

Oracle® Designer

Oracle Designer Tutorial

Release 6.0

June 1999

Part No. Z24613-01

ORACLE®

Oracle Designer Tutorial

Part No. Z24613-01

Release 6.0

Copyright © 1995, 1999 Oracle Corporation. All rights reserved.

Author: Martin Wykes

The Programs (which include both the software and documentation) contain proprietary information of Oracle Corporation; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent and other intellectual and industrial property laws. Reverse engineering, disassembly or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Oracle Corporation does not warrant that this document is error free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Oracle Corporation.

If the Programs are delivered to the US Government or anyone licensing or using the Programs on behalf of the US Government, the following notice is applicable:

Restricted Rights Notice

Programs delivered subject to the DOD FAR Supplement are 'commercial computer software' and use, duplication and disclosure of the Programs including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are 'restricted computer software' and use, duplication and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be licensee's responsibility to take all appropriate fail-safe, back up, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and Oracle disclaims liability for any damages caused by such use of the Programs.

Oracle is a registered trademark, and Oracle Designer, Oracle Developer, Net8, Oracle7, Oracle8, Personal Oracle, PL/SQL, SQL*Net and SQL*Plus are trademarks or registered trademarks, of Oracle Corporation. All other company or product names mentioned are used for identification purposes only and may be trademarks of their respective owners.

Contents

Send Us Your Comments v

Read This First!..... vii

 Purpose of this document vii

 Intended audience for this document viii

 Prerequisites for using this document ix

 Structure of this document x

 How to use this document..... x

 Conventions used in this document..... x

 Related documentation xi

 Your comments are welcome xi

Part I Before You Start

1 Introducing Oracle Designer

 What is Oracle Designer?..... 1-1

 The components of Oracle Designer 1-2

 Starting Oracle Designer tools and utilities for the tutorial 1-3

2 Preparing for the tutorial

Scope of the tutorial.....	2-1
Time to allow for the tutorial.....	2-2
How to follow the tutorial.....	2-2
Removing existing tutorial tables.....	2-3
Background to the tutorial.....	2-4

Part II Tutorial

Lesson 1 Create process steps and add flows

Create a process step.....	Le 1-2
Add a flow, a trigger and an outcome.....	Le 1-5

Lesson 2 Create entities and add relationships

Create an entity.....	Le 2-2
Add a relationship.....	Le 2-5
Create attributes.....	Le 2-7

Lesson 3 Refine default functions

Identify functions to automate.....	Le 3-1
Add entity usages to a function.....	Le 3-6
Create attribute usages using the function/attribute matrix utility.....	Le 3-7

Lesson 4 Transform the data model into a table design

Create a table definition from entities.....	Le 4-2
--	--------

Lesson 5 Refine the default database design

Modify a default table definition.....	Le 5-2
--	--------

Lesson 6 Build a database

Create the database tables..... Le 6-1

Lesson 7 Create default working forms

Generate default module definitions Le 7-2

Accept the candidate module definitions for generation Le 7-3

Generate a working form..... Le 7-6

Lesson 8 Refine modules for generation

Change form characteristics Le 8-1

Create a lookup table..... Le 8-3

Lesson 9 Generate a refined Oracle Developer Form

Set preferences to influence generation Le 9-1

Change a display characteristic of the generated form..... Le 9-3

Generate, run and use a form..... Le 9-4

Lesson 10 Generate a WebServer Application

Implement the Speedy Pizza system as a Web application..... Le 10-1

Lesson 11 Conclude the tutorial

Exit from Oracle Designer and drop the tutorial tables..... Le 11-1

A Tutorial installation notes

Importing the TUTORIAL application system A-1

Setting up for WebServer Generator users A-4

What the Oracle Installer installs on the client..... A-4

How tutorial users are instructed to start the tutorial A-5

Send Us Your Comments

Oracle Designer Tutorial, Release 6.0

Part No. Z24613-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the chapter, section, and page number (if available). You can send comments to us in the following ways:

- docdes@uk.oracle.com
- +44 (0)118 9245019. Attn: Oracle Designer
- Oracle Corporation UK Ltd
Oracle Designer Documentation
Oracle Parkway
Thames Valley Park
Reading, Berks. RG6 1RA
United Kingdom

If you would like a reply, please give your name, address, and telephone number below.

If you have problems with the software, please contact your local Oracle World Wide Support Center

Read This First!

Before you start reading this document, it is important that you understand the following:

- the purpose of this document
- the intended audience for this document
- the prerequisites for using this document
- the structure of this document
- how to use this document
- conventions used in this document
- related documentation
- that your comments are welcome!

Purpose of this document

This document is a tutorial for Oracle Designer. It aims to:

- provide you with practical hands-on experience using Oracle Designer
- reinforce concepts explained in the Oracle Designer help system and other parts of the documentation set

Upon completion of this tutorial, you will know how to:

- create process steps and flows
- create entities and relationships
- create definitions of database objects in the Repository

- refine the default database design
- generate objects in the database
- refine default functions
- create default module definitions
- set preferences for generation
- create lookup tables
- generate a form
- generate a WebServer application

This tutorial does not attempt to describe in detail the functionality of every Oracle Designer product component. Instead, the tutorial concentrates on how to use the components together to progress from a high level business model to a working generated application. For comprehensive information, refer to the Oracle Designer help system.

Note that to enable you to use several different Oracle Designer product components within a relatively short period, the tutorial assumes a very small business scenario. This scenario is not representative of the more sophisticated environments in which Oracle Designer is typically used.

Intended audience for this document

This tutorial is intended for analysts and designers who want to know how to use the Oracle Designer tools and utilities to model, design and implement business systems.

This tutorial does not explain in detail terms that are familiar to such an audience (for example, entity relationship diagrams, function hierarchy diagrams and SQL DDL commands).

Prerequisites for using this document

This tutorial does not assume you have any prior experience with Oracle Designer. However, this tutorial does assume that:

- you are familiar with the operation of your personal computer (PC) and its keyboard.
- you are familiar with the Microsoft Windows 95/98 or Windows NT environment, as appropriate.
- the PATH variable on your PC includes the appropriate path to the \bin directory below the <ORACLE_HOME> directory (the directory in which Oracle products are installed). For example:

Environment	Example ORACLE_HOME	PATH must include
Windows 95	C:\orawin95	C:\orawin95\bin
Windows NT	C:\orant	C:\orant\bin

- all components of Oracle Designer have been successfully installed on your PC (including the tutorial form and menu executables and templates).
- the Forms and Reports components of Oracle Developer have been successfully installed on your PC.
- you have a user account with access to the TUTORIAL application system on a Oracle Designer Repository

For more information about importing the TUTORIAL application system, refer to Appendix A “Tutorial Installation Notes”.

Note: If you are going to follow lesson 10, "Generate a WebServer Application", you will need to have WebServer Generator installed. You will also need to know additional connection details, as set out at the beginning of the lesson. Your System Administrator will be able to set up WebServer Generator and tell you the necessary connection details.

Structure of this document

The tutorial is organized into the following Parts:

Part I “Before You Start”, introduces Oracle Designer and prepares you for the tutorial.

Part II “Tutorial”, comprises 11 lessons, each made up of one or more tasks.

How to use this document

If you are not familiar with the Oracle Designer toolset and how the tools relate to one another, refer to Part I, “Before You Start”. Each lesson in Part II begins with a brief discussion of the concepts behind the tasks in the lesson. If you are already familiar with these concepts, go straight to the first task in the lesson. The concepts are covered in more detail in the Oracle Designer help system.

Conventions used in this document

Menu selections are shown with an arrow separating the name of the menu and the name of the option. For example, to show a selection of the New option from the File Menu, the following appears in the text:

Select File → New

Characters you type are shown in monotype. For example:

Type Order Request

The screenshots in this document are of Oracle Designer Release 2.1 running in a Windows NT environment. You might notice some minor differences between what you see on your screen and the corresponding screenshot if:

- you are running Oracle Designer in a Windows 95 or Windows 98 environment
- you are running a later release of Oracle Designer

Related documentation

Information about Oracle Designer products is available in printed form in the following manual:

- *Oracle Designer Installation Guide*
Part No. Z24614-01

Your comments are welcome

We appreciate your comments as an Oracle user and a reader of the documentation. If you would like to contact us with any comments that you may have about this (or other) Oracle documentation, please use the following address or telephone us on 01189 245019 (for Oracle users overseas, + 44 1189 245019).

Oracle Designer User Documentation, Oracle Corporation UK Ltd, Oracle Parkway, Thames Valley Park, Reading RG6 1RA, United Kingdom.

Part I

Before You Start

Introduces Oracle Designer and prepares you for the tutorial.

Introducing Oracle Designer

This chapter introduces the Oracle Designer tools and utilities and contains the following sections:

- What is Oracle Designer?
- The Components of Oracle Designer
- Starting Oracle Designer Tools

What is Oracle Designer?

Oracle Designer is a software tool for analyzing business requirements and for designing and generating client/server systems that meet those requirements. Oracle Designer incorporates support for business process modeling, systems analysis, software design and system generation.

Oracle Designer provides a multi-user Repository and is closely integrated with Oracle Developer, Oracle's client/server development toolset. In this way, Oracle Designer allows organizations to design and rapidly deliver scalable, client/server systems that can adapt to changing business needs.

The components of Oracle Designer

The Oracle Designer tools are grouped into areas that reflect the primary needs of its different types of user:

- **Modeling System Requirements**
Use the tools in this area to: model business processes; re-examine the methods used to achieve the goals of the organization; create diagrammatic representations of business processes; record details about them; describe business requirements in detail; create diagrammatic models of the entities, functions and flows of data in the systems that make up the organization.
- **Generating Preliminary Designs**
Use the Transformers to generate preliminary designs from the models created earlier.
- **Designing and Generating**
Use the tools in this area to: design a system that meets the business requirements of an organization; provide a development environment for system engineers and designers; create server-side components and client-side applications from definitions recorded in the Repository.
- **Utilities**
Use the tools in this area to: enter and edit information in the Repository; show relationships between elements in the Repository; generate predefined and custom reports on the contents of the Repository; administer Repositories; write SQL-based interactive queries.

In the development environment, Oracle Designer can be installed flexibly. For example, you could install the entire Oracle Designer suite on a client PC in a single operation, or you could install the components of Oracle Designer separately. Each of the generators can be installed separately.

For this tutorial, you must install Oracle Designer as described in Appendix A.

For full details about the installation process, see the *Oracle Designer Installation Guide*.

Oracle Corporation also publishes the Repository Application Programmatic Interface (API), which provides a programmatic way of creating and maintaining information in the Repository. The API is viewable as part of the Oracle Designer help system.

Starting Oracle Designer tools and utilities for the tutorial

Different ways to start Oracle Designer tools and utilities

You can start Oracle Designer tools and utilities in one of two different ways:

- from within a currently running tool, using the Tools menu or the Utilities menu as appropriate
- from the appropriate button in the Oracle Designer window.

During this tutorial you will be told to start most of the Oracle Designer tools from the Oracle Designer window.

Displaying the Oracle Designer Window for the tutorial

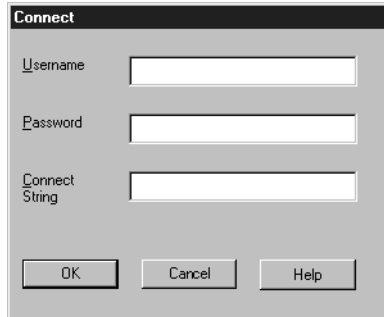
When you are following the tutorial, select the Oracle Designer for Tutorial User program in the Oracle Designer Tutorial folder. **Do not** use the program in the Oracle Designer folder. Using the Oracle Designer for Tutorial User program will ensure that:

- items produced by the Oracle Designer tools and utilities during the course of the tutorial are kept separate from any other work you do with Oracle Designer
- you won't have to specify the location of the object libraries and templates required for the tutorial, only their names.

Note: The working directory specified for the Oracle Designer for Tutorial User program must be writeable.

Connecting to a database for the tutorial

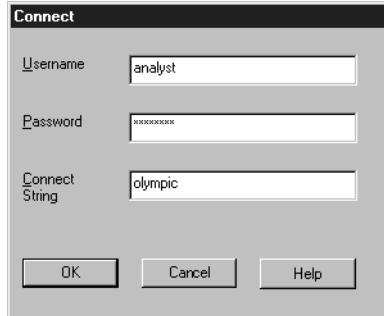
Depending on how Oracle Designer has been installed on your system, the Connect dialog box might appear when you start the Oracle Designer for Tutorial User program.



The image shows a dialog box titled "Connect". It contains three text input fields: "Username", "Password", and "Connect String". Below the fields are three buttons: "OK", "Cancel", and "Help".

If this dialog box appears, enter the username and password for the user account you will be using during the tutorial. In the Connect String field, enter the name of the database you will be using.

Below is an example of a completed Connect dialog box.

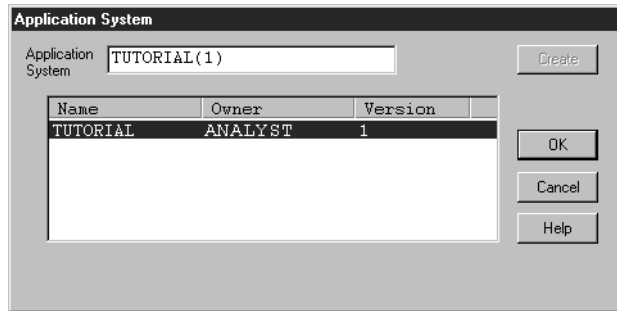


The image shows the same "Connect" dialog box, but with the fields filled in. The "Username" field contains the text "analyst", the "Password" field contains a series of asterisks "*****", and the "Connect String" field contains the text "olympic". The "OK", "Cancel", and "Help" buttons are still present at the bottom.

In this example, a user with the username 'analyst' and a password of 'teachme1' is attempting to connect to a database called 'olympic'.

If you are not sure what to enter in the Connect dialog box, ask your system administrator.

The application system dialog box appears, enabling you to select the application system you will be using to follow the tutorial.

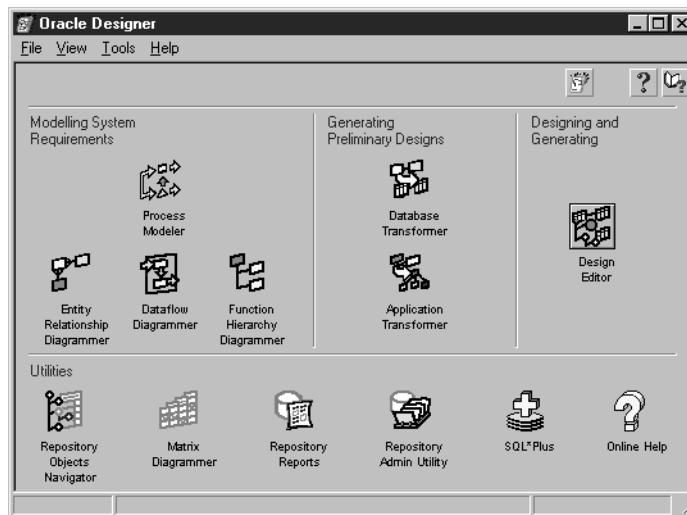


Unless your system administrator has told you to use a different application system, select TUTORIAL from the Application System drop-down list and click on OK.

The application system you specify is used when you start any of the Oracle Designer tools.

Starting tools and utilities from the Oracle Designer window

When you have logged on to a database and selected an application system, the Oracle Designer window appears.



To start a Oracle Designer tool from the Oracle Designer window, you click once on the button for that tool.

Preparing for the tutorial

This chapter describes the tutorial in this document, and includes the following sections:

- Scope of the tutorial
- Time to allow for the tutorial
- How to follow the tutorial
- Removing existing tutorial tables
- Background to the tutorial

Scope of the tutorial

This tutorial does not attempt to describe in detail the functionality of every component of Oracle Designer. Instead, the tutorial concentrates on how to use the components together to progress from a high level business model to a working generated application. For comprehensive information, refer to the Oracle Designer help system.

Note that to enable you to use several different Oracle Designer product components within a relatively short period, the tutorial assumes a very small business scenario. This scenario is not representative of the more sophisticated environments in which Oracle Designer is typically used.

Time to allow for the tutorial

The length of time to allow to complete the tutorial will depend on:

- the specification of your system
- your current familiarity with Oracle Designer (the tutorial does not assume any prior knowledge of Oracle Designer)

Typically, allow around 90 minutes to complete the tutorial.

How to follow the tutorial

Before you follow the tutorial, the following must already have been done:

- The TUTORIAL application system must have been imported into the Repository. (If you are responsible for importing the application system, refer to Appendix A for more information.)

To follow the tutorial for the first time, you must begin by ensuring that the database tables that will be created during the tutorial do not already exist. You do this by following the instructions in the next section "Removing existing tutorial tables". The remainder of this chapter sets the scene for the tutorial.

Start the tutorial with the first task in Lesson 1 and continue until you reach the end of the final task in the last lesson. If you attempt to start the tutorial half way through, you will be unable to complete some of the lessons.

If you want to take a break during the tutorial and resume later, we suggest you stop between lessons, rather than in the middle of a lesson.

Removing existing tutorial tables

To remove existing tutorial tables:

1. Select the Drop Tutorial Tables program in the Oracle Designer Tutorial folder.

The SQL*Plus window appears.

2. When prompted to enter a username, enter the username, password and database you will be using to follow the tutorial on one line, in the following format:

```
Enter user-name:<your_username>/<your_password>@<your_database>
```

For example, if the username is 'analyst', the password is 'teachme1' and the database you are using is called 'olympic', you would type the following:

```
Enter user-name:analyst/teachme1@olympic
```

If you are not sure what to enter, ask your system administrator.

You are prompted to confirm you want to delete the following tables (if they exist):

```
CUSTOMERS  
ORDERS  
PRICED_PRODUCTS  
ITEMS  
PRODUCTS
```

Do not delete these tables unless you are sure these tables are no longer required. If you are not sure, type N and press [Return]. Ask your system administrator to create a new user for you to follow the tutorial, and repeat steps 1 and 2.

3. To delete the tables, type Y and press [Return].

The tutorial tables are removed.

When the operation is complete, you are prompted to close the window.

4. Type EXIT and press [Return] to close the SQL*Plus window.

To drop the tutorial tables after you have completed the tutorial:

- Follow the procedure set out in "To remove existing tutorial tables", above.

For more information, refer to Lesson 11, "Conclude the Tutorial".

Background to the tutorial

Analysts and developers at a pizza delivery company called Speedy Pizza have already carried out some analysis and design activities to produce a system for taking customer orders and despatching pizzas.

Your task is to make additions to the design of the system and to implement a small part of it.

In particular, you will create an on-screen form for entering details of the company's new products. These details will be stored in a database that you will design and create.

If you have Oracle's WebServer Generator installed, you can implement the same system as a Web application.

During this tutorial

During the course of this tutorial you will make the necessary changes to the information held in the Oracle Designer Repository. You will:

- Add new process steps for "Add new product to the menu" and "Publish menu on the web".
- Enter a new entity, **PRODUCT**, required by the "Add new product to the menu" process step. You will define the attributes of **PRODUCT** and specify the relationship between it and other entities.
- Enter new entity usages and attribute usages for the "Add new product to the menu" function (the "Add new product to the menu" process step is stored in the Repository as a function).
- Create default table definitions (database design) in the Repository from the **PRODUCTS** entity and other entities.
- Refine the default database design by adding an extra column to a table.
- Define and generate the database in which the Speedy Pizza company will store its information.

- Create default module definitions that provide the functionality to implement the system. You will make these default definitions ready for generation, then use one of them to generate an initial working form ("Priced Products") for the input of product information. You can use this form to record information on the database, if you wish.
- Refine one of the module definitions further, to make significant changes to the appearance and operation of the Priced Products form.
- Make further refinements to the appearance of the Priced Products form, then generate a final version and use it.
- If you have Oracle's Webserver Generator installed, implement the Priced Products form as a Web application.

Part II

Tutorial

Comprises 11 lessons, each made up of one or more tasks.

Lesson 1

Create process steps and add flows

Processes play a critical role in the way a business works. Modeling a process enables you to both understand and improve existing processes, and also to design new processes.

With Oracle Designer, you model a business process using a process diagram. The diagram shows the sequence of process steps that create a product or service within the business, and the flow of information between them. Some products and services might be for customers outside the organization. Other processes provide internal benefit and help manage the business.

Process Modeler enables you to create process diagrams to model business processes. The data from a process diagram is stored in the Oracle Designer Repository and is available for use by other Oracle Designer tools. In particular, process steps are stored in the Repository as functions, which you can access using the Function Hierarchy Diagrammer.

A process diagram has already been created to model customer order processing at a pizza delivery company. During this lesson you will:

- enter new process steps called ADD NEW PRODUCT TO THE MENU and PUBLISH MENU ON THE WEB in the Satisfy Customer Order process
- indicate the flow of information between the new process steps

Note: Before you start this lesson, make sure you have followed the instructions in Chapter 2, “Preparing for the Tutorial”.

Create a process step

A process step represents an individual task within a business process. Each business process undertaken by an organization can be broken down into individual process steps.

When you save a process step in the Repository, it is saved as a business function.

Objective

To add new process steps to the existing diagram modeling the Satisfy Customer Order process. These process steps are to be represented on the diagram by two of the icons shipped with Process Modeler.

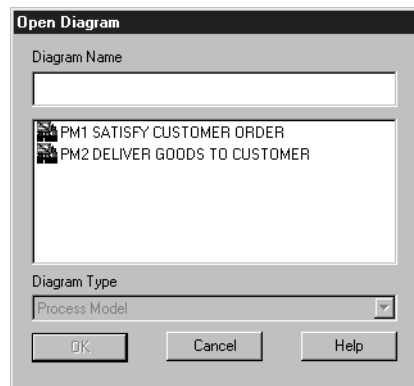
Instructions

1. Display the Oracle Designer front panel (follow the instructions in “Starting Oracle Designer Tools and Utilities for the Tutorial” in Chapter 1).
2. Click on the Process Modeler button on the Oracle Designer front panel to start Process Modeler.

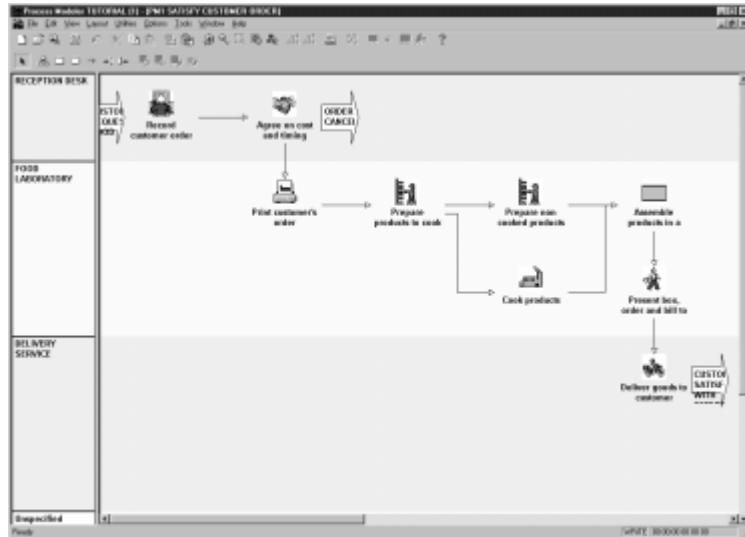


Process
Modeler

3. Select File → Open to display the Open Diagram dialog box.



4. Double click on PM1 SATISFY CUSTOMER ORDER to open the diagram.
5. Select View → Iconic to make sure the diagram appears in iconic mode (as shown below).



6. Click on the Create Process Step toolbar button.



7. Move the mouse pointer into the lowest swim lane (the horizontal strip next to the 'Unspecified' organization unit).
8. Click the mouse button.

The Create Process Step dialog box is displayed.

9. Enter the following details for the new process step:

Short Definition: Add new product to the menu

Label: DELI 10.4 (AUTO)

10. Click on OK to close the dialog box.

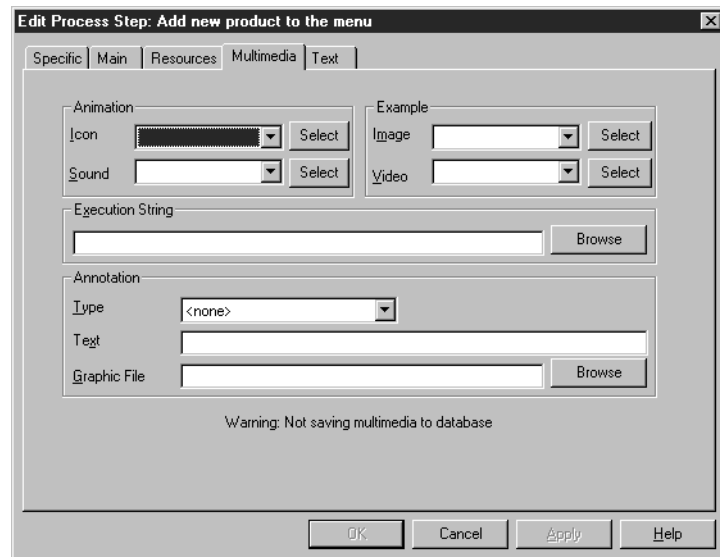
The new process step is represented on the diagram by an icon, as shown below.



**Add new product
to the menu**

You can now change the icon that represents the new process step.

11. Double click on the ADD NEW PRODUCT TO THE MENU process step to display the Edit Process Step dialog box.
12. Display the Multimedia tab of the Edit Process Step dialog box.



Warning: Not saving multimedia to database

13. Select perscmp1.bmp from the Animation Icon drop-down list.
14. Click on OK to close the Edit Process Step dialog box.

15. In the lowest swim lane, add a second process step as described in step 7 to 10, above. For this second process step, enter the following details:

Short Definition: Publish menu on the Web

Label: DELI 10.5 (AUTO)

16. Using the procedure described in steps 11 to 14, above, change the icon of the new process step to globe1.bmp.

Add a flow, a trigger and an outcome

A flow represents the movement of information between individual elements (e.g. process steps) on a process diagram. A trigger is an event that initiates the business process that a process diagram illustrates. An outcome is an event that results from the business process that a process diagram illustrates.

Objective

To add a flow to the process diagram, between the ADD NEW PRODUCT TO THE MENU process step and the PUBLISH MENU ON THE WEB process step, and to add a trigger and outcome.

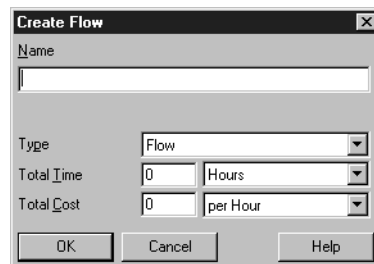
Instructions

1. Click on the Create Flow toolbar button.



2. Click on the ADD NEW PRODUCT TO THE MENU process step.
3. Click on the PUBLISH MENU ON THE WEB process step.

The Create Flow dialog box appears.



4. Click on OK to close the dialog box and add the flow to the diagram.

5. Click on the Create Trigger toolbar button.



6. Click on the ADD NEW PRODUCT TO THE MENU process step.
The Create Trigger dialog box appears.

7. In the Name box, type NEED FOR MENU CHANGE and click OK.

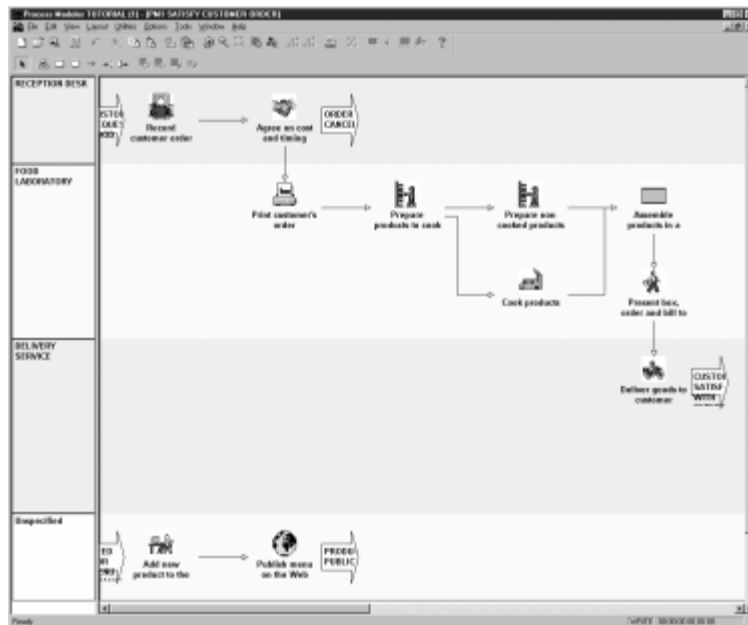
8. Click on the Create Outcome toolbar button.



9. Click on the PUBLISH MENU ON THE WEB process step.
The Create Outcome dialog box appears.

10. In the Name box, type PRODUCT PUBLICISED and click OK.

The diagram now appears as shown.



11. Select File → Save Diagram to save the diagram.

12. Select File → Exit to close the Process Modeler.

Lesson summary

In this lesson you used the Process Modeler to enter new steps called ADD NEW PRODUCT TO THE MENU and PUBLISH MENU ON THE WEB in the Satisfy Customer Order process. You indicated an information flow and recorded a trigger and outcome.

Lesson 2

Create entities and add relationships

Any application you develop will store and manipulate information, so an essential part of developing a system is constructing an accurate model of the information needs of an organization. The information model will act as a framework for the development of new or enhanced systems.

One technique used to model an organization's information requirements is entity relationship modeling. Entity relationship modeling involves identifying the things of importance in an organization (entities), the properties of those things (attributes) and how they are related to one another (relationships).

The Entity Relationship Diagrammer, one of the Oracle Designer tools, enables you to create an entity relationship model in the form of a diagram showing the elements of information and the way they are logically related.

An entity relationship diagram has already been created for Speedy Pizza, modeling the information needs of the business. During this lesson, you will:

- enter a new entity, PRODUCT
- specify the relationship between the PRODUCT entity and the PRICED PRODUCT entity.
- define the attributes of PRODUCT

Create an entity

An entity is a thing of significance, whether real or conceptual, about which the business being modeled needs to hold information.

Objective

To add an entity representing PRODUCT to the entity relationship diagram modeling the order processing area of the business.

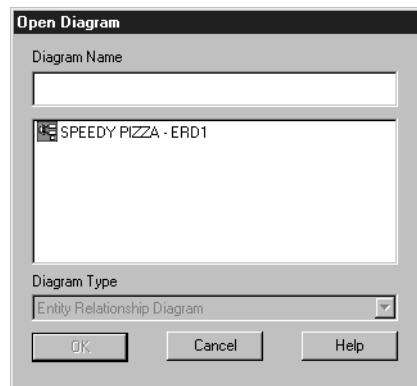
Instructions

1. Click on the Entity Relationship button on the Oracle Designer front panel to start the Entity Relationship Diagrammer.

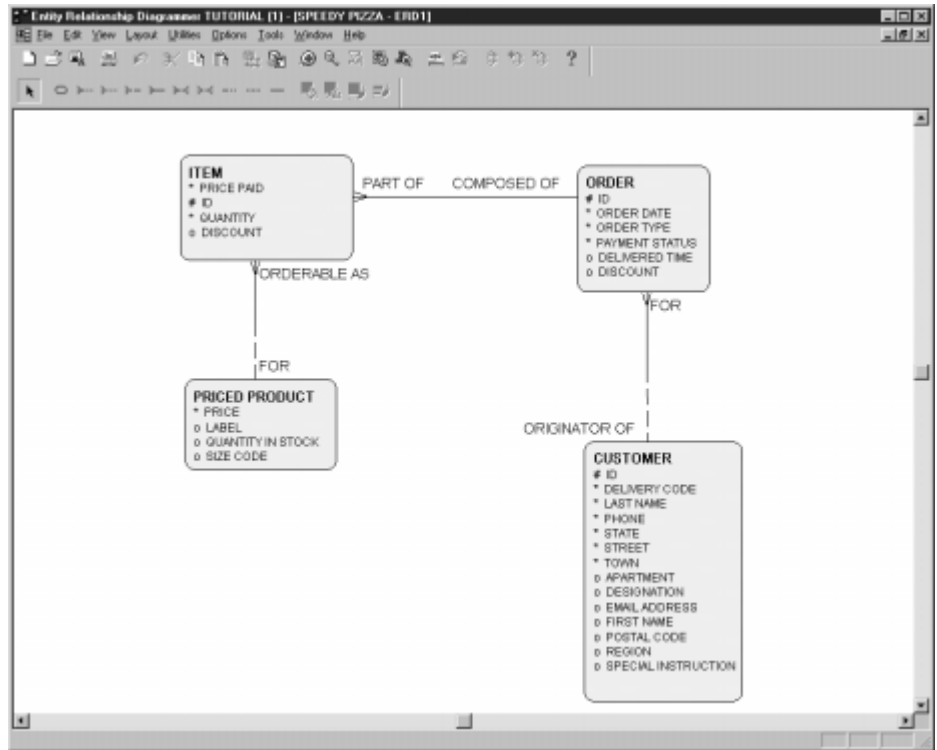


Entity
Relationship
Diagrammer

2. Select File → Open to display the Open Diagram dialog box.



3. Double click on SPEEDY PIZZA - ERD1 to open the diagram.



Hint: You might have to resize the window to view all the entities.

4. Click on the Entity toolbar button.



5. Move the mouse pointer to below the PRICED PRODUCT entity and click to create the entity.

The Create Entity dialog box appears.

Name	<input type="text"/>
Short Name	<input type="text"/>
Plural	<input type="text"/>
OK Cancel Help	

6. Enter the following details in the Create Entity dialog box:

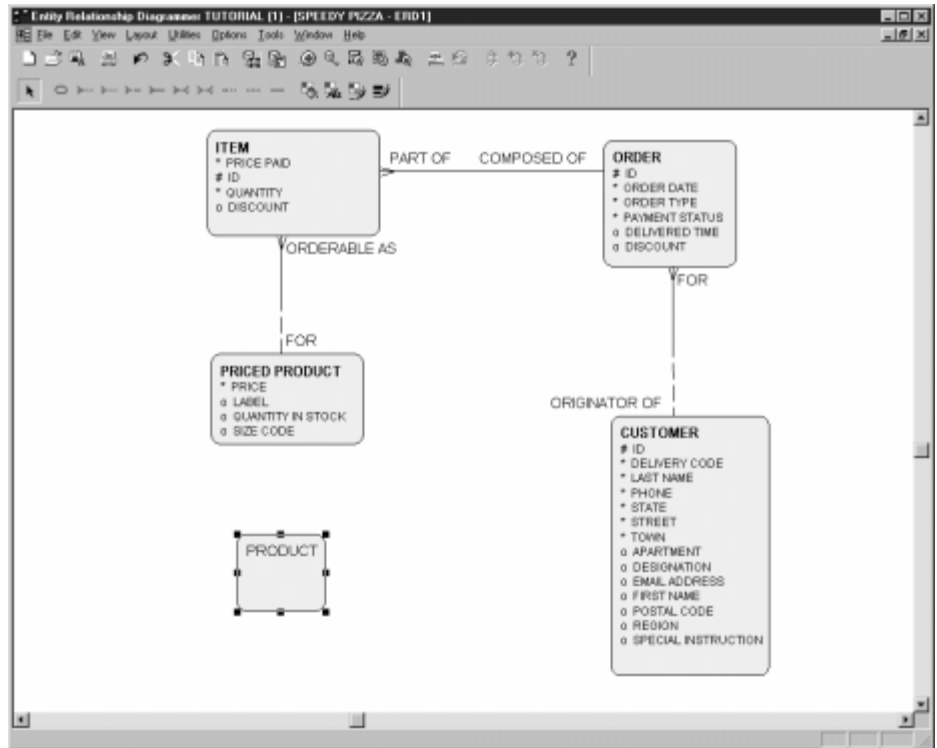
Name: PRODUCT

Short Name: PDT

Plural: PRODUCTS

7. Click on OK to close the Create Entity dialog box.

The PRODUCT entity appears on the diagram.



Hint: You might have to resize the PRODUCT entity.

Add a relationship

Relationships are named significant associations between two entities. Relationships on an entity relationship diagram display graphical indications of their optionality (whether a relationship is mandatory or optional) and cardinality (the number at each end of a relationship).

Objective

To create a relationship between the PRICED PRODUCT entity and the PRODUCT entity.

Instructions

1. Click on the Many-to-One (Mandatory to Optional) Relationship toolbar button.



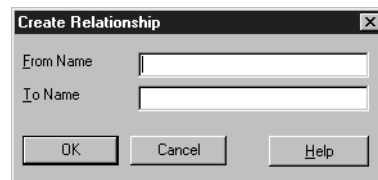
Note: It is important to click on the correct toolbar button. Check that the tooltip is "M:1 (M to O) Relationship" and that the text in the status bar is "Creates a M:1 (mandatory to optional) relationship".

2. Click on the PRICED PRODUCT entity.

A line starts from the edge of the entity that you clicked on and stretches to wherever you move the mouse pointer.

3. Click on the PRODUCT entity.

A line appears between the PRICED PRODUCT entity and the PRODUCT entity, and the Create Relationship dialog box appears.

A dialog box titled "Create Relationship" with a close button (X) in the top right corner. It contains two text input fields: "From Name" and "To Name". Below the fields are three buttons: "OK", "Cancel", and "Help".

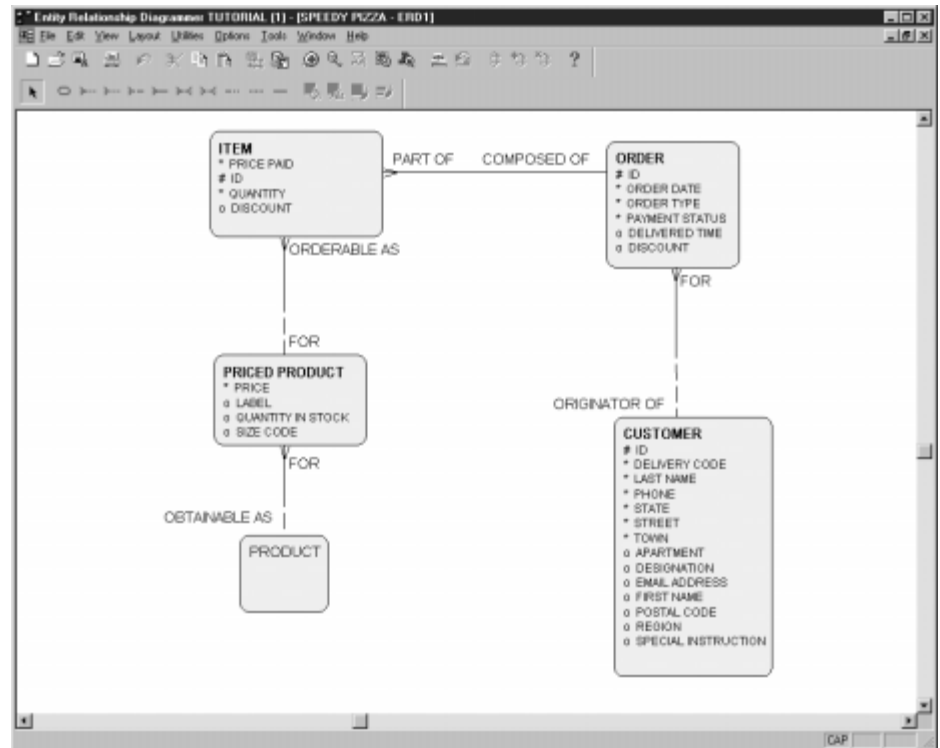
4. In the Create Relationship dialog box, enter the following details:

From Name: FOR

To Name: OBTAINABLE AS

5. Click on OK to close the Create Relationship dialog box.

The names of the two ends of the relationship appear on the entity relationship diagram.



6. Select File → Save Diagram to save the entity relationship diagram.

Create attributes

An attribute is any detail that serves to qualify, identify, classify, quantify or express the state of an entity.

A unique identifier is a combination of attributes and/or relationships that uniquely identifies an occurrence of an entity.

Objective

To add attributes to the new PRODUCT entity.

Instructions

1. Double click on the PRODUCT entity to display the Edit Entity dialog box.

The screenshot shows the 'Edit Entity - PRODUCT' dialog box. The 'Attributes' tab is active. The 'Short Name' field contains 'PDT', 'Name' contains 'PRODUCT', and 'Plural' contains 'PRODUCTS'. The 'Owned' checkbox is checked. The 'Volume' section has four empty input fields for 'Initial', 'Maximum', 'Average', and 'Growth Rate'. The 'Datawarehouse Type' dropdown is set to '<null>'. At the bottom are 'OK', 'Cancel', 'Apply', and 'Help' buttons.

You are now ready to enter the attributes of the PRODUCT entity.

If a particular attribute is always required, the attribute is described as mandatory. If the attribute has a value for only some of the time, it is described as optional. An attribute's Optionality (Opt) property specifies whether the attribute is mandatory or optional.

The type of data represented by an attribute is defined by the attribute's Format property. Some attribute formats require further definition, including length and the number of decimal places (specified by the Max Len and Dec properties respectively).

Attributes can be associated with a domain (a common set of properties that apply to a group of attributes).

2. On the Attributes tab, enter the following details:

Name	Seq	Domain	Opt	Format	Max Len	Dec	Primary
DISCOUNT DAY		DISCOUNT DAY	<input checked="" type="checkbox"/>	CHAR	1		
PRODUCT DESCRIPTION				VARCHAR2	30		
PRODUCT ID				INTEGER	6		<input checked="" type="checkbox"/>
PRODUCT TYPE		PRODUCT TYPE		VARCHAR2	10		

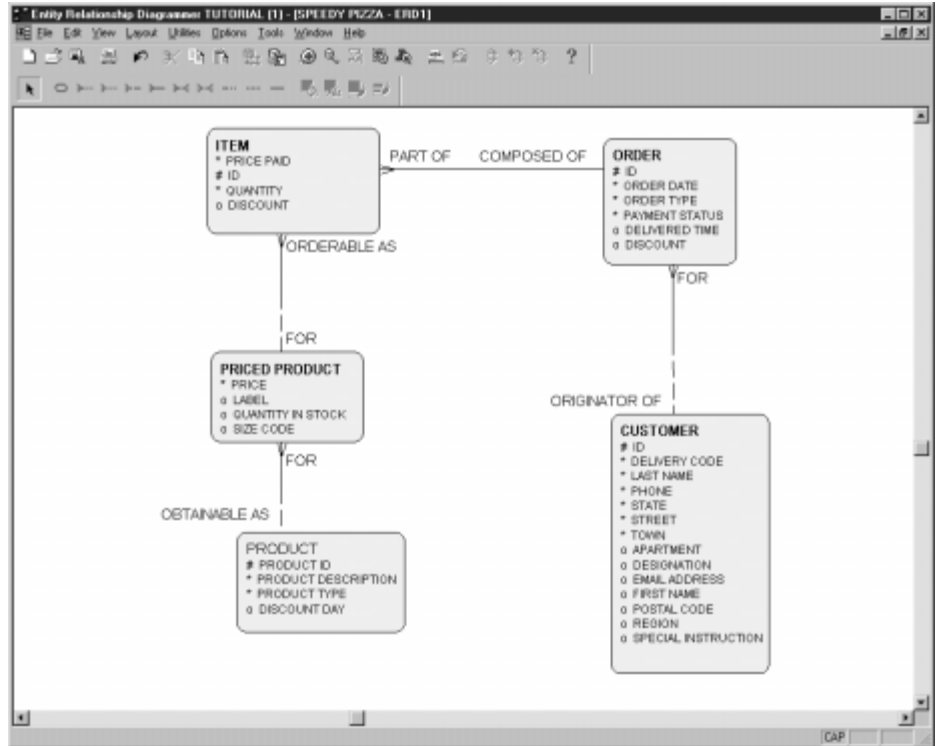
Hint: Use the horizontal scroll bar to see the Format, Max Len, Dec and Primary fields.

3. Double check that you have entered all of the attribute details correctly (this is important for the rest of the tutorial).

Each occurrence of an entity can be uniquely identified by one or more of its attributes. Because you have checked Primary against ID, that attribute will be used as the unique identifier (UID). You can confirm this on the UIDs tab. You could also use the UIDs tab to change the composition of the UID or to create alternate UIDs. For this tutorial, we will not make these changes.

4. Click on OK to close the Edit Entity dialog box.

The attributes for the PRODUCT entity are displayed on the diagram.



Hint: You might have to resize the PRODUCT entity to see all the attributes.

5. Select File → Save Diagram, then select File → Exit to close the Entity Relationship Diagrammer.

Lesson summary

In this lesson you used the Entity Relationship Diagrammer to enter a new entity, PRODUCT. You specified the relationship between the PRODUCT entity and the PRICED PRODUCT entity. You defined the PRODUCT entity's attributes.

Lesson 3

Refine default functions

Functions are arranged in a hierarchy. A function hierarchy shows the information used by functions, in terms of entities and attributes. Each function in the hierarchy is decomposed until the lowest level functions (called elementary business functions) that comprise it have been identified. It is the elementary functions that become the forms, reports and utilities in the finished system.

You use the Function Hierarchy Diagrammer to decompose functions, define elementary functions and show how functions use data.

Identify functions to automate

The process model showed the process steps for the business carried out by Speedy Pizza. All of the process steps were recorded in the repository as functions, but only some of these functions are suitable for automation (e.g. "Record customer order"). The remaining functions (e.g. "Assemble products in a box") will be implemented as manual processes

Objective

To identify which of the existing functions are to be automated.

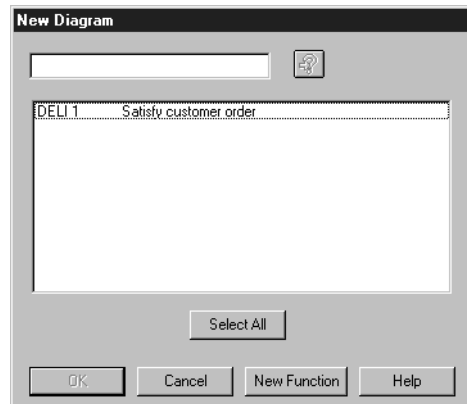
Instructions

1. Click on the Function Hierarchy button on the Oracle Designer front panel to start the Function Hierarchy Diagrammer.



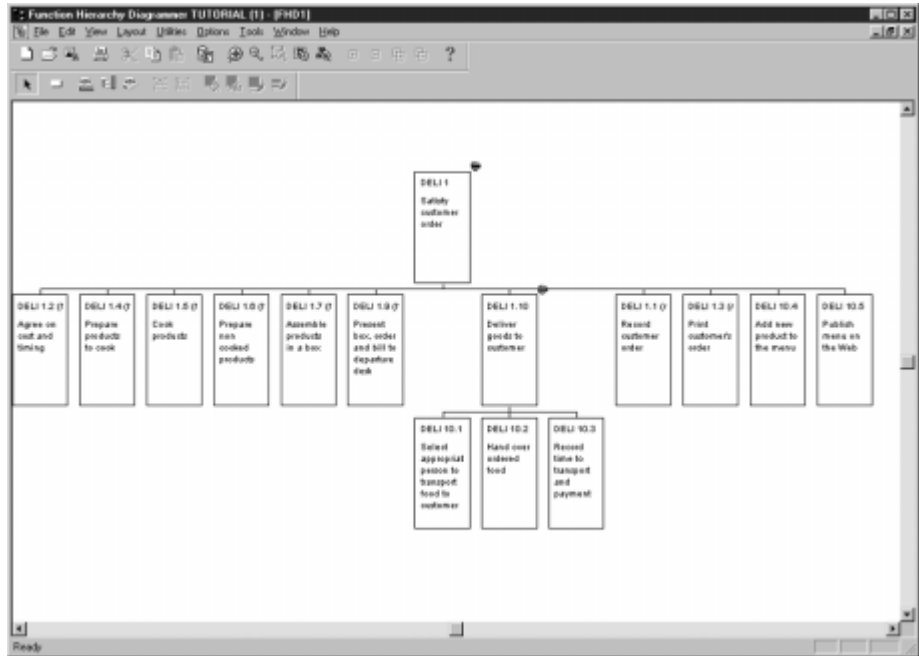
2. Select File → New to create a new diagram based on existing function definitions.

The New Diagram dialog box appears.



3. Highlight "DELI 1 Satisfy Customer Order" in the list and click OK.

The DELI 1 SATISFY CUSTOMER ORDER function and its child functions appear.

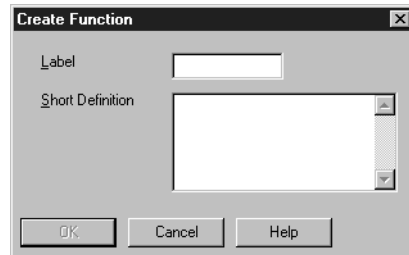


Hint: You might have to resize the window or use zoom to view the entire function hierarchy.

4. Click the Function button in the toolbar.



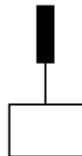
5. Click in an empty part of the diagram. This opens the Create Function dialog box.



6. In the Label box, type `AUTO 1` and in the Short Definition box type in `Handle automated functions`. Click the OK button. The `AUTO 1` function appears on the diagram as a root function.

This new root function can now be used as the parent of functions that already exist on the diagram. To do this, you drag the child functions and drop them onto the root function. It is important when doing this that you position the mouse pointer precisely in the centre of the root function before releasing the mouse button. Otherwise, the moved function will become another root function. The mouse pointer changes shape to help with this.

7. Drag the Record Customer Order function (i.e., `DELI 1.1 (AUTO)`) to the centre of the `AUTO 1` function until the mouse pointer shape changes to that shown below.



Releasing the mouse button reparents `DELI 1.1 (AUTO)` Record Customer Order under the `AUTO 1` root function.

8. Using the technique described in step 7, above, reparent the following additional functions under AUTO 1:

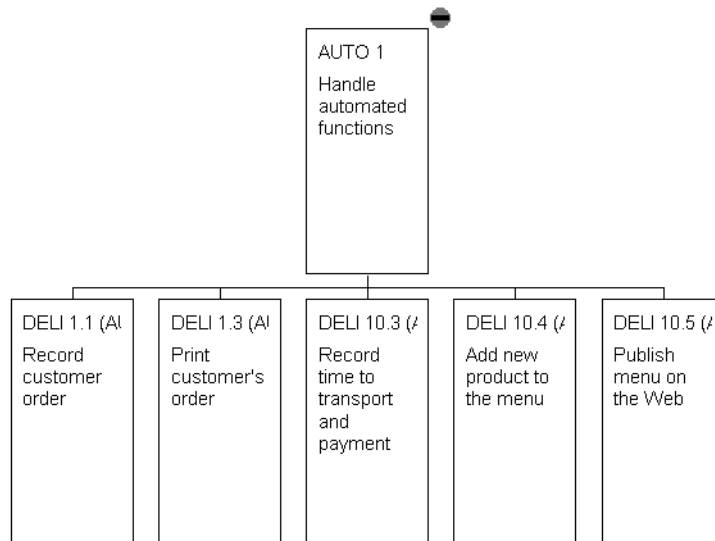
DELI 1.3 (AUTO) Print customer's order

DELI 10.3 (AUTO) Record time to transport and payment

DELI 10.4 (AUTO) Add new product to the menu

DELI 10.5 (AUTO) Publish menu on the Web

The new function hierarchy for the AUTO 1 module is as follows:



Add entity usages to a function

To create working applications, the information used by a function must be identified. This typically involves specifying whether a function can create, retrieve, update or delete an instance of an entity.

Objective

To specify how the PRODUCT entity is used by the "Add new product to the menu" and "Publish menu on the Web" functions.

Instructions

1. Double click the function DELI 10.4 (AUTO) Add new product to the menu. This opens the Edit Function dialog box.

Error! Not a valid filename.

2. On the Entity Usages tab, add the PRODUCT and PRICED_PRODUCT entities to the Entity column (by selecting them from the drop down list in each cell of the Entity column) and set their usages as set out in the following table.

Entity	Create	Retrieve	Update	Delete	Archive	Other
PRODUCT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
PRICED_PRODUCT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

The usage flags indicate how the function uses the information represented by the entities.

Hint: Use the horizontal scrollbar to see the Delete, Archive and Other usage flags.

3. Click on OK to close the Edit Function dialog box.
4. Set up entity usages for entities DELI 10.5 (AUTO) as below.

Entity	Create	Retrieve	Update	Delete	Archive	Other
PRODUCT		<input checked="" type="checkbox"/>				
PRICED_PRODUCT		<input checked="" type="checkbox"/>				

5. If you wish, save the diagram with a name of your choice. Note that you do not have to save the diagram in order to save the data that you have input in this lesson.

Create attribute usages using the function/attribute matrix utility

Having specified the entities used by a business function, and how those entities are used, you specify how the function uses the attributes of those entities, including whether a function can:

- create or insert a value into an optional attribute
- retrieve the value of an attribute
- update the value of an attribute
- delete the value of an optional attribute

You can use the Function/Attribute Matrix utility to create usages for all attributes of the entities used by a function.

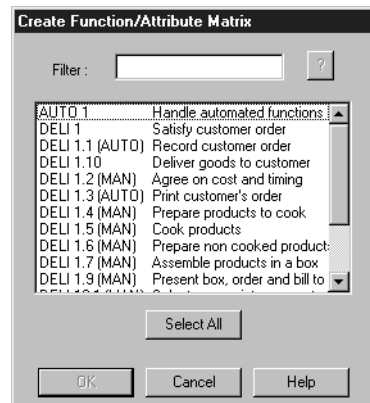
Objective

To create attribute usages for the "Add new product to the menu" and "Publish menu on the Web" functions.

Instructions

1. In the Function Hierarchy Diagrammer, select Utilities>Function/Attribute Matrix.

This displays the Create Function/Attribute Matrix dialog box.



2. Highlight the following functions in the list:

DELI 10.4 (AUTO) Add new product to the menu

DELI 10.5 (AUTO) Publish menu on the Web

3. Click the OK button.

The Create Function/Attribute dialog box closes and the attribute usages are created.

To confirm that the attribute usages have been created, open the Edit Function dialog box for the functions (see previous section) and view the Attribute Usages tab. For each entity used by the function, the attributes will be listed and the usages shown by completed check boxes. If you scroll across to the Comments column, you will see "Created by system on <date>".

4. Close the Function Hierarchy Diagrammer.

Lesson summary

In this lesson you identified which functions should be automated and created entity and attribute usages.

Lesson 4

Transform the data model into a table design

When you record an entity model in the Repository, you are specifying the things the business needs to manipulate and about which information has to be stored. The next stage is to design a database in which to hold this information.

The Database Design Transformer, one of the Oracle Designer tools, enables you to create and maintain a database design based upon the entity model recorded in the Repository. The resulting database design comprises a number of database object definitions including table definitions (to record instances of entities), column definitions (to store attributes), and foreign key constraint definitions (to implement relationships between entities). The Database Design Transformer also creates and maintains additional elements to support the design.

Like the entity model, the database design generated by the Database Design Transformer is stored in the Repository. Later in the tutorial, you will use other Oracle Designer tools to create the database itself, based on this design.

During this lesson, you will use the Database Design Transformer to create new table and column definitions in the Repository from the entity and attributes that you defined in the previous lesson (and from the supplied entities).

Create a table definition from entities

The Database Design Transformer maps entities to tables to create table definitions, and creates column definitions from attributes.

Note: The instructions below direct you to start the Database Design Transformer from the Entity Relationship Diagrammer. However, you can also start the Database Design Transformer directly from the Oracle Designer front panel using the Database Transformer button.

Objective

To create table definitions in the Repository from the design information on the entity-relationship diagram.

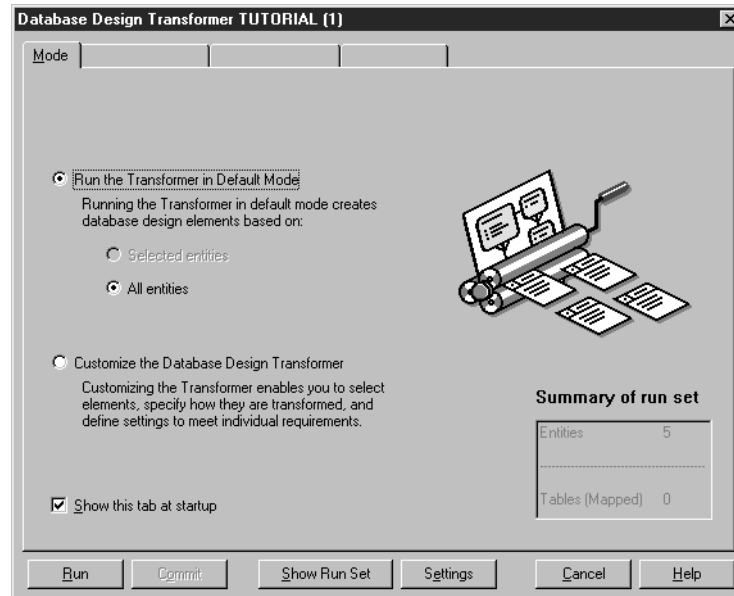
Instructions

1. In the Entity Relationship Diagrammer, open the SPEEDY PIZZA - ERD1 diagram.

Note that the PRICED PRODUCT entity has no unique identifier (i.e., no attribute marked with a #). This will be created later by the Database Design Transformer.

2. Select Utilities → Database Design Transformer.

A progress box, then the Database Design Transformer dialog box, appears.



3. Ensure that the "Run the Transformer in Default Mode" and "All entities" options are selected.
4. Click on Run to run the Database Design Transformer.

The Design Transformer Output window appears, detailing the operations performed by the Database Design Transformer.

The Database Design Transformer creates table and column definitions from the entities and attributes. The name of the table definition is the plural version of the entity name, with spaces converted to underscores. The names of the column definitions are the attribute names with spaces converted to underscores.

In the next lesson, you will use the Design Editor to view the definitions created by the Database Design Transformer.

5. Close the Design Transformer Output window and the Database Design Transformer dialog box.
6. Select File → Exit to close the Entity Relationship Diagrammer.

Lesson summary

In this lesson you used the Database Design Transformer to create new table and column definitions in the Repository from entities and attributes.

Lesson 5

Refine the default database design

The database design created by the Database Design Transformer from the entity relationship model consists of a set of interrelated tables, columns and constraints. This is often referred to as the 'first-cut database design'. The design objects that make up the first-cut database design can be represented on one or more server model diagrams and refined to a level ready for implementation.

After you have produced the first-cut database design, you can begin to make improvements to the design itself, and to establish defaults for how generated applications will use data. The process of amending and enriching the database design is one of the most important tasks in the application system development lifecycle.

The Design Editor, one of the Oracle Designer tools, enables you to model database designs on server model diagrams. Each server model diagram represents a collection of database objects that can be distributed physically between one or more databases. Using the Design Editor, you can add more information to the first-cut design produced by the Database Design Transformer.

During this lesson, you will refine the definition of tables that you created in the previous lesson.

Modify a default table definition

The Database Design Transformer has converted the entity relationship model into design-level table and constraint definitions held in the Repository.

To prepare the default design for implementation, you now add extra information that was not relevant during the analysis phase of development.

Objective

To refine the definition of tables that you created in the previous lesson.

Instructions

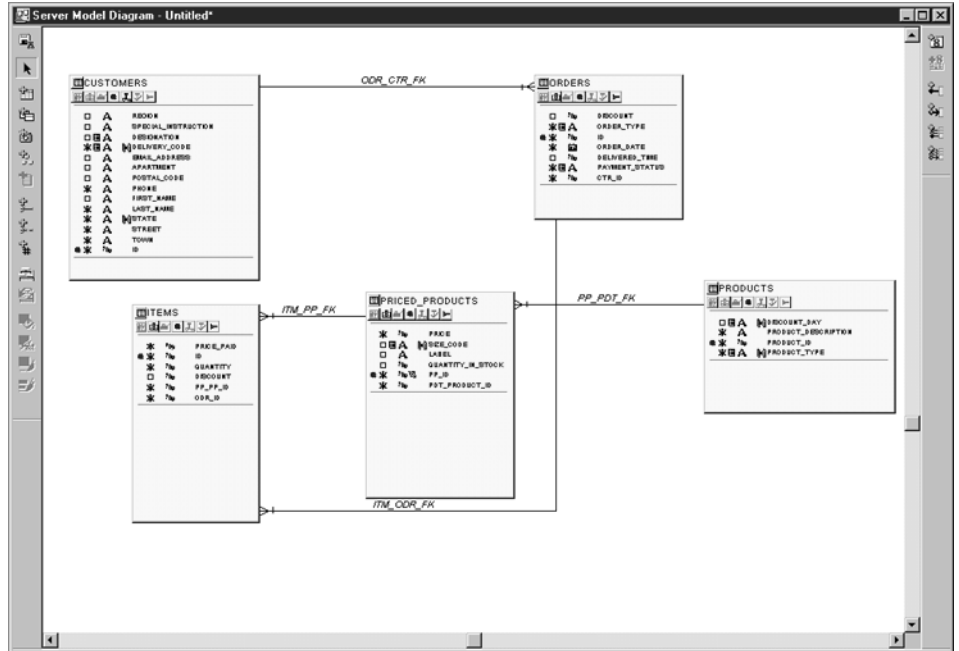
1. Click on the Design Editor button on the Oracle Designer front panel to start the Design Editor.



2. If the Welcome To The Design Editor dialog box opens, select the Server Model option, uncheck the Use A Guide and Open A Diagram boxes, and click the OK button.
3. Click on the plus sign next to Relational Table Definitions. This reveals the list of relational table definitions that were mapped from entities by the Database Design Transformer.
4. Highlight all of the relational table definition names listed under Relational Table Definitions (CUSTOMERS, ITEMS, ORDERS, PRICED_PRODUCTS, PRODUCTS), then right-click on one of the highlighted names and choose Show On New Diagram.

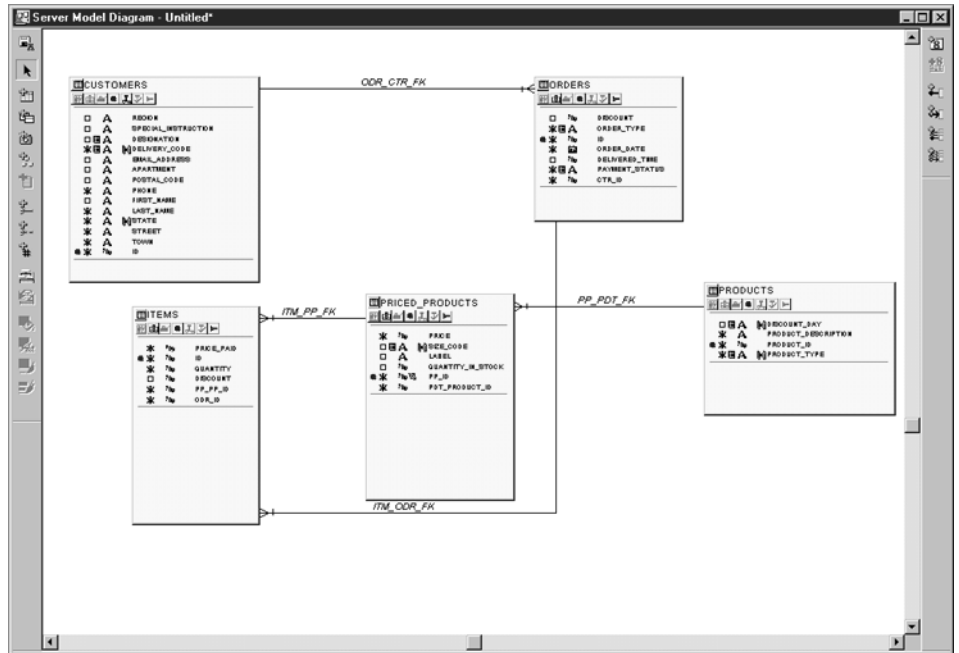
An alternative method is to drag and drop the relational table definition names into the blank area of the Design Editor window.

A server model diagram is created in a separate window within the Design Editor. You may have to zoom out to see all of the diagram.



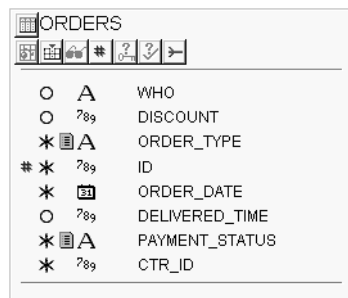
Note that PRICED_PRODUCTS now has a primary key (PP_ID).

- On the server model diagram, select the diagram object representing the ORDERS table, then click the (create) Column button on the left side toolbar.

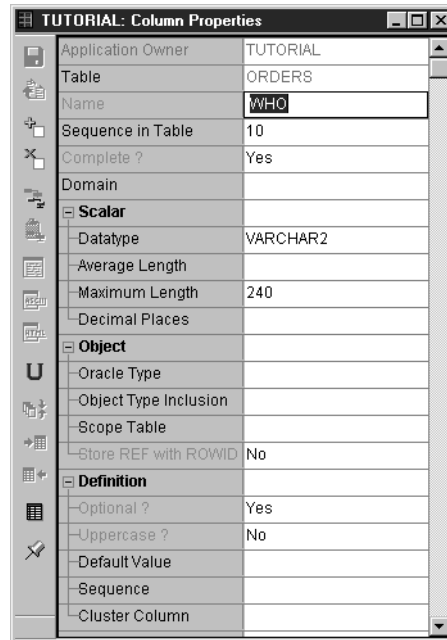


The following steps will add a column that could be used to record which user created the order. This is to familiarize you with the tool and will not be part of the form that you generate in later lessons.

- On the first line of the Fast Create Columns dialog box, type WHO and click OK. The new column name appears in the diagram object representing the ORDERS table.



7. Switch to using the property palette by choosing Options → Use Property Palette (this option may already be the current one).
8. On the diagram object representing the ORDERS table, double click the **A** icon next to the name of the new column (WHO). You will see the column properties of the new column displayed in the Property Palette.



The property names for a particular object are listed in the cells in the left column. The values for the properties are listed in the cells in the right column. When you click on a values cell, it changes to an appropriate input control. The controls you will be using are the text box and the drop-down box.

9. Change the Maximum Length property to 20, then scroll down to AutoGen Type and change its value to "Created By".
10. Save the data by clicking the Save button in the Property Palette toolbar.



Keep the Design Editor running, ready for the next lesson.

Lesson summary

In this lesson you used the Design Editor to refine the definition of tables that were created by the Database Design Transformer.

Lesson 6

Build a database

Having refined the database design, you use the Server Generator to generate the server-side components for the system.

The Server Generator can create objects on a local or remote database from definitions recorded in the Repository.

During this lesson you will:

- create the database tables on a local Oracle database

Create the database tables

Oracle Designer can take the table definitions created by the Database Design Transformer and create the database that your applications will use to store data.

Objective

To create the database tables required for your system.

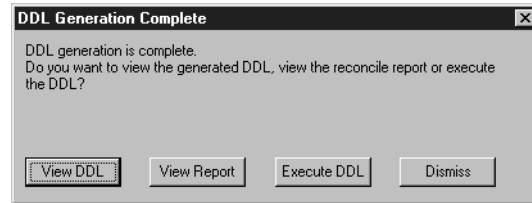
Instructions

1. In the Design Editor, select **Generate** → **Generate Database From Server Model**. This opens the **Generate Database From Server Model** dialog box.
2. On the **Target** tab, choose the **Database** option button and enter the connection details (the same as for your current connection).
3. On the **Objects** tab, move all objects into the **Generate** list by clicking the double-right-arrow button.



4. Click the Start button. The messages window is opened in the Design Editor. This reports the progress of the generation process.

When generation is complete, the Generation Complete dialog box opens.



5. Click the Execute DDL button. This creates the database tables and other objects.

After the 'Processing Complete' message appears in the message window, you can use the Database Navigator to see the database objects that have been created.

You open the Database Navigator by selecting Tools → Database Navigator. Note that no triggers have been created for any of the tables. You should close the Database Navigator before continuing.

You can close the Design Editor. There is no need to save the server model diagram that you created, but you may do so if you wish.

Lesson summary

In this lesson you used the Server Generator to create the tables on the database.

Lesson 7

Create default working forms

Once the analysis data is as complete as possible and the database design is stable and complete, you are ready to create a first-cut application design based on information stored in the Repository.

The Oracle Designer utility to produce the first-cut application design is the Application Design Transformer. The Application Design Transformer converts functions into candidate modules, which can then be implemented as forms, reports and utilities. The Transformer also converts function data usages to module data usages that define how a module uses the tables and columns in the database.

The modules created by the Application Design Transformer are 'candidate' modules and are not included in the application design until you accept them (by changing their 'Candidate' property to 'No'). Modules you accept, and their usages, can then be used for generating applications.

Generate default module definitions

Objective

To create module definitions for the automated functions.

Instructions

1. Click on the Function Hierarchy button on the Oracle Designer front panel to start the Function Hierarchy Diagrammer.



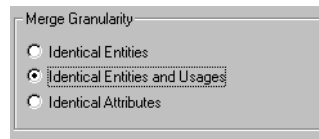
2. Create a new diagram based on the function AUTO 1 "Handle automated functions".

You do this by selecting File → New, then double clicking on AUTO 1 in the New Diagram dialog box. You may have to resize the diagram to see all of the functions.

3. On the Function Hierarchy Diagram, select the function AUTO 1 "Handle automated functions".
4. Select Utilities → Application Design Transformer.
This opens the Application Design Transformer dialog box.
5. Ensure that the following options are set (these should be the default settings):

Common Parameters - Start Function	AUTO 1
Common Parameters - Module Prefix	TUTORI

6. Set the Merge Granularity option to Identical Entities And Usages.



Note: If you leave this option set to the default (i.e., Identical Entities), the Application Design Transformer will create a single module definition for functions DELI 10.4 (AUTO) and DELI 10.5 (AUTO). This single module definition will not enable you to continue with the tutorial.

7. Click the Generate button.
8. At the end of the generation process (wait for the percentage completion figure to reach 100%), close the progress box and the dialog box, then exit from the Function Hierarchy Diagrammer. You can save the diagram if you wish, but there is no need to, because all the underlying data was saved in the Repository as you worked.

Accept the candidate module definitions for generation

The Application Design Transformer has created "candidate" module definitions. These will not be used for generation unless you change this status.

Objective

To make the default module definitions ready for generation.

Instructions

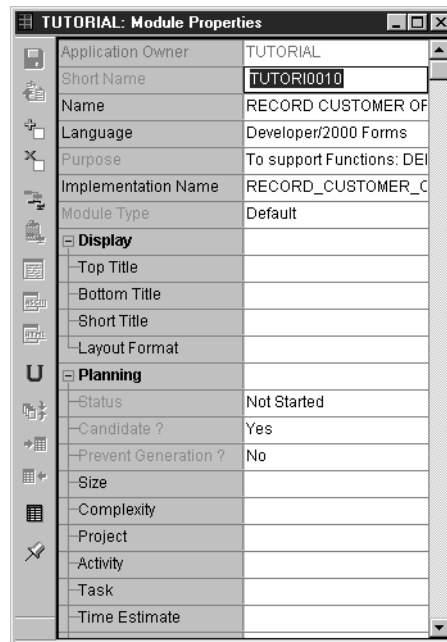
1. Click on the Design Editor button on the Oracle Designer front panel to start the Design Editor.



2. Choose the Module Applications option on the Welcome screen, then click the OK button. (If the Welcome screen did not appear, change directly to the Modules tab in the Navigator window.)
3. Select Options → Use Property Palette (the option may already be chosen), then expand the list of Modules in the Navigator and double click on the icon for the first module in the list.




The properties for the module are shown in the Property Palette.



The property names for a particular object are listed in the cells in the left column. The values for the properties are listed in the cells in the right column. When you click on a values cell, it changes to an appropriate input control. The controls you will be using include the drop-down box and the list box. When using the list box, you obtain the list of possible values by clicking the icon on the right of the cell.

The changes that you make to the values of a particular object (in this case module definitions), will be saved automatically if you display the values for another object. You can also save the values manually.

To save the changes that you make on the Property Palette manually, select File → Save Uncommitted Data, or click  on the Property Palette toolbar.

4. Under the heading "Planning", change the Candidate? property to No, then save the data manually (as described above).

The icon for the first module changes to represent a Form.



5. Change the Candidate? properties of all the other modules to No.

The icons for the modules will change to represent various module languages.

6. Display the properties for module TUTORI0050 "Publish menu on the Web", change the language to Oracle WebServer, then save the change.

Leave the Design Editor open for the next stage.

Generate a working form

You can now generate a first-cut working form. The generated form will provide basic functionality that you can refine in later lessons.

Objective

To generate a default working form from a module definition.

Instructions

1. In the Navigator, highlight module TUTORI0040 ("Add new product to menu").
2. Select Generate → Generate Module. This opens the Generate Form dialog box.

The screenshot shows the 'Generate Form' dialog box with the following details:

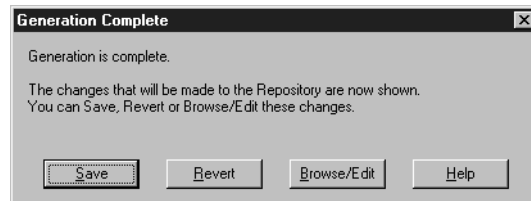
- Module:** TUTORI0040
- Generate Option:** Form (selected from a dropdown menu)
- Options:**
 - Ignore User Application Logic
 - Include Module Network
 - Generate Associated Menus
 - Generate Attached Libraries
- Preserve Layout:**
 - No
 - Yes
- Source Form Name:** TUTORI0040.fmb (with 'Browse' and 'Reset' buttons)
- Standards:**
 - Object Library Name:** D:\orant\CTUTX70\TUTORIAL\Tutorial.olb (with 'Browse...' and 'Reset' buttons)
 - Template Name:** D:\orant\CTUTX70\TUTORIAL\Tutorial.fmb (with 'Browse...' and 'Reset' buttons')
- Buttons:** Start, Options..., Cancel, Help

3. In the Standards group on the Generate Form, ensure that the Object Library Name box is set to find tutorial.olb and that the Template Name box is set to find tutorial.fmb. If they are not already set, click each Browse button to find them.

Note: The tutorial.olb and tutorial.fmb files are copies of the files ofgstnd1.olb and ofgpc1t.fmb files, as they were originally supplied with Oracle Designer. If you use modified versions of these files, or alternate files, you will not get the results described in the remainder of this tutorial.

4. Click the Options button to open the Form Generator Options dialog box, and on the Run tab set the Run Form option to Yes. Click the OK button.
5. On the Generate Form dialog box, click the Start button.

Progress is reported in the message window. Blue messages will tell you about changes that the generator intends to make to the module definition. The Generation Complete message box is then displayed, giving you the option to save the changes to the module definition or to revert to the pre-generation state.



Click the Save button in the toolbar to save the information to the database.

Click the Clear Record button and make further entries if you wish (e.g., for a beverage), but not too many at this stage.

Now click the Price field and enter 9.99.

Tab to the Size Code field. Enter M (a list of values is available).

Tab to Label and type in Medium Ham and Mushroom Supreme.

Tab to Quantity In Stock but don't enter a value.

Tab to Product Id. Enter 1001 from the list of available values.

Click the Save button in the toolbar. The message "Transaction complete - 1 record applied and saved" is shown at the foot of the window.

Make a few more entries, if you wish.

You will have noticed that the fields on the form are held within two blocks, one relating to the Priced Products table and one to the Products table. In the next lesson, the Products table will be made into a lookup table and you will enter only Priced Products information through the generated form.

7. To close the generated form, press ALT+F4.

The generated form is closed.

Leave the Design Editor open for the next lesson.

Lesson summary

In this lesson you:

- created default module definitions
- modified the default module definitions to make it possible for them to be used in generation
- generated a working form based on a default module definition.

Lesson 8

Refine modules for generation

In lesson 7 you generated (and possibly used) a form that had been given default characteristics based on design information supplied earlier. You can now change the default characteristics to meet your further requirements. One way to do this is to use the Navigator to find the object that you want to change (e.g., the Discount Day field), then open the Property Palette and make the changes. Another way is to open a diagram so that the objects can be easily seen, then to open the Property Palette from the diagram. The latter method is described in this lesson.

Change form characteristics

By default, the form generator will provide a button on the toolbar for opening a list of values for a field on a form. As an alternative, you can provide a list of values by making the field a drop-down list.

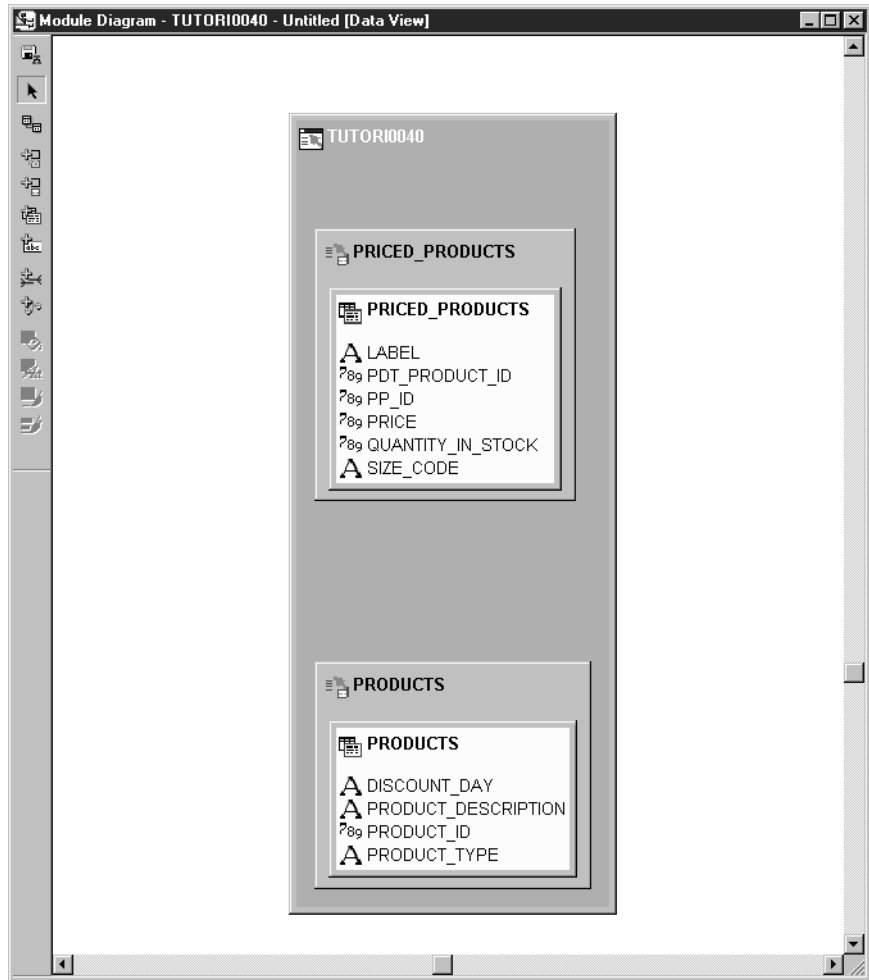
An automatically generated attribute is given a name that might not be meaningful when used as a prompt on a generated form. You can change such prompts to a more meaningful one.

Objective

To set up a drop-down list and change a field prompt on the Priced Products form.

Instructions

1. In the Design Editor, create a module diagram for the TUTORI0040 module. You do this by dragging TUTORI0040 from the list of modules in the Navigator to the blank work area of the Design Editor.



2. Display the Property Palette for the SIZE_CODE bound item property, in one of the following ways.

If the Property Palette is already open, you display the correct properties by single clicking SIZE_CODE on the diagram.

If the Property Palette is not already open, you open it by selecting

Options → Use Property Palette then double clicking the **A** icon next to SIZE_CODE.

The Property Palette title bar will say "TUTORIAL: Bound Item Properties".

3. Scroll down to the Display Type property and change it from "Text" to "Pop list", then save the change by clicking the Save button on the Property Palette toolbar.
4. Display the Property Palette for PP_ID. In the Display section, change Display? to Yes and change the Prompt text to "Priced Products Id". Save the changes.

Create a lookup table

In the default generated form that you have already seen, the fields for inputting Products data appeared on the Priced Products form. Now we are going to assume that Speedy Pizza will store product data on the database as a separate operation, and that the "Add new product to the menu" process uses this information to create specific "menu" items. For example, Speedy Pizza has a basic product called "Ham and mushroom supreme", one of its pizzas. This is available in small, medium and large sizes, and each of these variants must appear on its menu. This is achieved through the Priced Products form, by creating separate records for small, medium and large pizzas, all incorporating "Ham and mushroom supreme" from the Products lookup table.

Objective

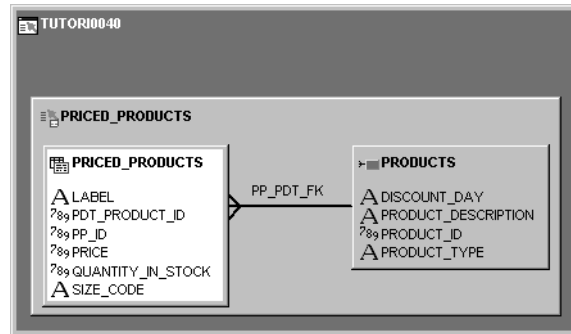
To change the Products table into a lookup table.

Instructions

1. On the module diagram for TUTORIO0040, drag the PRODUCTS table usage (yellow box) so that its left side overlaps the right side of the PRICED_PRODUCTS table usage.

2. When a broken vertical indicator line appears to the right of the PRICED_PRODUCTS table usage, release the mouse button.

A lookup link appears on the diagram. The crow's foot always points to the table on the left, which is the usage that contains the foreign key column.



Leave the Design Editor open for the next lesson.

Lesson summary

In this lesson you:

- made it possible to choose the values for Size Code from a drop-down list
- changed a field prompt to make it more meaningful
- set up a lookup table.

Lesson 9

Generate a refined Oracle Developer Form

During this lesson, you will make further changes to the default design of the Priced Products form. You will then regenerate the form to incorporate all the changes you have made.

Set preferences to influence generation

The final appearance of a generated form is determined by many of the design decisions and defaults that you have already encountered. There is, however, one very important set of preferences that you have not yet seen. These are the Generator Preferences, and the purpose of this exercise is to make you aware of their potential by changing just one of them. If you do not change the Generator Preferences, a default set will apply in all circumstances and this might be perfectly acceptable to you.

Objective

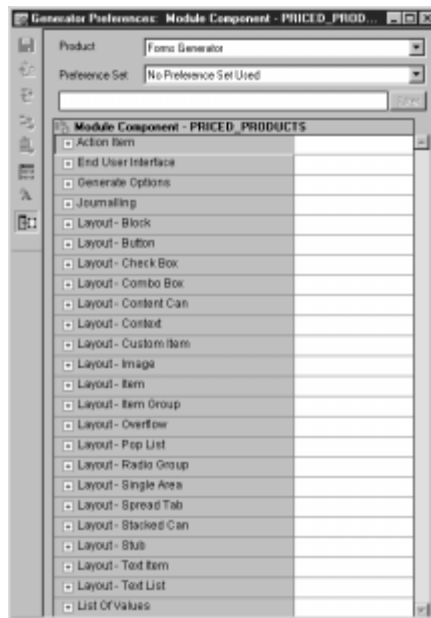
To increase the width of border lines on the generated form.

Instructions

1. On the module diagram for the TUTORI0040 module, select the PRICED_PRODUCTS module (outer, gray box) and click the Generator Preferences toolbar button.



This opens the Generator Preferences palette.



2. Expand the Layout - Block preference node and change the "Block brush width" (BLKBWD) preference to 4.

You can obtain an explanation of a preference by clicking on the Context Sensitive Help button on the toolbar and then clicking on the preference name.

3. Save the preference data by clicking the Save button on the palette toolbar.
4. Close the Generator Preferences palette.

Change a display characteristic of the generated form

So far, you have looked at module diagrams in "data view". There is an alternative, which is "display view". In display view, you can change certain physical aspects of the generated form by direct manipulation of objects on the diagram.

Objective

To change the order of the fields on the generated form so that Product Id is the first.

Instructions

1. Make the Module Diagram for TUTORI0040 active, then select View → Display View.

The diagram appears as follows:



2. With the mouse pointer, grab each of the module component items in turn and move them up or down the list, into the following order:

Product Id

Product Id

Product Description

Product Type

Discount Day

Size Code

Price

Quantity In Stock

Label

Priced Products Id

Generate, run and use a form

You generated and used a form in a previous lesson, but that was a default design. Now is the time to generate the changes you have made to that design.

This lesson also deliberately gives alternate instructions for procedures you have already followed, so that you can see that there is more than one way to do some things.

Objective

To generate, run and use the refined Products form definition.

Instructions

1. Select Options → Generator Options → Forms. This opens the Form Generator Options dialog box.
2. On the Run tab, set Run Form to No, then click the OK button.
3. Select the TUTORI0040 module in the Navigator or on a Module Diagram.

4. Select Generate → Generate Module.

This opens the Generate Form dialog box. Check that the Object Library Name and Template Name boxes are still set to tutorial.olb and tutorial.fmb.

5. Click the Start button.

The message window is opened in the lower part of the Design Editor. Blue messages tell you what further changes the generator intends to make to the module definition. When generation is complete, the message "Generation of module 'TUTORI0040' complete" appears in the message window and the Generation Complete dialog box opens.

6. On the Generation Complete dialog box, click the Save button.

If you wish, you can close the message window by choosing View → Message Window.

7. Select Run → Forms Runtime.

This opens the Oracle Developer Forms Runtime window and the Forms Runtime Options dialog box.

8. In the File box, type in the name of (or browse to) the generated module (cgenf60\ADD_NEW_PRODUCT_TO_THE_MENU.fmx) , enter your user id, password and the database name, then click OK.

The Priced Products form is displayed, showing the changes that you have made since lesson 7.

The screenshot shows a window titled "Priced Products" with a menu bar and toolbar. The main area contains four identical record templates. Each record has fields for Product Id, Product Description, Product Type, Discount Day, Size Code (a drop-down list), Price, Quantity In Stock, Label, and Priced Products Id. The bottom status bar shows "Enter value for Product Id - list of values available" and "Record: 1/1".

If you wish to use the form and did not enter Product information in the default form on lesson 7, you must now populate the PRODUCTS table using the script provided. You do this by choosing the Populate Tutorial Tables program from the Oracle Designer Tutorial folder. if you are prompted to enter a username, enter your username, password and database connection name in the following format:

```
<username>/<password>@<connection_name>
```

You can now use the form to record priced products data on the database. You will have noticed the thicker border. Notice also that the Product Id field is the first in each record. Also that the Size Code field has a drop-down list.

You can save the data that you enter on the form by clicking the Save button on the toolbar. You can close the form by selecting Action → Exit.

Close the module diagram in the Design Editor, but leave the Design Editor open for the next lesson.

Lesson summary

In this lesson you:

- set a generator preference
- used the display view of a module diagram to change a display aspect of the module definition
- regenerated the Products form to show all the changes that you made.

Lesson 10

Generate a WebServer Application

The WebServer Generator creates fully functional WebServer applications that enable users to query, update, delete and enter information in an Oracle database over the World Wide Web. In this lesson, you learn how to generate basic WebServer applications.

Note: You can follow this lesson only if Oracle's WebServer Generator has been installed. You will need a username, password and connect string for the WebServer, and these may be different to those you have been using for the previous lessons. You will also need to know the full path of a Web browser executable and the Web Agent URL. Your System Administrator will be able to tell you these if WebServer Generator has been set up.

Implement the Speedy Pizza system as a Web application

In lesson 1 you created a process step called "Add new product to the menu". You progressed this process step as part of the Speedy Pizza system and implemented it as a form. In this lesson, you will implement the same process as a Web application.

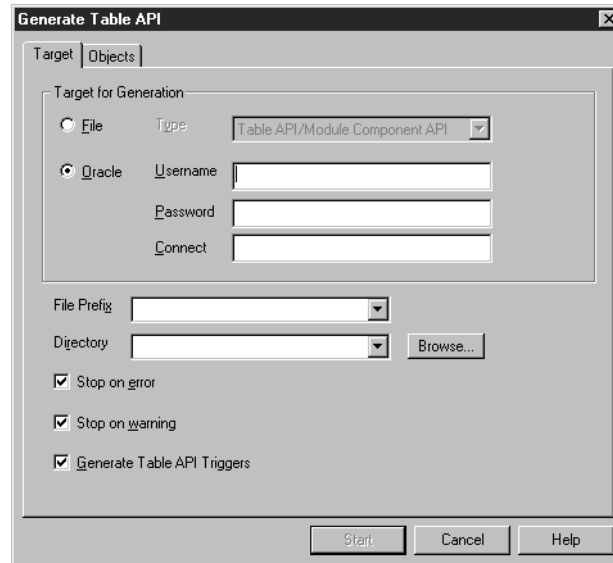
Objective

To implement part of the Speedy Pizza system as a Web application.

Instructions

1. In the Design Editor, in the list of relational table definitions on the Modules or Server Model tabs, highlight all the table names and select Generate → Generate Table API.

This opens the Generate Table API dialog.

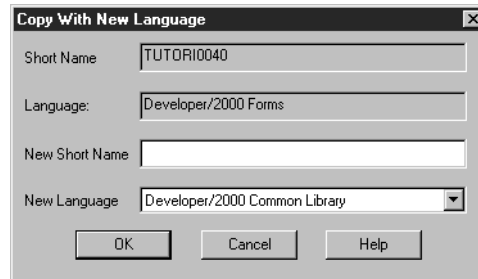


2. Select the Oracle option button and enter the connection details if these are not already shown.
3. Click the Start button.

The Design Editor message window is opened, through which progress is reported.

4. When generation of the table API is complete, close the message window by choosing View → Message Window.
5. In the Navigator, expand the Modules node and highlight TUTORI0040.

6. Select Utilities → Copy With New Language. This opens the Copy With New Language dialog box.



7. Enter TUTORI0060 in the New Short Name box and select Oracle WebServer from the New Language drop-down list, then click the OK button. The new module is listed in the Navigator.

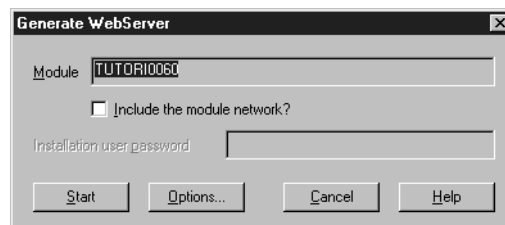
8. Create a module diagram for the TUTORI0060 module.

9. Allow Insert as an operation for all of the items in the Products lookup table.

One way of doing this as follows. Open the Property Palette for one of the items (e.g., DISCOUNT_DAY). Highlight all the lookup items on the diagram (i.e., DISCOUNT_DAY, PRODUCT_DESCRIPTION, PRODUCT_ID, PRODUCT_TYPE). In the Operations section of the Property Palette, change the Insert? value to Yes. Click the Save button on the Property Palette toolbar.

10. Select Options → Generator Options → WebServer and use the connection information supplied by your System Administrator to complete the dialog. Check the "Auto-install after Generation" check box. For a full description of the details to be entered into the Options dialog box, click the Help button. Click the OK button to save the information and close the dialog box.

11. Highlight module TUTORI0060 in the Navigator (or on the module diagram) and select Generate → Generate Module. This opens the Generate WebServer dialog box.



12. Click the Start button.

Progress is reported in the message window.

13. When the "Generation is complete" message box is displayed, click the OK button.

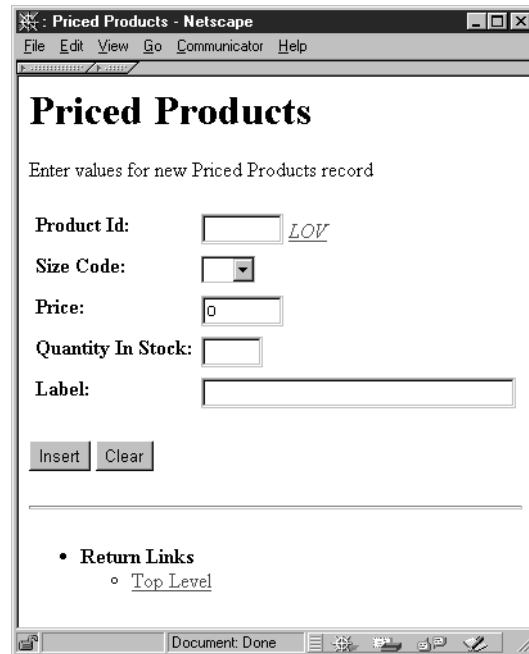
14. On the message window toolbar, click the List Actions button.



15. Highlight "Run generated application using browser..." and click the Run button.

Your browser executable is opened and the generated application is shown.

The first page of the Web application is a query form. Click the New button to move to the "add new product to menu" functionality (the Priced Products page).

A screenshot of a Netscape browser window titled "Priced Products - Netscape". The browser's menu bar includes "File", "Edit", "View", "Go", "Communicator", and "Help". The address bar shows a URL starting with "http://". The main content area displays the title "Priced Products" in a large, bold font. Below the title, the text "Enter values for new Priced Products record" is shown. The form contains several input fields: "Product Id:" with a text box containing "LOV", "Size Code:" with a dropdown menu, "Price:" with a text box containing "0", "Quantity In Stock:" with a text box, and "Label:" with a wide text box. Below these fields are "Insert" and "Clear" buttons. At the bottom of the form, there is a section titled "Return Links" with a sub-link "Top Level". The browser's status bar at the bottom shows "Document: Done" and various navigation icons.

Just as with the Forms application, the first field you fill in is Product Id. This is the identity of the product (e.g., 1011 for a Seafood Special pizza) that forms the basis of the new menu item. You can choose from the products on the Products table by clicking LOV next to the Product Id box.



List Of Values: Product Id

Search criterion for Product Id:

Find Close

Product Id	Product Id	Product Description	Product Type	Discount Day
1011	1011	Seafood Special	PIZZA	
1012	1012	Diet cola	BEVERAGE	
1013	1013	Walnut, apple and mayo	SALAD	
1014	1014	Chocolate Surprise	DESSERT	

Click on one of the active Product Ids in the List Of Values table to add it to the Priced Products page. Fill in the other fields to describe the new menu item, then click the Insert button to create the record.

Lesson summary

In this lesson you created module TUTORI0060 and implemented it as a WebServer application.

Lesson 11

Conclude the tutorial

This lesson describes what you must do after completing the tutorial.

Exit from Oracle Designer and drop the tutorial tables

Instructions

1. Close any Oracle Designer tools you might have open.
2. Select File → Exit to close the Oracle Designer front panel.

Where to go for more information

You have now completed the tutorial. Remember that the tutorial does not attempt to describe in detail the functionality of every Oracle Designer product component. Instead, the tutorial concentrates on how to use the components together to progress from a high level business model to working generated applications.

For comprehensive information about Oracle Designer, refer to the help system referenced in the “Read This First!” section at the start of this tutorial.

Dropping the tutorial tables

To delete the tutorial tables when you have finished the tutorial:

1. Select the Drop Tutorial Tables program in the Oracle Designer Tutorial folder.
The SQL*Plus window appears.
2. Enter the username, password and database, in the format
`<your_username>/<your_password>@<your_database>`
You are prompted to confirm that you want to delete the tutorial tables.
3. To delete the tables, type `Y` and press [Return].
When the operation is complete, you are prompted to close the SQL*Plus window.
4. Type `EXIT` and press [Return] to close the SQL*Plus window.

Tutorial installation notes

This appendix is intended for Repository Managers who are responsible for enabling users to follow the Oracle Designer tutorial. This appendix explains how to import the TUTORIAL application system. For information purposes only, this appendix also lists the files installed by the Oracle Installer, and describes what users have to do before starting the tutorial. The sections in this appendix are:

- importing the TUTORIAL application system
- setting up for WebServer Generator users
- what the Oracle Installer installs on the client
- how tutorial users are instructed to start the tutorial

Importing the TUTORIAL application system

The tutorial in this document requires that users have access to an application system containing predefined entity and function definitions. This application system is shipped with Oracle Designer, and was originally created with the name TUTORIAL.

To import the application system, you must start the Repository Object Navigator as a user with:

- the Oracle Designer Repository Manager privilege

You can import the TUTORIAL application system:

- as the tutorial user
- as a different user (in which case you will have to grant the tutorial user access to the TUTORIAL application system)

The instructions given to users elsewhere in this document assume the application system is called TUTORIAL. When you import the TUTORIAL application system, you have the option to give the imported application system a different name. If you do give the application system a name other than TUTORIAL, make sure you tell users the name.

To import the TUTORIAL application system:

1. Start Oracle Designer as a user with the Repository Manager privilege.
2. Select any application system from the drop-down list in the Application System dialog box and click on OK. If you do not have access to any existing application systems, click on Cancel.
3. Start the Repository Object Navigator.
4. Select Application → Restore to display the Application-Restore dialog box.
5. Click on the Browse button to display the Select XT Archive Import File dialog box.
6. Specify tutapsys.dmp as the file to import.

By default, this file is located in the \ctutx71 directory, below the ORACLE_HOME directory. For example:

Environment	Example location of tutapsys.dmp
Windows 95	C:\orawin95\ctutx71\tutapsys.dmp
Windows NT	C:\orant\ctutx71\tutapsys.dmp

Click the Open button to close the Select XT Archive Import File dialog box

7. Ensure that tutapsys.dmp is specified in the Import File box, then click on the Import button to begin the import.

If the current user has not imported or exported an application system before, you will be prompted to create the archive tables before the file is imported. Click on OK to accept the default tablespaces for tables and indexes.

If the archive tables already exist but are not empty, you will be prompted to delete existing data before continuing. Click on OK.

8. If you want to change the name that the application system will be given in your repository, type the new name in the New Name field.

9. Click on the original name of the application system (i.e. TUTORIAL) to select it.
10. Click on the Restore button to create the application system in the Repository with the name you specified.

When the process is complete, the application system status shown in the dialog box is set to **Restored**.

11. Click on Close to close the Application - Restore dialog box.
12. Either follow the instructions below to grant access to the new application system, or close the Repository Object Navigator.

If you are prompted to save changes (i.e. to the unnamed application system created when you started the Repository Object Navigator if you did not have access to any application systems), click on No.

Granting access to the TUTORIAL application system

If you were not logged on as the tutorial user when you imported the TUTORIAL application system, you must grant access to the TUTORIAL application system to the tutorial user now.

1. In the Repository Object Navigator, select Application → Grant Access By Appl to display the Application - Grant Access By Application dialog box.
2. Select the name you have given the TUTORIAL application system from the For Application drop-down list.
3. Click on the Add User button to display the Add User dialog box.
4. Click on the tutorial user name.
5. Click on the OK button.
6. Check the Access Rights checkboxes as indicated below:

Select	Insert	Update	Delete	Share	Admin
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

7. Click on the Grant button to grant the tutorial user access to the TUTORIAL application system with the specified access rights.
8. Click on Close to close the Application - Grant Access by Application dialog box.

9. Close the Repository Object Navigator.

If you are prompted to save changes (i.e. to the unnamed application system created when you started the Repository Object Navigator if you did not have access to any application systems), click on No.

Setting up for WebServer Generator users

Tutorial users will not be able to follow lesson 10 unless WebServer Generator has been installed and set up.

Set up WebServer Generator as described in the supplied on-line documentation.

Be prepared to make the following information available to the tutorial user:

- the username, password and connect string for the WebServer
- connection information for the Web Agent URL

The connection information is needed for the WebServer Generator Options dialog box. You can see this dialog box by opening the Design Editor and selecting Options → Generator Options → WebServer. Use the Help button on the dialog box to obtain a detailed description of what is required. You may wish to enter this information before the tutorial user starts the lessons.

The tutorial user will also need to know the location of a Web browser executable, which may be local.

What the Oracle Installer installs on the client

This section describes the operation of the Oracle Installer during installation of the Oracle Designer tutorial, and is for information purposes only.

During installation, the Oracle Installer:

- Copies into the <ORACLE_HOME>\ctutx71 directory the following files:

tutapsys.dmp	destut.drp	destut.sql
destut.dat	yesno.sql	destut.txt
- Creates a directory in <ORACLE_HOME>\ctutx71\ called tutorial.

- Copies into the <ORACLE_HOME>\ctutx71\tutorial directory the following files:
 - tutorial.fmb
 - tutorial.olb
- Creates a folder called Oracle Designer Tutorial.
- Populates the Oracle Designer Tutorial folder with the following programs:
 - About the Tutorial (displays the text file destut.txt)
 - Drop Tutorial Tables (runs the SQL script destut.drp)
 - Oracle Designer for Tutorial User (starts Oracle Designer with the working directory set to <ORACLE_HOME>\ctutx71\tutorial)
 - Populate Tutorial Tables (runs the SQL script destut.sql).

Note: The working directory specified for the Oracle Designer for Tutorial User program must be writeable.

How tutorial users are instructed to start the tutorial

When following the tutorial, users are instructed to start Oracle Designer by selecting the Oracle Designer for Tutorial User program in the Oracle Designer Tutorial folder.

The working directory of this program item is set to <ORACLE_HOME>\ctutx71\tutorial.

