCHAR2
    CITY                    VARCHAR2
    COUNTY                  VARCHAR2
    DESCRIPTION             VARCHAR2
    ID                      NUMBER
    NAME                    VARCHAR2
    STATE                   VARCHAR2
    STORE_NO                VARCHAR2
    ZIPPOSTALCODE           VARCHAR2
⊟Levels
    COUNTRY
    REGION
    STORE
⊟Hierarchy
    STANDARD                Default hierarchy
⊟Storage Type
```

11.The DATE_DIM,PRODUCT,STORE are created



Cube

1.In **Project Explorer**| **ACME_DW_PROJECT**|**Databases** Node| **Oracle**|**ACME_DWH**|**CUBES** then right click on the **CUBE** Node|**New**|**Using Wizard…** Then click on **NEXT**



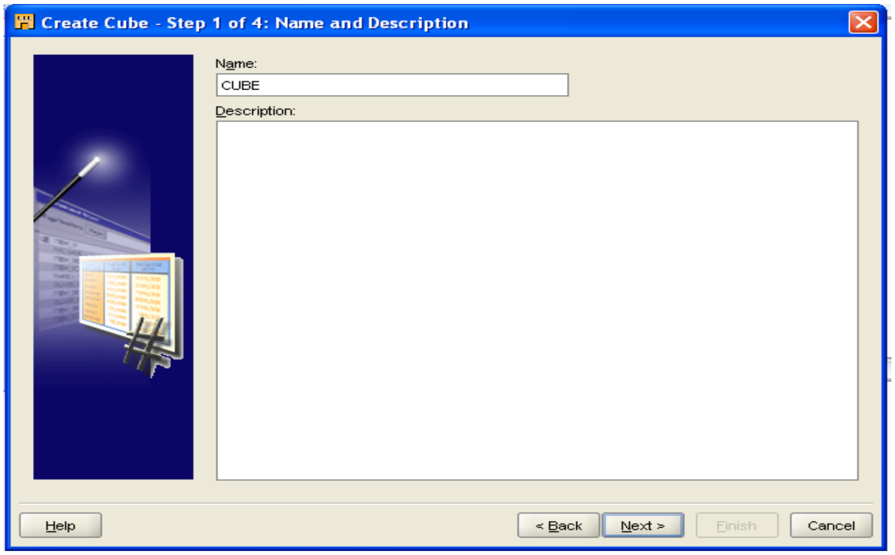
Welcome to the Create Cube Wizard

This wizard guides you through the creation process.

Step 1: Provide a name and description.
Step 2: Set storage type for the dimension.
Step 3: Choose the dimensions of the cube.
Step 4: Specify the measures in the cube.

Click Next to continue.

2. Give the name as SALES and click NEXT.

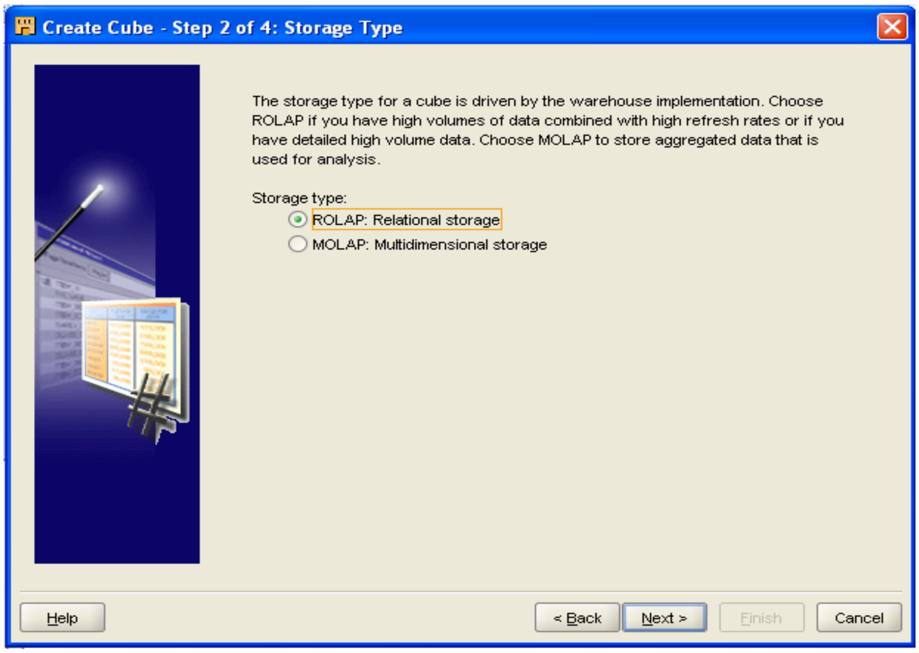


Create Cube - Step 1 of 4: Name and Description

Name: CUBE

Description:

3. Select storage as ROLAP and click NEXT



Create Cube - Step 2 of 4: Storage Type

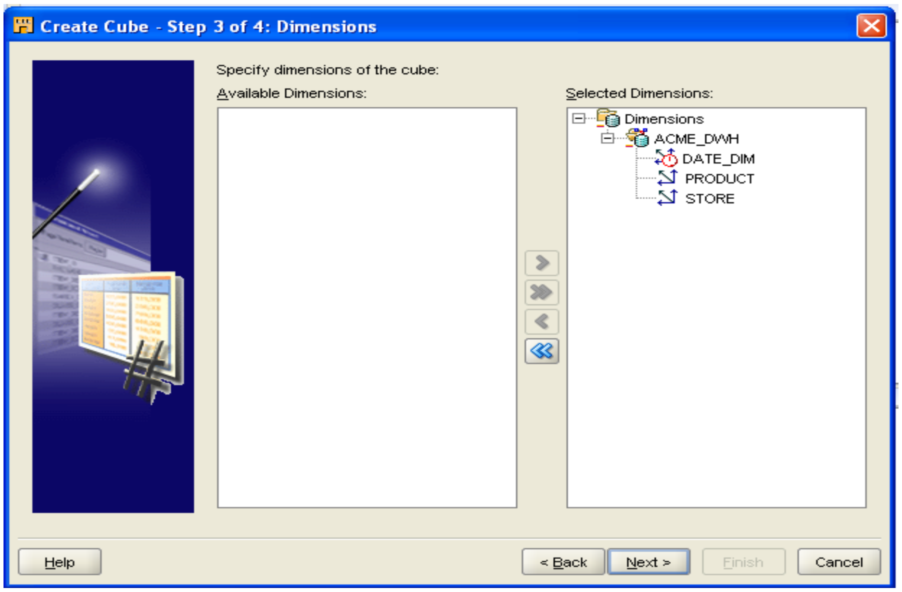
The storage type for a cube is driven by the warehouse implementation. Choose ROLAP if you have high volumes of data combined with high refresh rates or if you have detailed high volume data. Choose MOLAP to store aggregated data that is used for analysis.

Storage type:

● ROLAP: Relational storage

○ MOLAP: Multidimensional storage

4. Choose the dimension and then click on the double arrow in the center to move all the dimensions and select them.



Create Cube - Step 3 of 4: Dimensions

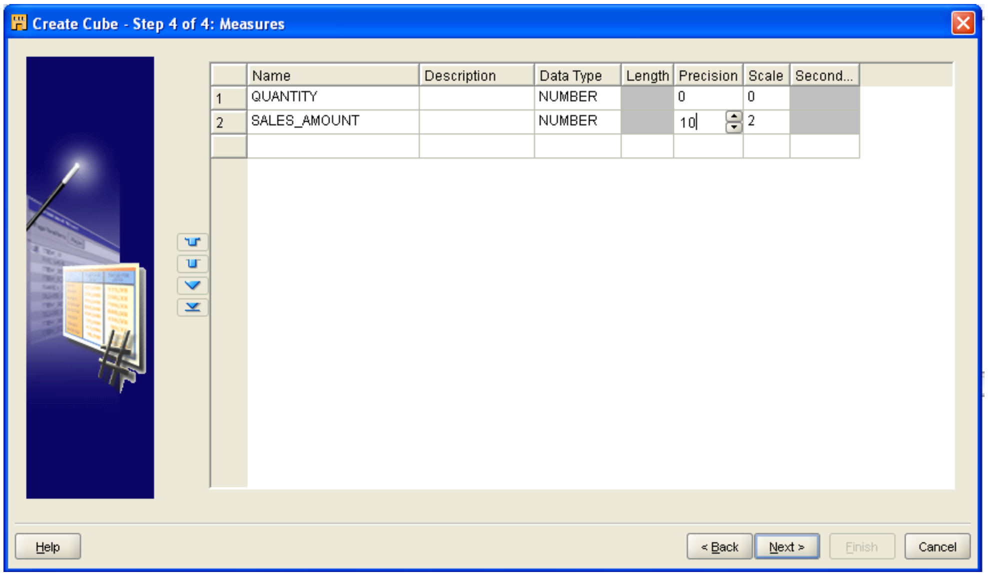
Specify dimensions of the cube:

Available Dimensions:

Selected Dimensions:

- Dimensions
  - ACME_DWH
    - DATE_DIM
    - PRODUCT
    - STORE

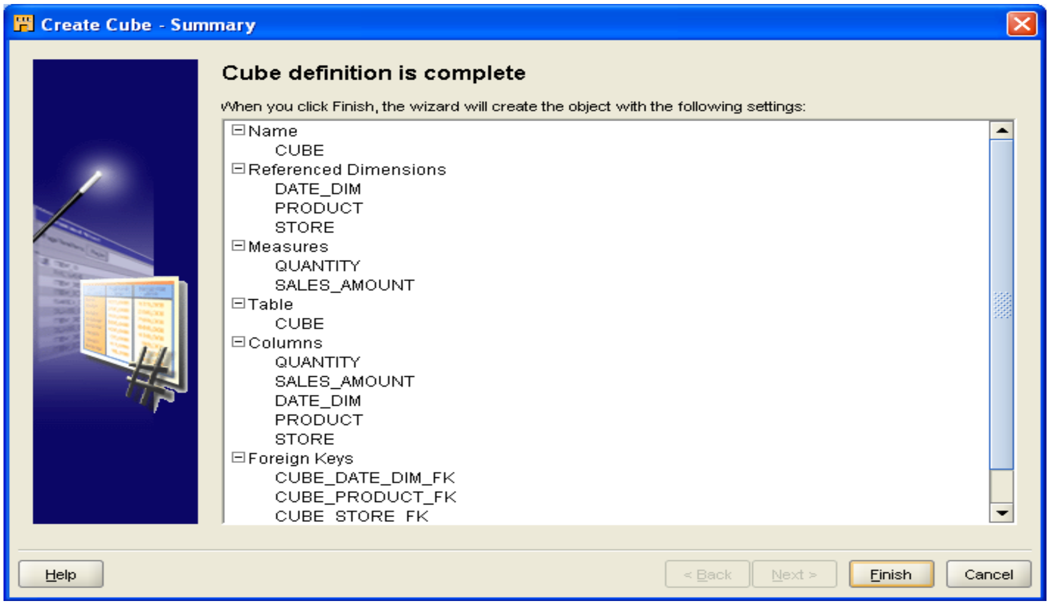
5. Enter the measures QUANTITY for first measure whose Data Type is Number, Precision is 0 and Scale is 0.Enter the SALES_AMOUNT for second measure whose Data Type is Number, Precision is 10 and Scale is 2.Click NEXT.



Create Cube - Step 4 of 4: Measures

	Name	Description	Data Type	Length	Precision	Scale	Second...
1	QUANTITY		NUMBER		0	0	
2	SALES_AMOUNT		NUMBER		10	2	

6. Click FINISH on Summary screen appear.



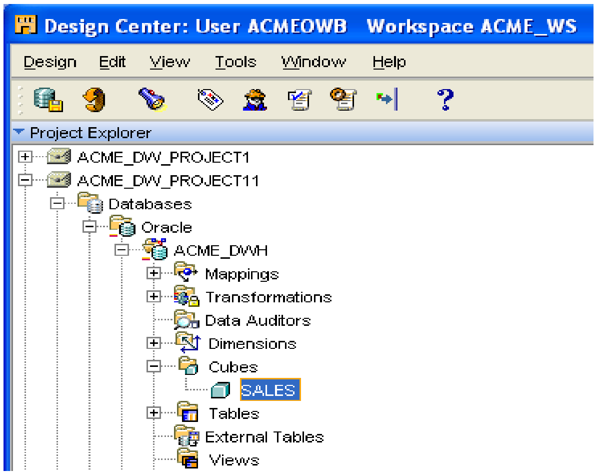
Create Cube - Summary

Cube definition is complete

When you click Finish, the wizard will create the object with the following settings:

```
⊟Name
    CUBE
⊟Referenced Dimensions
    DATE_DIM
    PRODUCT
    STORE
⊟Measures
    QUANTITY
    SALES_AMOUNT
⊟Table
    CUBE
⊟Columns
    QUANTITY
    SALES_AMOUNT
    DATE_DIM
    PRODUCT
    STORE
⊟Foreign Keys
    CUBE_DATE_DIM_FK
    CUBE_PRODUCT_FK
    CUBE_STORE_FK
```

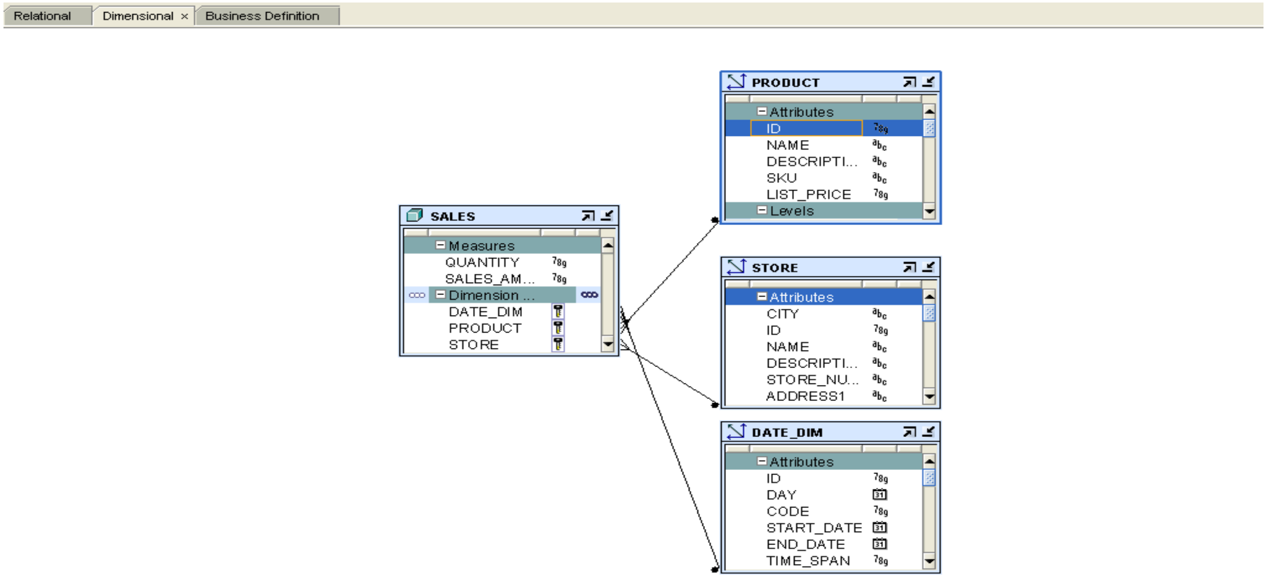
7.Sales will be created



Design Center: User ACMEOWB Workspace ACME_WS

```
⊞ ACME_DW_PROJECT1
⊟ ACME_DW_PROJECT11
    ⊟ Databases
        ⊟ Oracle
            ⊟ ACME_DWH
                ⊞ Mappings
                ⊞ Transformations
                  Data Auditors
                ⊞ Dimensions
                ⊟ Cubes
                    SALES
                ⊞ Tables
                  External Tables
                  Views
```

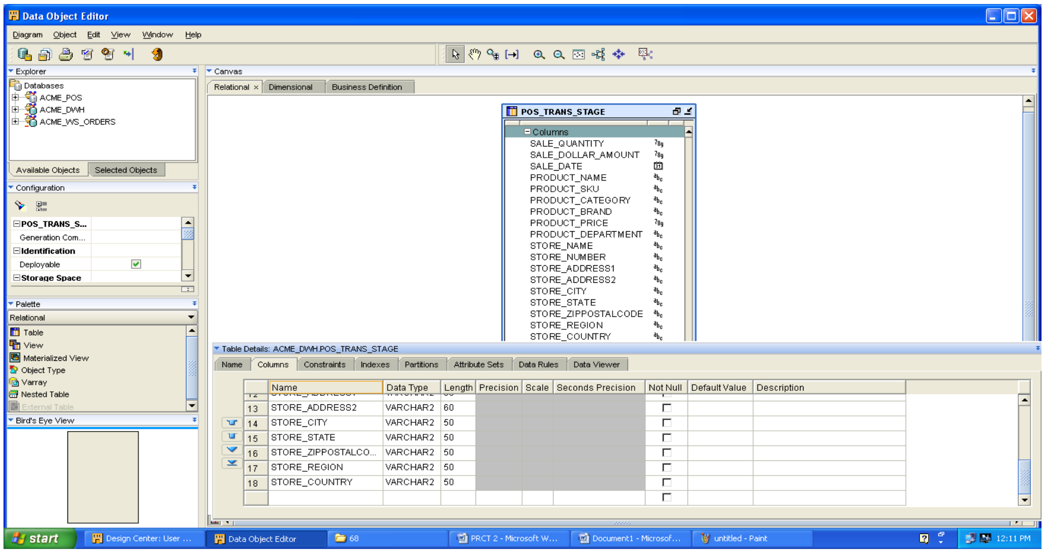
8.Sales will be displayed



PRACTICAL NO: 04

Aim: Design and built the ETL mapping.

1] **DATABASE|ORACLE|ACME_DWH** module, right click on the table node and select new. We will call our table **POS_TRANSE_STAGE** and enter the field in the column.



POS_TRANS_STAGE

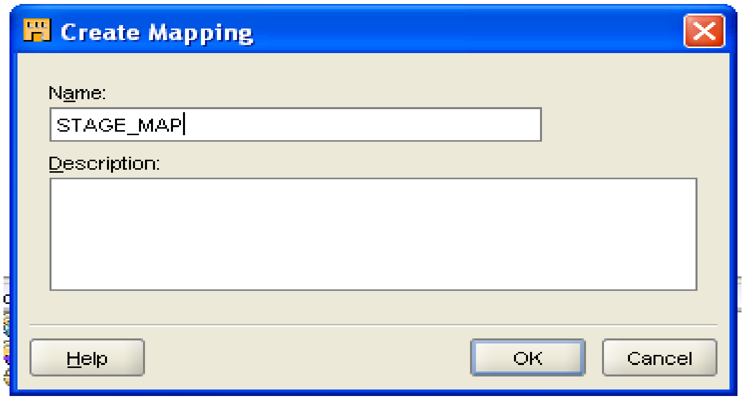
Columns
SALE_QUANTITY
SALE_DOLLAR_AMOUNT
SALE_DATE
PRODUCT_NAME
PRODUCT_SKU
PRODUCT_CATEGORY
PRODUCT_BRAND
PRODUCT_PRICE
PRODUCT_DEPARTMENT
STORE_NAME
STORE_NUMBER
STORE_ADDRESS1
STORE_ADDRESS2
STORE_CITY
STORE_STATE
STORE_ZIPPOSTALCODE
STORE_REGION
STORE_COUNTRY

Table Details: ACME_DWH.POS_TRANS_STAGE

Name | Columns | Constraints | Indexes | Partitions | Attribute Sets | Data Rules | Data Viewer

	Name	Data Type	Length	Precision	Scale	Seconds Precision	Not Null	Default Value	Description
13	STORE_ADDRESS2	VARCHAR2	60				□		
14	STORE_CITY	VARCHAR2	50				□		
15	STORE_STATE	VARCHAR2	50				□		
16	STORE_ZIPPOSTALCO...	VARCHAR2	50				□		
17	STORE_REGION	VARCHAR2	50				□		
18	STORE_COUNTRY	VARCHAR2	50				□		
							□		

2] **ACME_DW_PROJECT|DATABASE|ORACLE|ACME_DWH|MAPPINGS** and right click on it and select new. We will name this mapping **STAGE_MAP.**



Name: STAGE_MAP

Description:

3] **DATABASES|NON-ORACLE|ODBC|ACME_POS** node. Drag **POS_TRANSACTION** in the mapping window.

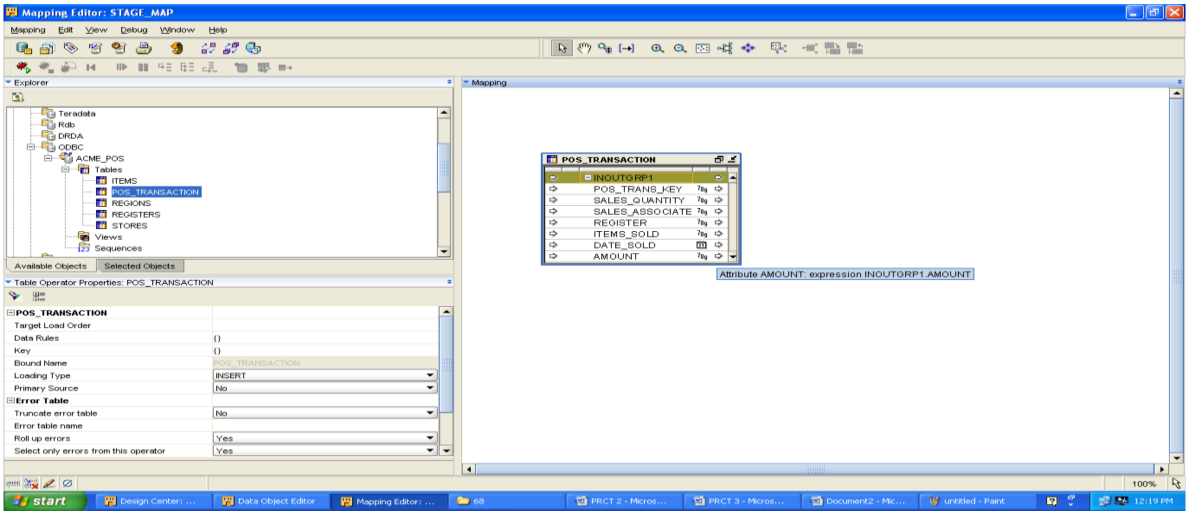


Table Operator Properties: POS_TRANSACTION

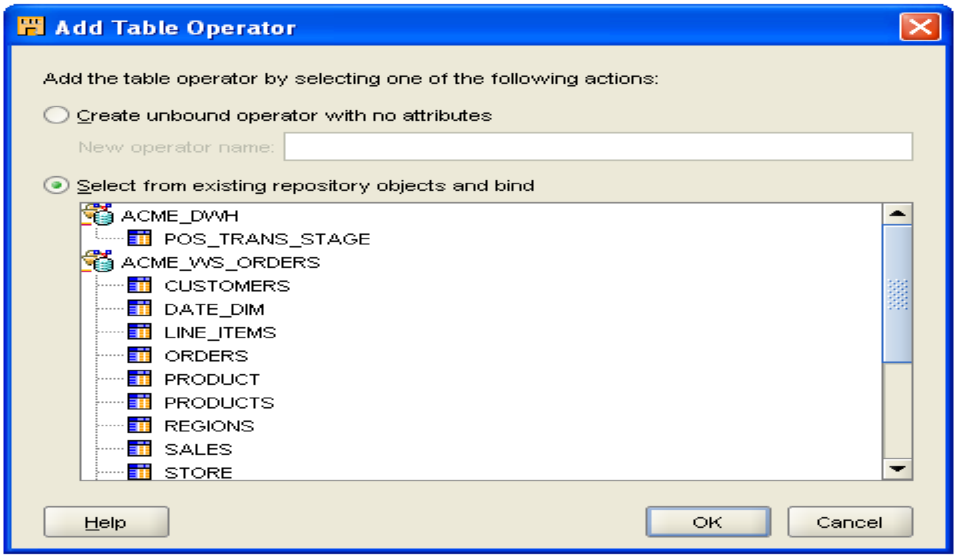
POS_TRANSACTION	
Target Load Order	
Data Rules	{}
Key	{}
Bound Name	POS_TRANSACTION
Loading Type	INSERT
Primary Source	No
Error Table	
Truncate error table	No
Error table name	
Roll up errors	Yes
Select only errors from this operator	Yes

POS_TRANSACTION

- INOUTGRP1
  - POS_TRANS_KEY
  - SALES_QUANTITY
  - SALES_ASSOCIATE
  - REGISTER
  - ITEMS_SOLD
  - DATE_SOLD
  - AMOUNT

Attribute AMOUNT: expression INOUTGRP1.AMOUNT

4] From the palette drag the table operator to the mapping window.



Add Table Operator

Add the table operator by selecting one of the following actions:

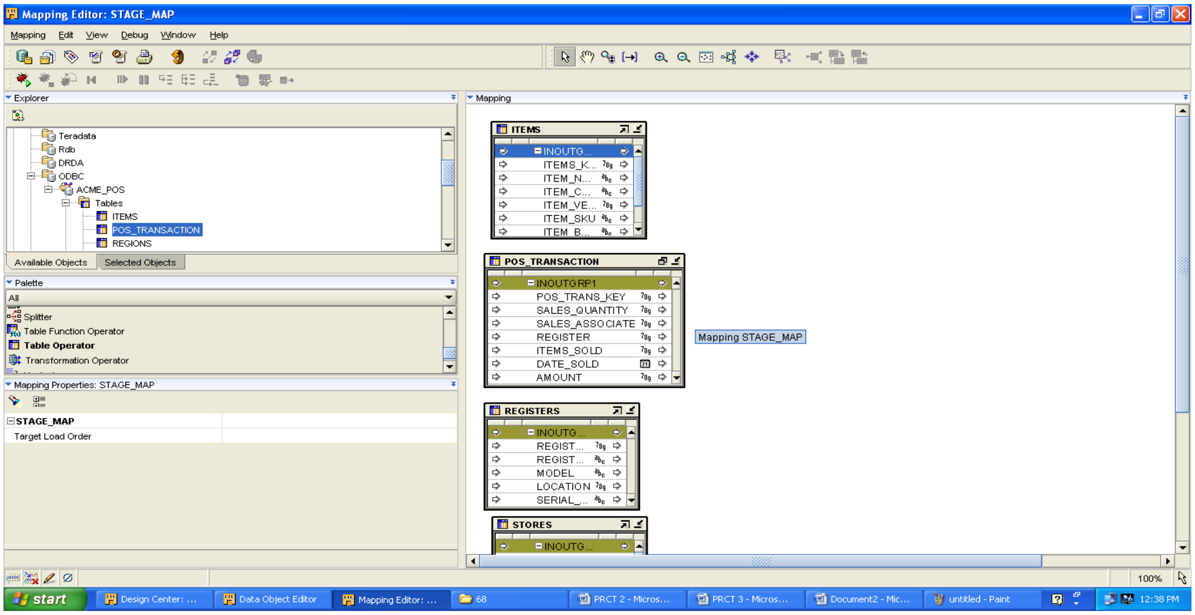
○ Create unbound operator with no attributes

New operator name:

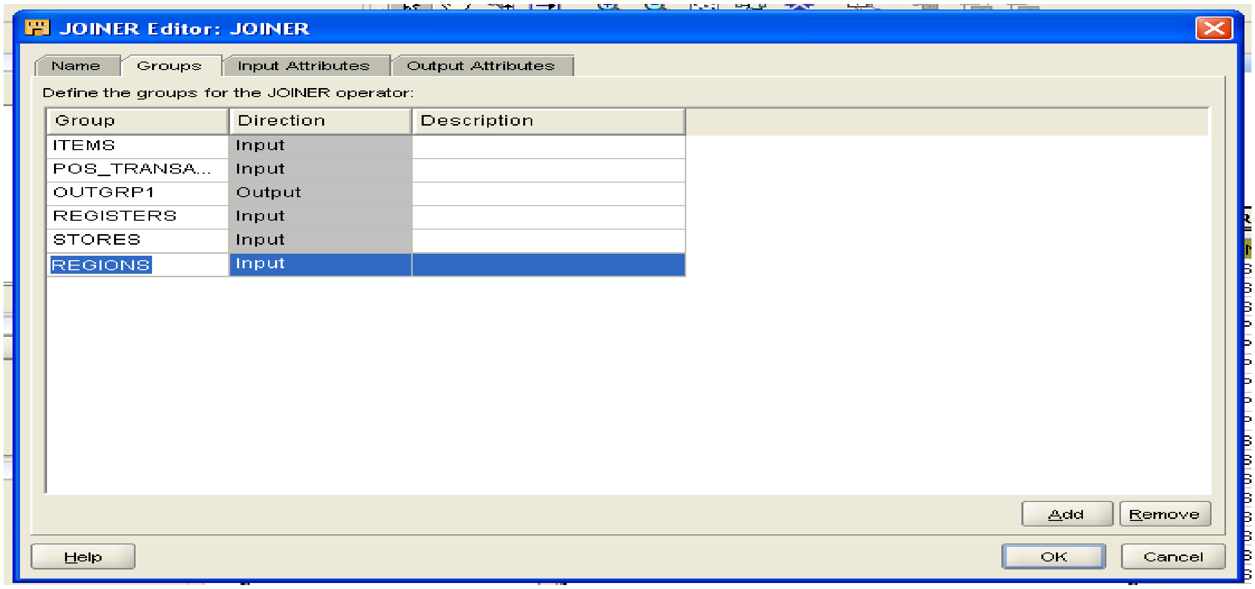
● Select from existing repository objects and bind

- ACME_DWH
  - POS_TRANS_STAGE
- ACME_WS_ORDERS
  - CUSTOMERS
  - DATE_DIM
  - LINE_ITEMS
  - ORDERS
  - PRODUCT
  - PRODUCTS
  - REGIONS
  - SALES
  - STORE

5] Drag **REGISTER, STORE, REGION, ITEM** tables in the mapping window.

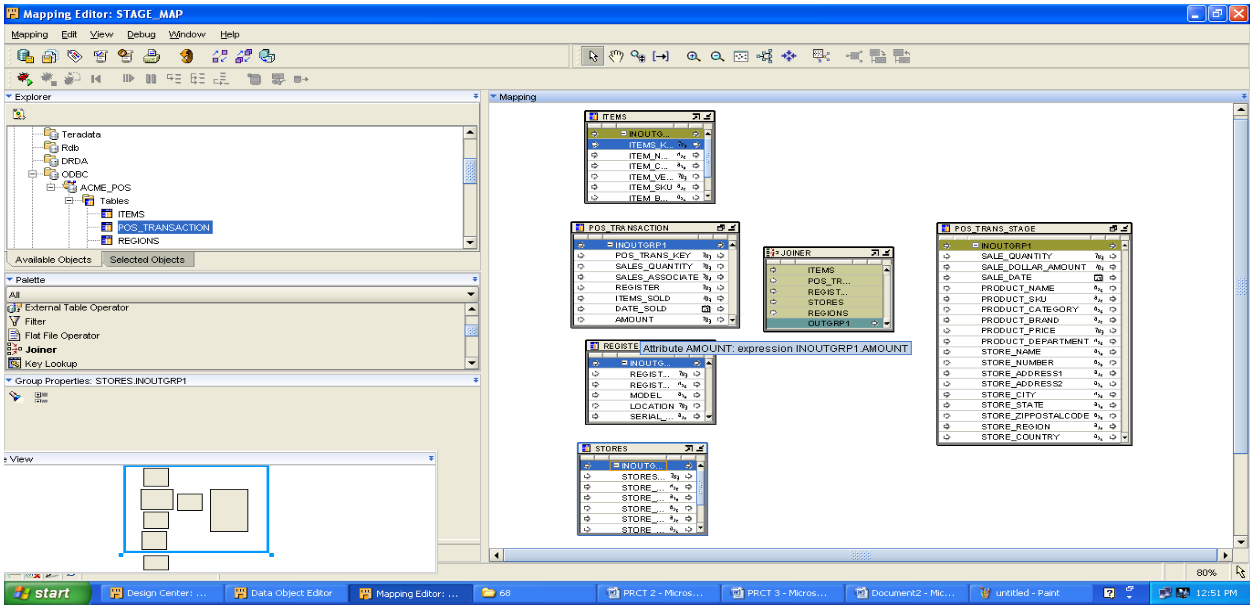


6] Right click on the header of the box and select open details to open the joiner editor. In the groups we will click three times in the add button and add the in group items. Click on the **OK** button.

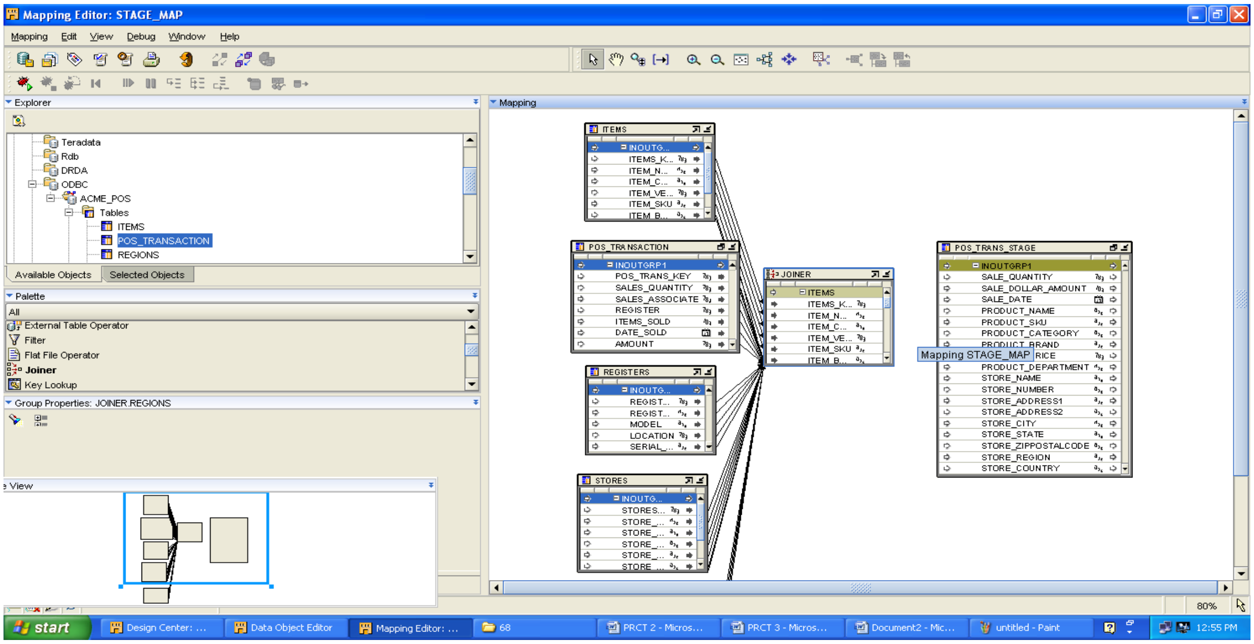


Define the groups for the JOINER operator:

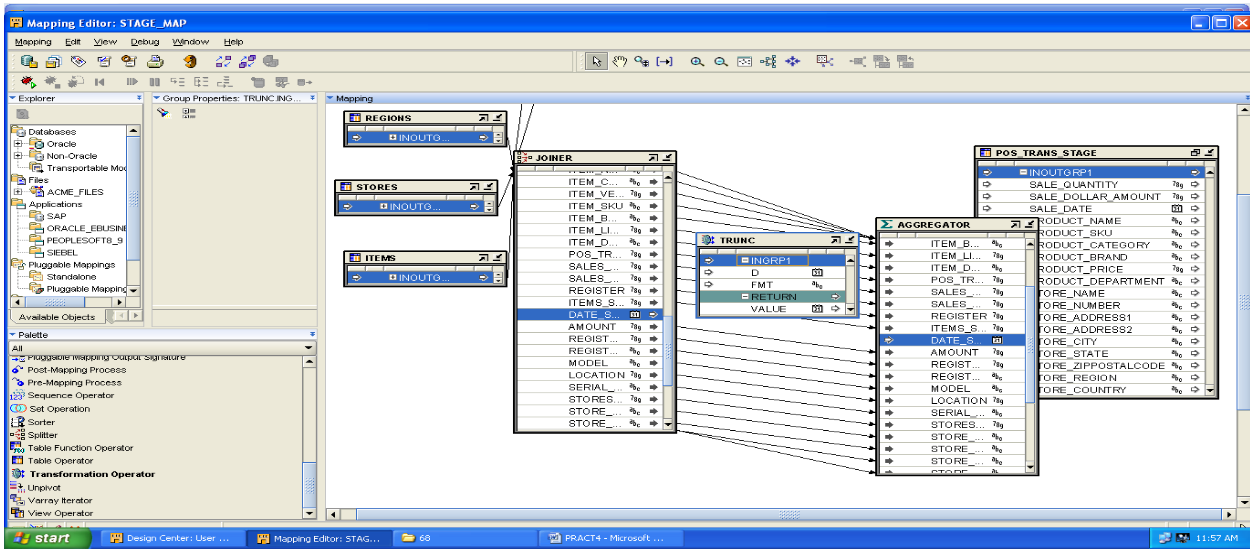
Group	Direction	Description
ITEMS	Input	
POS_TRANSA...	Input	
OUTGRP1	Output	
REGISTERS	Input	
STORES	Input	
REGIONS	Input	



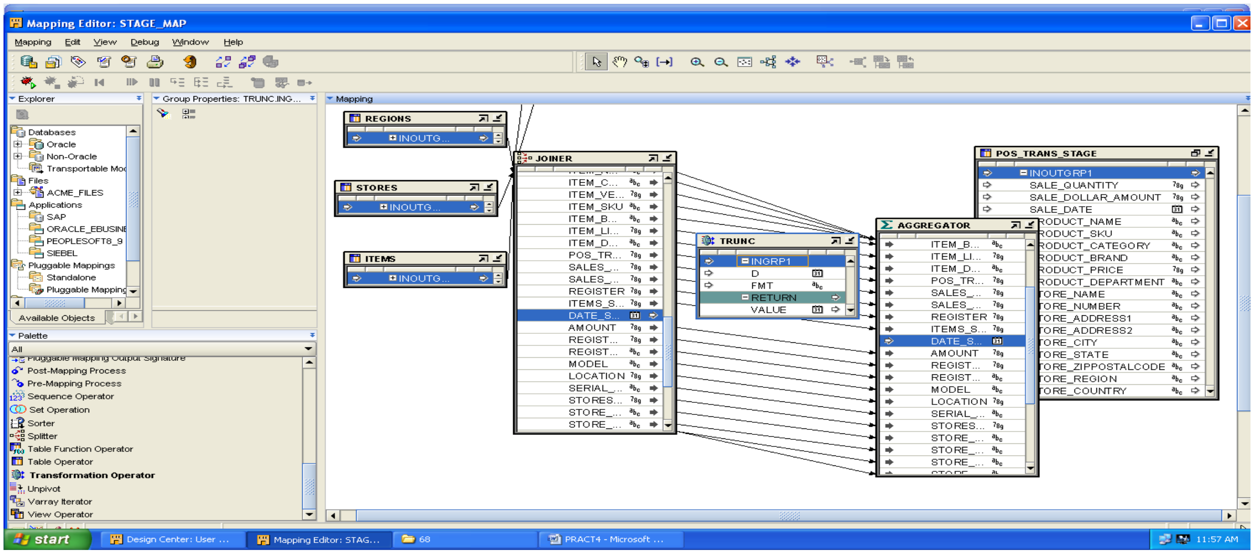
7] Connect **INOUTGRP1** of the items table to the item group of the joiner. Repeat the same procedure with all other tables.



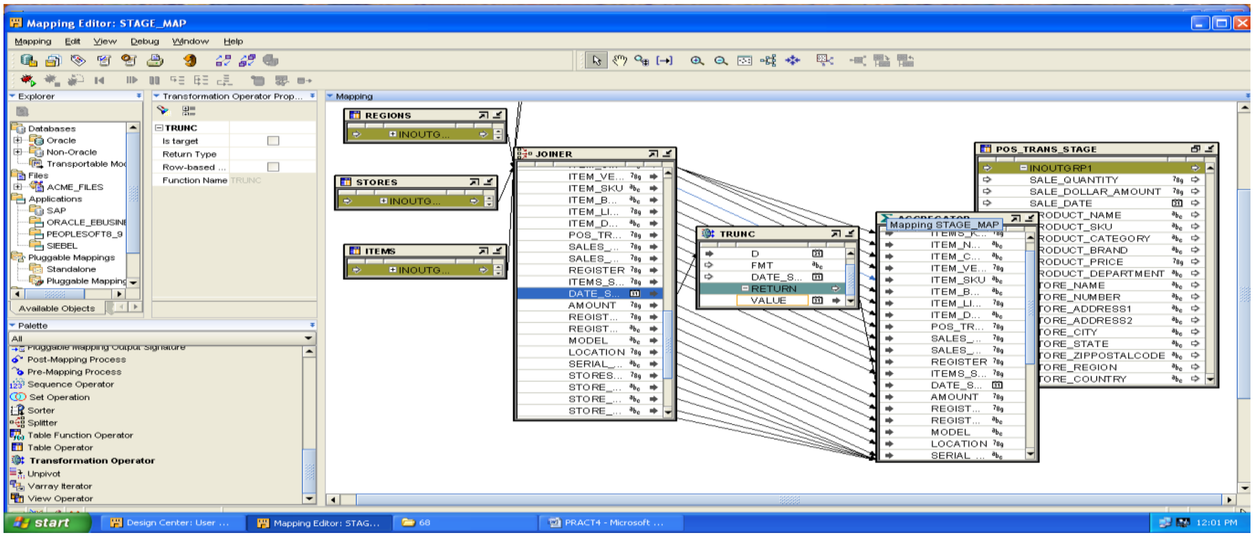
8] Click once on the header of the box to select it and the properties window. Click on the blank box to right of the **joint condition(…)** and write following query:



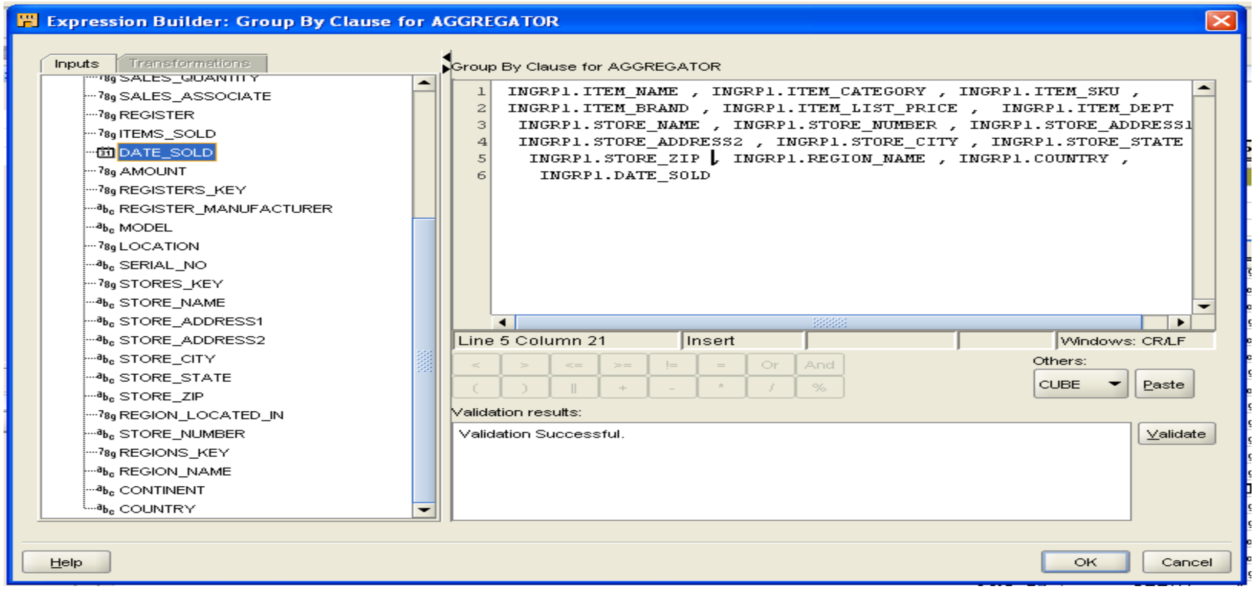
9] Drag an aggregator operator from the palette window and drop it to the right of the joiner operator. Connect the output attribute from the joiner operator as a input to the aggregator operator. Delete the line which connect to **DATE_SOLD.** Drag transformation operator from the palette window and drop it between the joiner and aggregator.



10] Connect the **DATE_SOLD** attribute in the **OUTGRP1** group of the joiner to **D** attribute of the **INGRP1** of the trunk. Then connect the value attribute of the return output group from **TRUNC** operator to the **DATE_SOLD** attribute if the INGRP1 group.



11] Select the aggregator operator and in the properties window we will click on the ellipse(…) on its right to open the **EXPRSSION BUILDER** FOR the **GROUP BY CLAUSE.** Write following query and click on the **OK** button.



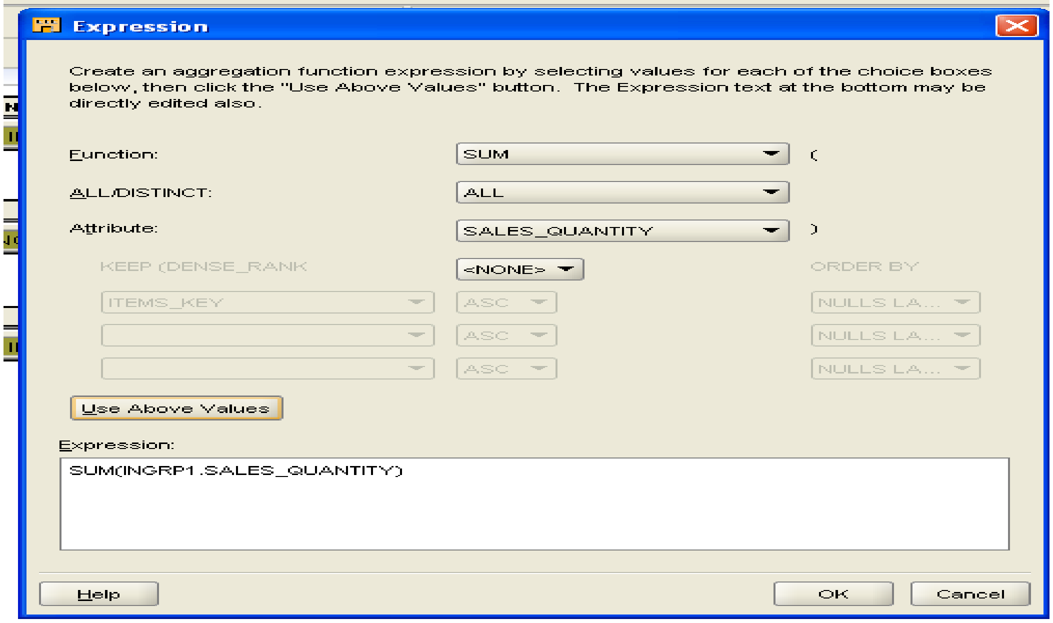
```
 INGRP1.ITEM_NAME , INGRP1.ITEM_CATEGORY , INGRP1.ITEM_SKU ,
 INGRP1.ITEM_BRAND , INGRP1.ITEM_LIST_PRICE ,  INGRP1.ITEM_DEPT
  INGRP1.STORE_NAME , INGRP1.STORE_NUMBER , INGRP1.STORE_ADDRESS1
  INGRP1.STORE_ADDRESS2 , INGRP1.STORE_CITY , INGRP1.STORE_STATE
   INGRP1.STORE_ZIP , INGRP1.REGION_NAME , INGRP1.COUNTRY ,
    INGRP1.DATE_SOLD
```

Line 5 Column 21

Validation results:

Validation Successful.

12] We will select SUM function from the drop down menu, **ALL** from the **ALL/DISTINCT** and **Sale_quantity** from the attribute. We will click on the **USE ABOVE VALUE** Button And Click **Ok**



Expression

Create an aggregation function expression by selecting values for each of the choice boxes below, then click the "Use Above Values" button. The Expression text at the bottom may be directly edited also.

Function: SUM (

ALL/DISTINCT: ALL

Attribute: SALES_QUANTITY)

KEEP (DENSE_RANK <NONE> ORDER BY

ITEMS_KEY ASC NULLS LA...

ASC NULLS LA...

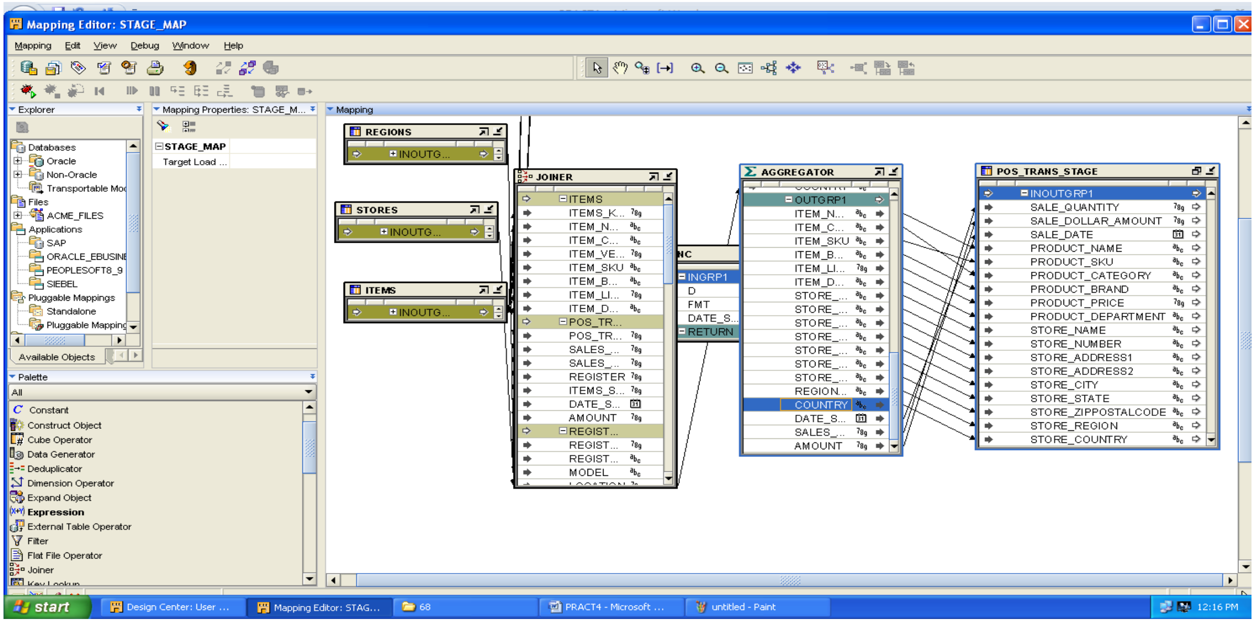
ASC NULLS LA...

Use Above Values

Expression:

```
SUM(INGRP1.SALES_QUANTITY)
```

13] Connect the following attribute from the **POS_TRANS_STAGE** table to the aggregator



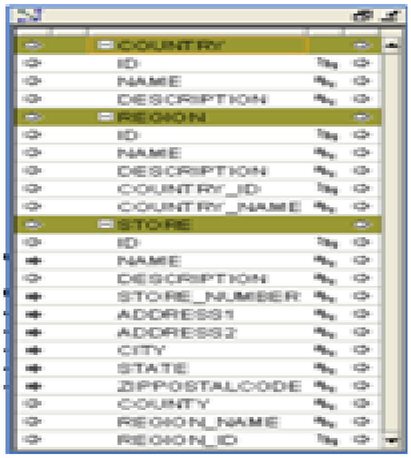
PRACTICAL NO: 05

Aim: Perform the ETL process and transform into data map.

1) Store mapping

Design centre |Right click on mappings |ACME_DW_PROJECT| Database|Oracle| ACME_DWH|Select new……. Enter store_map

1)Database|Oracle| ACME_DWH|Dimensions…. Mapping and drop it to the right of POS_TRANS_SATGE Table operator



I. Store name to name (Store level) name trim spaces & upper

II. Store number to number needs in store level need trim & upper level

III. Address1 to Address1 level needs trim(Store level)

IV. Address2 to Address2 level needs trim(Store level)

V. City to City need store level in trim

VI. State to state in store level need trim & upper

VII. Zippostalcode to zippostalcode in store level need trim spaces.

2) Region level

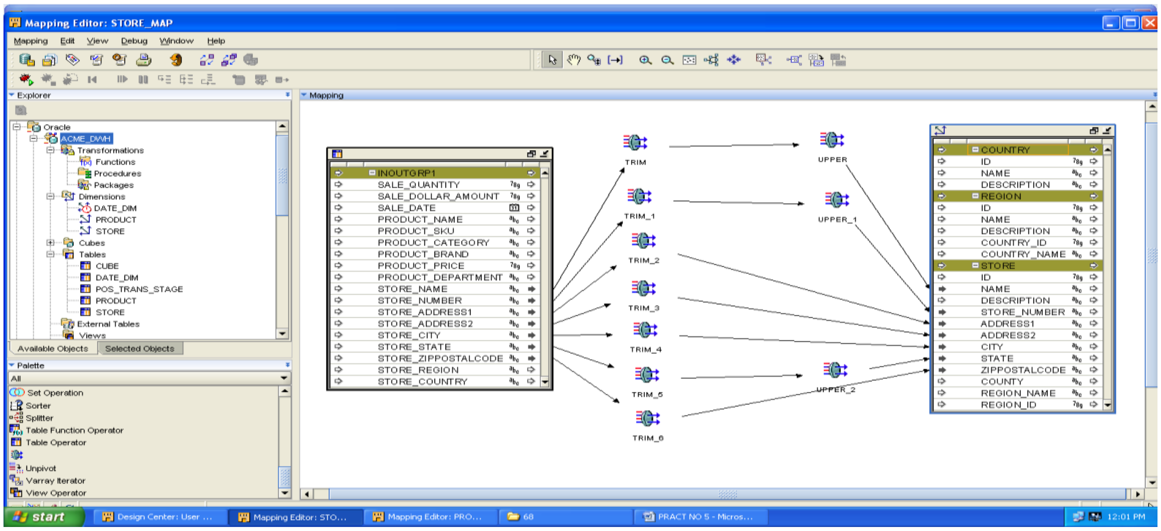
I. Store region to name in the region level using trim & upper transformation

II. Store region to Description in the region level using trim & upper transformation

III. Store country to country name in the region level using trim & upper transformation

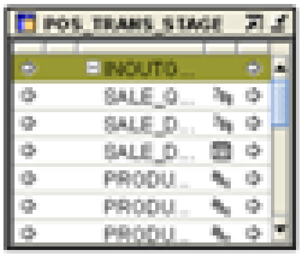
IV. Store country to name in the region level using trim & upper transformation

V. Store country to description in the region level using trim & upper transformation

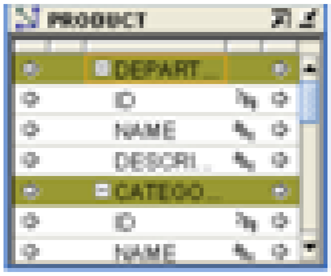


2) Product Mapping

1)Drag POS_TRANS_STAGE Table from the Explorer window drop it on the left of the mapping

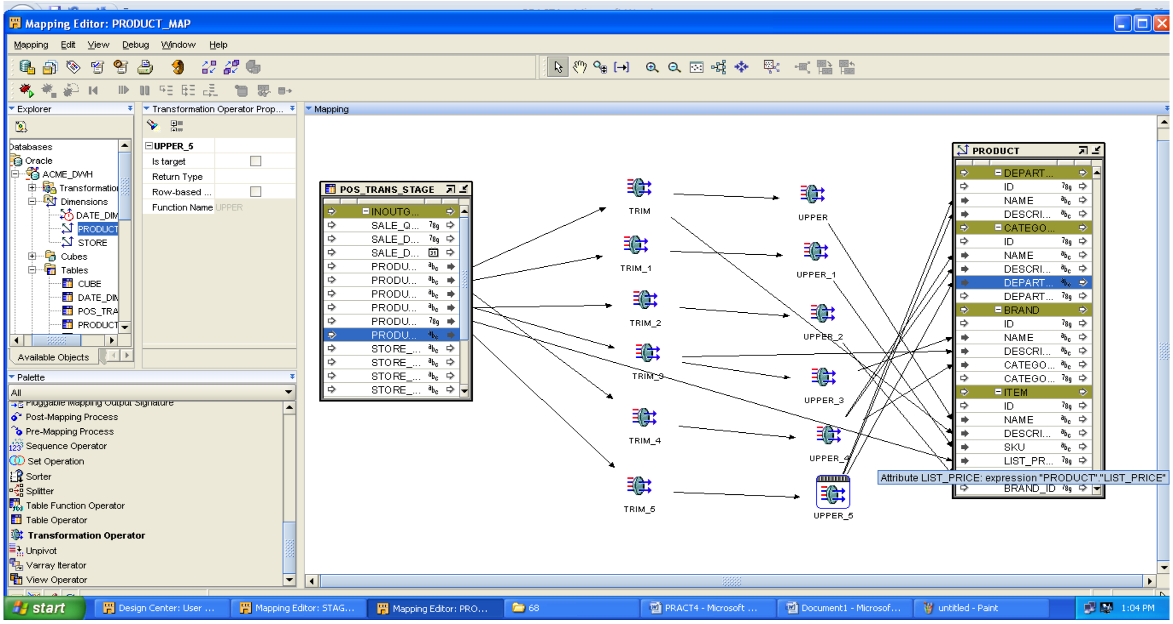


Drag the PRODUCT dimension from ACME_DWH| dimensions drop it to the right of the mapping



2)

I. PRODUCT_NAME to NAME (item level) needs trim & upper spaces.
II. PRODUCT_NAME TO DESCRIPTION (item level) needs trim & upper spaces.
III. PRODUCT_SKU TO SKU needs trim & upper spaces
IV. PRODUCT_PRICE TO LIST_PRICE no transformation needed
V. PRODUCT_BRAND TO BRAND_NAME needs trim & upper spaces
VI. PRODUCT_BRAND TO NAME needs trim & upper spaces
VII. PRODUCT_BRAND TO DESCRIPTIN (brand level) needs trim spaces
VIII. PRODUCT_CATEGORY TO CATEGORY NAME needs trim & upper spaces
IX. PRODUCT _CATEGORY TO NAME (category level) needs trim & upper spaces
X. PRODUCT_CATEGORY TO DESCRIPTION (category group) needs trim spaces
XI. PRODUCT_DEPARTMENT TO DEPARTMENT NAME needs trim & upper spaces
XII. PRODUCT_DEPARTMENT TO NAME (department level) needs trim & upper spaces
XIII. PRODUCT_DEPARTMENT TO DESCRIPTION (department level) needs trim spaces



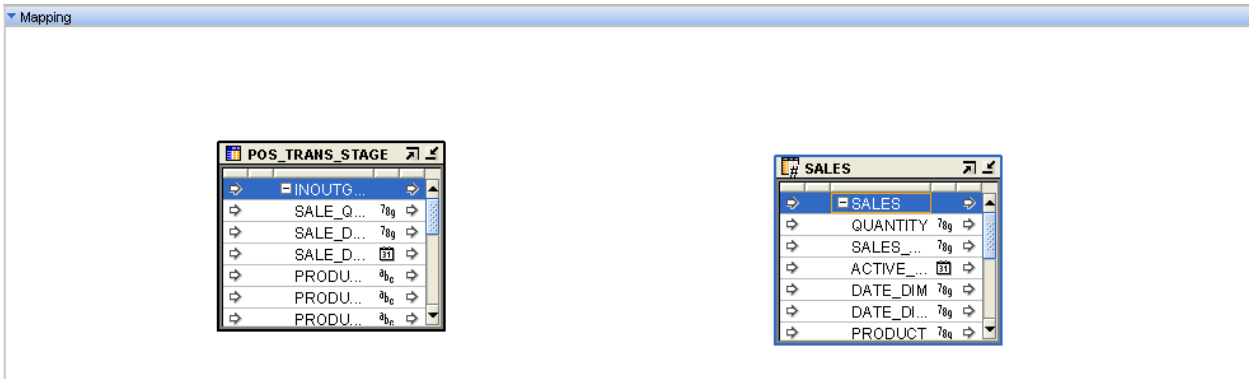
UPPER_5
Is target ☐
Return Type
Row-based ... ☐
Function Name UPPER

Attribute LIST_PRICE: expression "PRODUCT"."LIST_PRICE"

PRACTICAL NO: 06

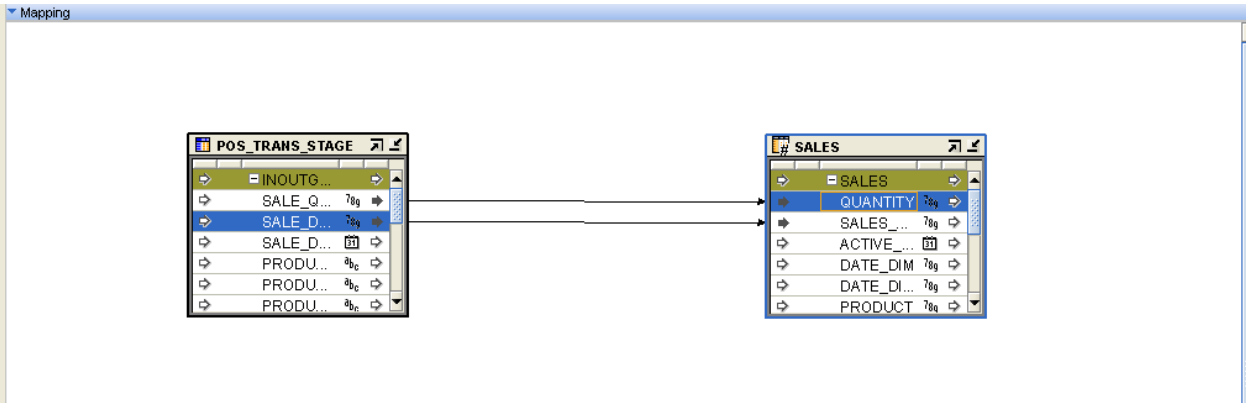
Aim: Create the cube and process it in OWB

1.SALE CUBE MAPPING:

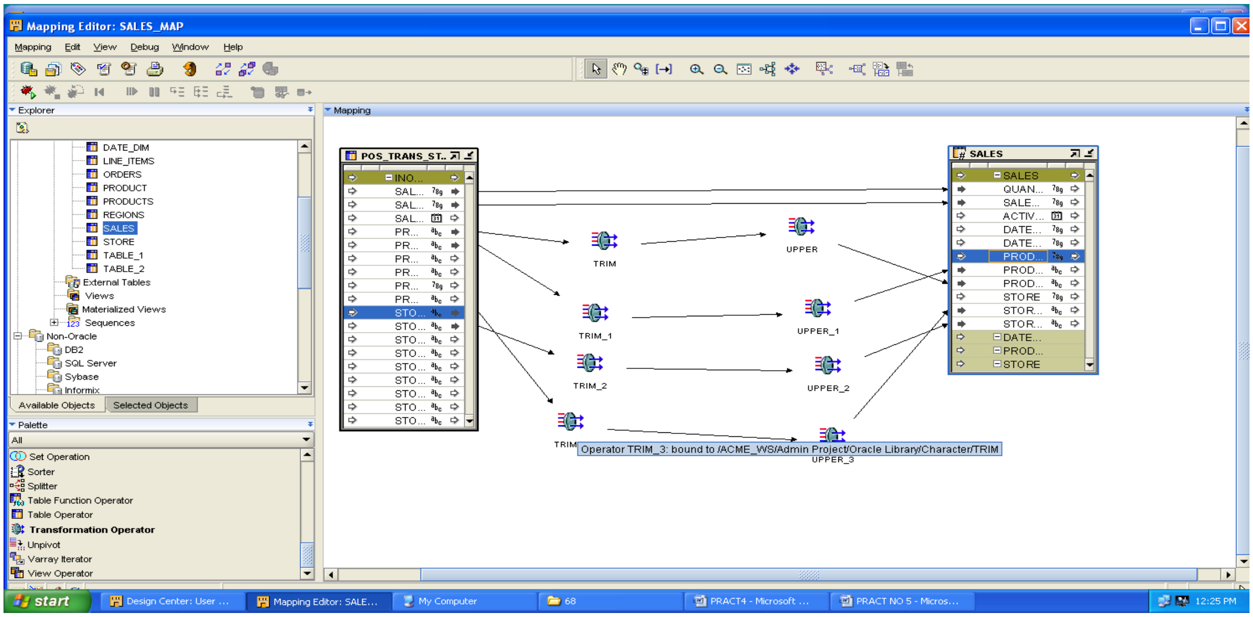


2.Mapping value to the cube attribute:

3.Drag line from SALE QUANTITY to QUANTITY & another from SALE_DOLLAR_AMOUNT to SALE_AMOUNT

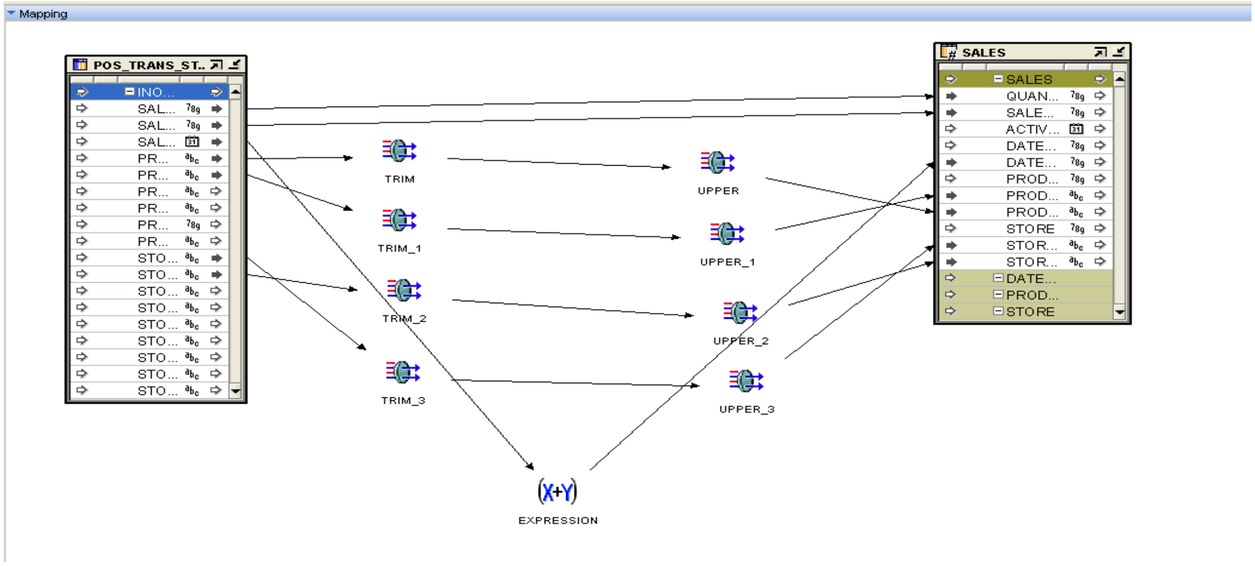


4. Mapping product & Store dimension value to the cube



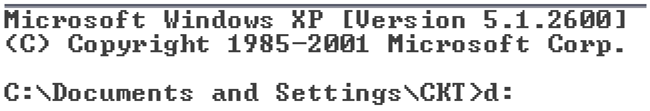
Operator TRIM_3: bound to /ACME_WS/Admin Project/Oracle Library/Character/TRIM

5.MAPPING DATE DIMENSION VALUE TO CUBE:

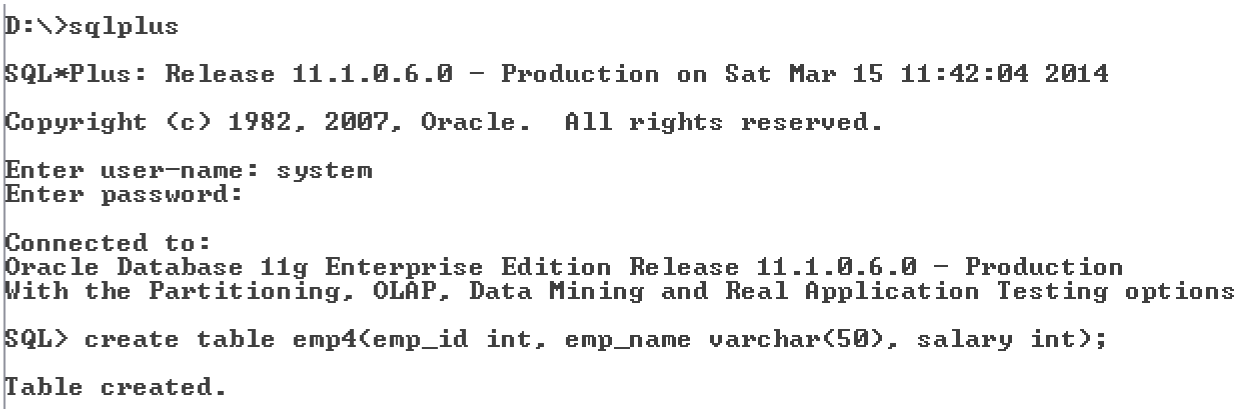


PRACTICAL NO.:-07

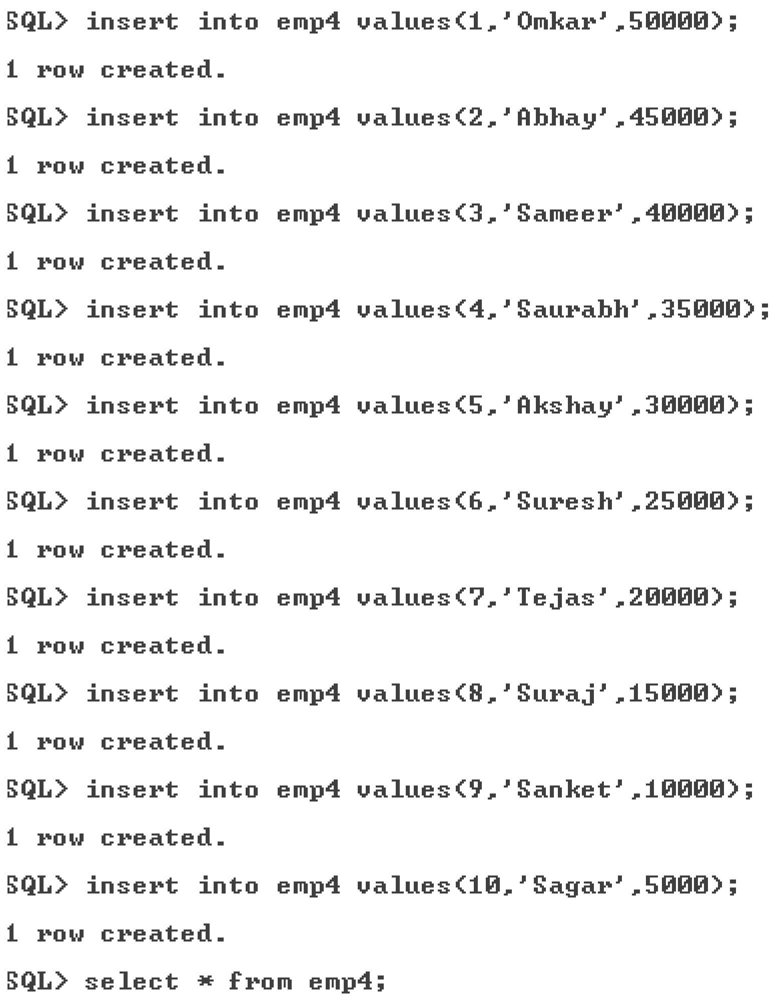
AIM:- GENERATING ORACLE REPORTS.



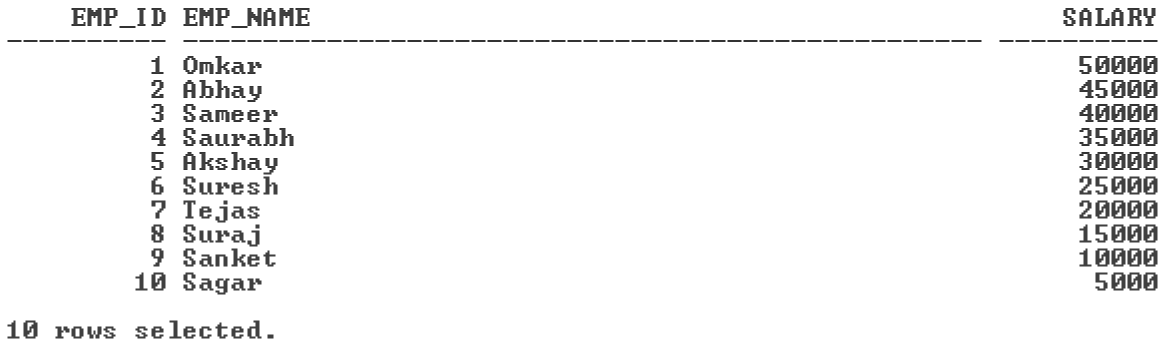
```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\CKT>d:
```



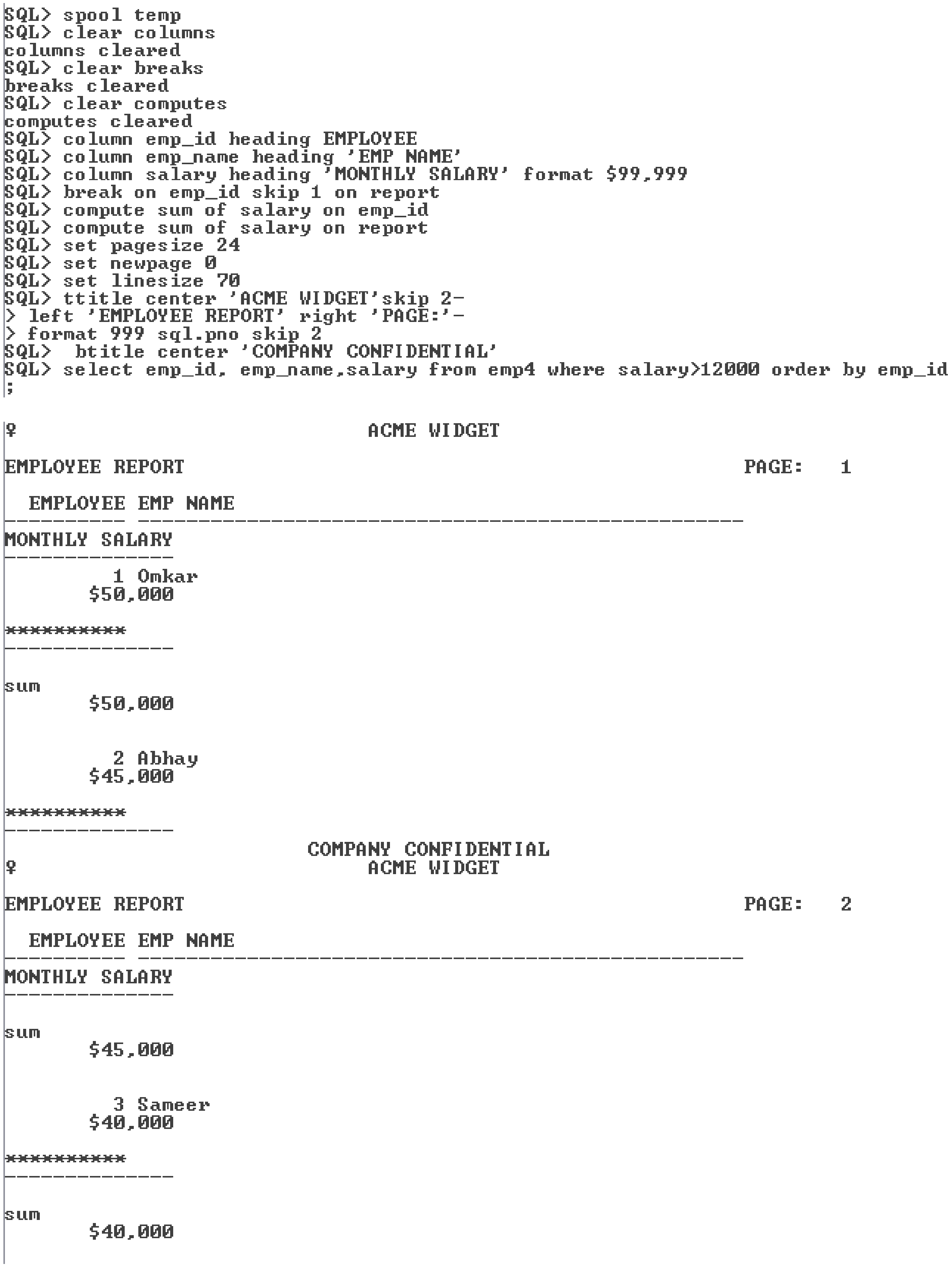
```
D:\>sqlplus

SQL*Plus: Release 11.1.0.6.0 - Production on Sat Mar 15 11:42:04 2014

Copyright (c) 1982, 2007, Oracle.  All rights reserved.

Enter user-name: system
Enter password:

Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> create table emp4(emp_id int, emp_name varchar(50), salary int);

Table created.
```



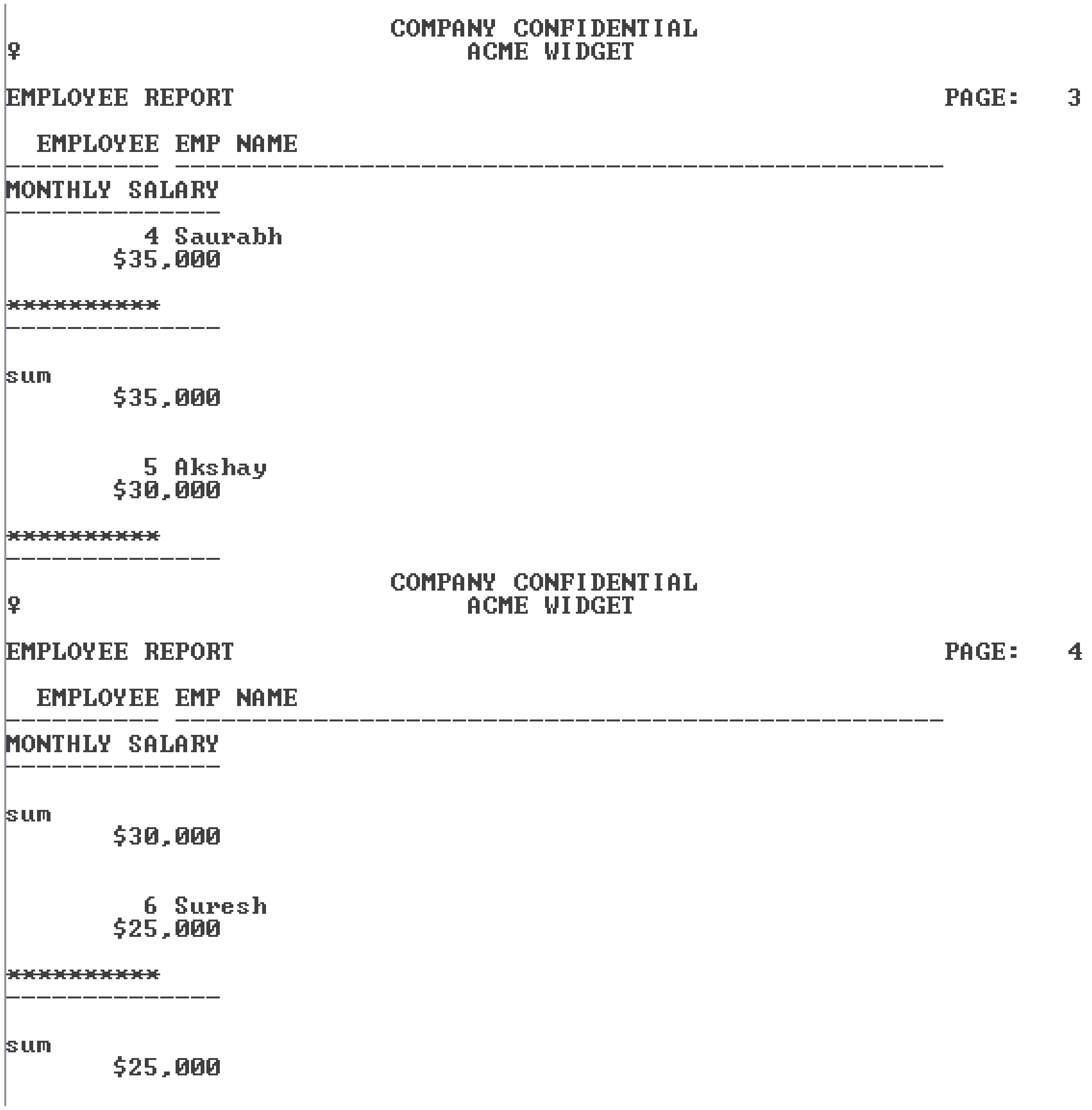
```
SQL> insert into emp4 values(1,'Omkar',50000);

1 row created.

SQL> insert into emp4 values(2,'Abhay',45000);

1 row created.

SQL> insert into emp4 values(3,'Sameer',40000);

1 row created.

SQL> insert into emp4 values(4,'Saurabh',35000);

1 row created.

SQL> insert into emp4 values(5,'Akshay',30000);

1 row created.

SQL> insert into emp4 values(6,'Suresh',25000);

1 row created.

SQL> insert into emp4 values(7,'Tejas',20000);

1 row created.

SQL> insert into emp4 values(8,'Suraj',15000);

1 row created.

SQL> insert into emp4 values(9,'Sanket',10000);

1 row created.

SQL> insert into emp4 values(10,'Sagar',5000);

1 row created.

SQL> select * from emp4;
```



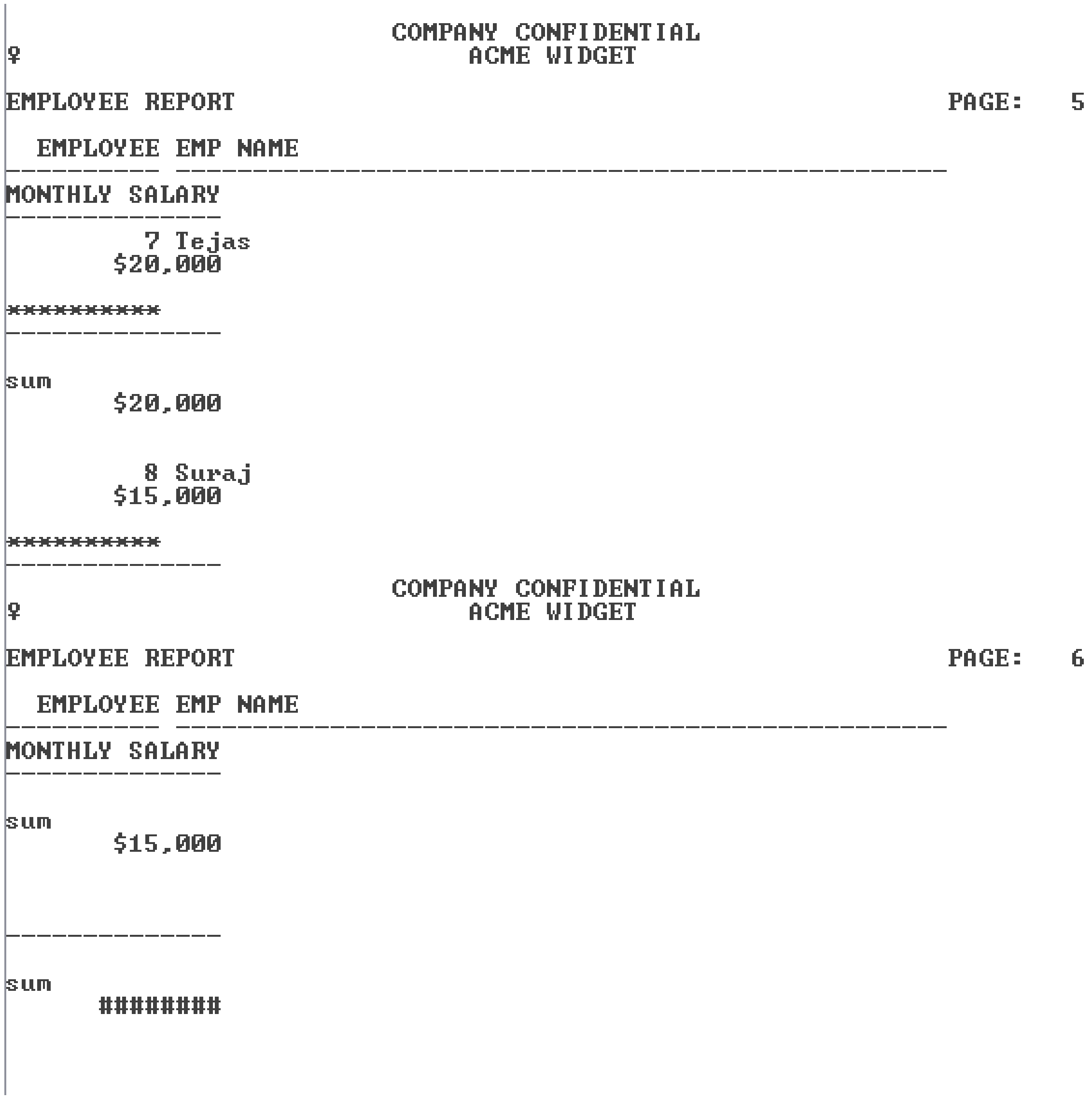
```
    EMP_ID EMP_NAME                                               SALARY
---------- -------------------------------------------------- ----------
         1 Omkar                                                   50000
         2 Abhay                                                   45000
         3 Sameer                                                  40000
         4 Saurabh                                                 35000
         5 Akshay                                                  30000
         6 Suresh                                                  25000
         7 Tejas                                                   20000
         8 Suraj                                                   15000
         9 Sanket                                                  10000
        10 Sagar                                                    5000

10 rows selected.
```



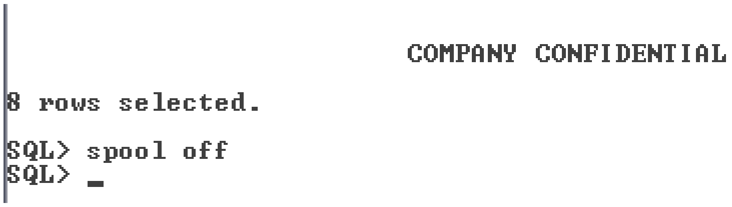
```
SQL> spool temp
SQL> clear columns
columns cleared
SQL> clear breaks
breaks cleared
SQL> clear computes
computes cleared
SQL> column emp_id heading EMPLOYEE
SQL> column emp_name heading 'EMP NAME'
SQL> column salary heading 'MONTHLY SALARY' format $99,999
SQL> break on emp_id skip 1 on report
SQL> compute sum of salary on emp_id
SQL> compute sum of salary on report
SQL> set pagesize 24
SQL> set newpage 0
SQL> set linesize 70
SQL> ttitle center 'ACME WIDGET'skip 2-
> left 'EMPLOYEE REPORT' right 'PAGE:'-
> format 999 sql.pno skip 2
SQL>  btitle center 'COMPANY CONFIDENTIAL'
SQL> select emp_id, emp_name,salary from emp4 where salary>12000 order by emp_id
;

♀                             ACME WIDGET

EMPLOYEE REPORT                                               PAGE:   1

  EMPLOYEE EMP NAME
---------- --------------------------------------------------
MONTHLY SALARY
--------------
         1 Omkar
       $50,000

**********
--------------

sum
       $50,000


         2 Abhay
       $45,000

**********
--------------
                         COMPANY CONFIDENTIAL
♀                             ACME WIDGET

EMPLOYEE REPORT                                               PAGE:   2

  EMPLOYEE EMP NAME
---------- --------------------------------------------------
MONTHLY SALARY
--------------

sum
       $45,000


         3 Sameer
       $40,000

**********
--------------

sum
       $40,000
```



```
                         COMPANY CONFIDENTIAL
♀                             ACME WIDGET

EMPLOYEE REPORT                                              PAGE:    3

  EMPLOYEE EMP NAME
---------- --------------------------------------------------
MONTHLY SALARY
--------------
         4 Saurabh
       $35,000

**********
--------------

sum
       $35,000


         5 Akshay
       $30,000

**********
--------------
                         COMPANY CONFIDENTIAL
♀                             ACME WIDGET

EMPLOYEE REPORT                                              PAGE:    4

  EMPLOYEE EMP NAME
---------- --------------------------------------------------
MONTHLY SALARY
--------------

sum
       $30,000


         6 Suresh
       $25,000

**********
--------------

sum
       $25,000
```



```
                         COMPANY CONFIDENTIAL
♀                             ACME WIDGET

EMPLOYEE REPORT                                              PAGE:   5

  EMPLOYEE EMP NAME
---------- --------------------------------------------------
MONTHLY SALARY
--------------
         7 Tejas
       $20,000

**********
--------------

sum
       $20,000


         8 Suraj
       $15,000

**********
--------------
                         COMPANY CONFIDENTIAL
♀                             ACME WIDGET

EMPLOYEE REPORT                                              PAGE:   6

  EMPLOYEE EMP NAME
---------- --------------------------------------------------
MONTHLY SALARY
--------------

sum
       $15,000



--------------

sum
      ########
```

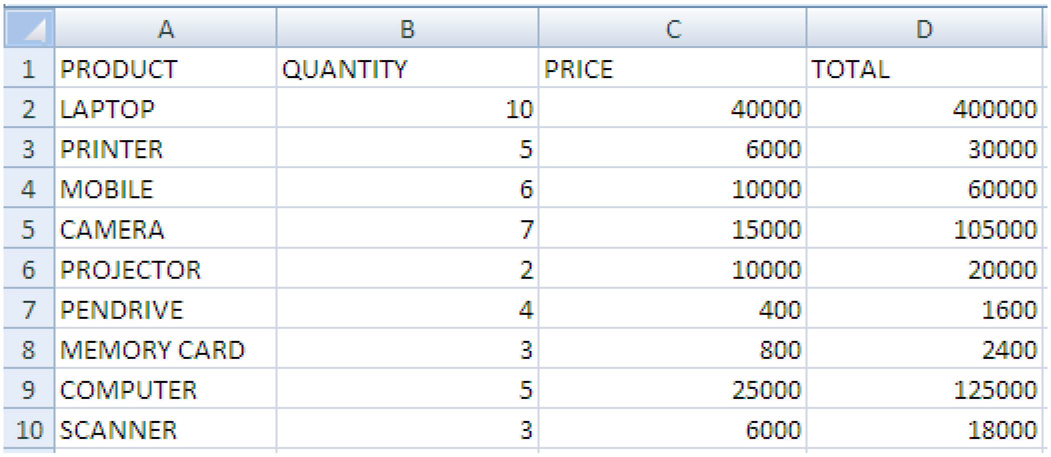


```
                         COMPANY CONFIDENTIAL

8 rows selected.

SQL> spool off
SQL> _
```

Practical No-08

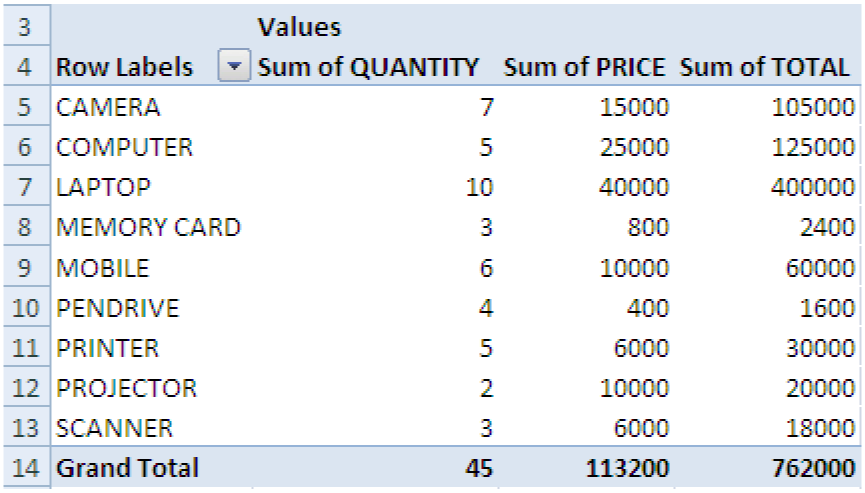
Aim:-Creating the Pivot table and pivot chart using some existing data or create a new data.

Sales Table:-



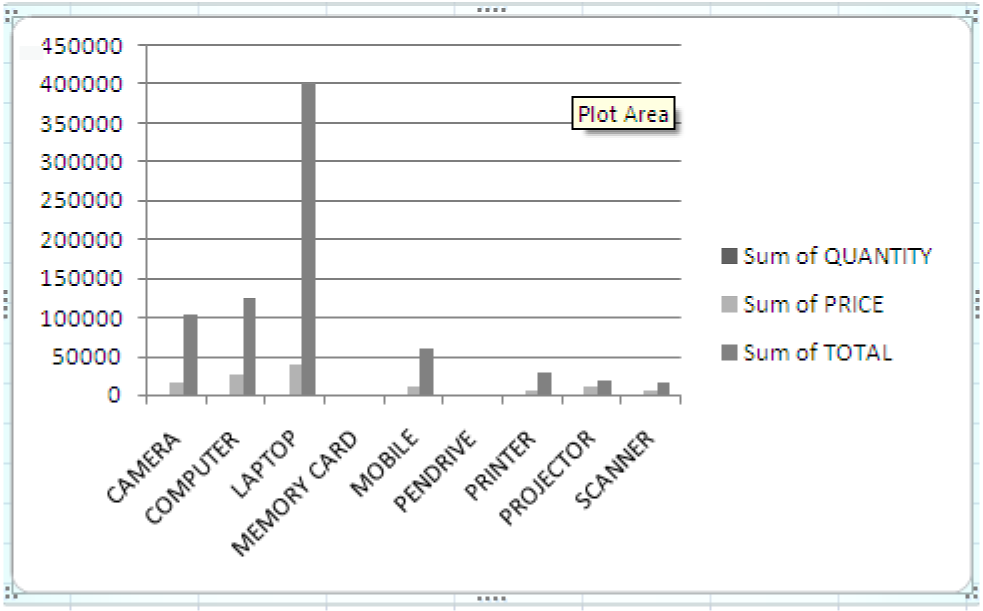
	A	B	C	D
1	PRODUCT	QUANTITY	PRICE	TOTAL
2	LAPTOP	10	40000	400000
3	PRINTER	5	6000	30000
4	MOBILE	6	10000	60000
5	CAMERA	7	15000	105000
6	PROJECTOR	2	10000	20000
7	PENDRIVE	4	400	1600
8	MEMORY CARD	3	800	2400
9	COMPUTER	5	25000	125000
10	SCANNER	3	6000	18000

Pivot Table:-



3		Values		
4	**Row Labels**	**Sum of QUANTITY**	**Sum of PRICE**	**Sum of TOTAL**
5	CAMERA	7	15000	105000
6	COMPUTER	5	25000	125000
7	LAPTOP	10	40000	400000
8	MEMORY CARD	3	800	2400
9	MOBILE	6	10000	60000
10	PENDRIVE	4	400	1600
11	PRINTER	5	6000	30000
12	PROJECTOR	2	10000	20000
13	SCANNER	3	6000	18000
14	**Grand Total**	**45**	**113200**	**762000**

Grant chart:-


450000
400000
350000
300000
250000
200000
150000
100000
50000
0
Plot Area
CAMERA
COMPUTER
LAPTOP
MEMORY CARD
MOBILE
PENDRIVE
PRINTER
PROJECTOR
SCANNER
Sum of QUANTITY
Sum of PRICE
Sum of TOTAL