Database Management I (420-D10-HR)

Lab 05 - Drawing Entity-Relationship Diagrams with Oracle SQL Developer Data Modeler

Date assigned: Wednesday, September 7, 2016

Date due: **Wednesday, September 7,2016 11:50am**

**Learning Objectives**

Upon successful completion of this lab exercise, the student will be able to:

1. Create a logical data model using Oracle SQL Developer Data Modeler.
2. Define domains in Oracle SQL Developer Data Modeler.
3. Define entities, attributes and relationships in Oracle SQL Developer Data Modeler.

**To Start:**

1. Create a folder called ***username*\_D10\_L05\_Data\_Modeler** in your **420-D10\Labs folder**.

**To Be Handed In:**

1. Your ***username*\_D10\_L05\_Data\_Modeler** folder should zipped and uploaded to **Moodle**.

# Tutorial: Data Modeling for a Small Database[[1]](#footnote-1)

***Purpose:*** Learn to draw aPern entity relationship diagram using Oracle Data Modeler.

***To Do:***

In this tutorial, you will use **Oracle SQL Developer Data Modeler** to a create model for a simplified library database.

The logical model for the database includes three entities: Book (describes each book in the library), Patron (describes each person who has a library card), and Transaction (describes each transaction involving a patron and a book). Before you create the entities, you will create some domains to make the entity creation more meaningful and specific.

## Click **Oracle SQL Developer Data Modeler** in the task bar to open it. NOTE: The Data Modeler is a different application than the SQL Developer you’ve been using. Hunt for it and use the right application.

## Set up your defaults:

### Click **Tools** and select **Preferences**.

### Click **Data Modeler** in the left hand panel.

### Change the **Default Designs Directory**, the **Default Save Directory** and the **Default System Types Directory** to your **420-D10\Labs\ *username*\_D10\_L05\_Data\_Modeler** folder.

### Expand **Data Modeler** and click **Diagram** in the left hand panel.

### Expand **Diagram** and click **Logical Model**.

### Make sure that the **Notation Type** is **Barker**.

### Click **OK**.

## **Adding Domains**

In planning for your data needs, you have determined that several kinds of fields will occur in multiple kinds of records, and many fields can share a definition. For example, you have decided that:

* The first and last names of persons can be up to 25 characters each.
* Street address lines can be up to 40 characters.
* City names can be up to 25 characters.
* Province codes are 2-character standard abbreviations.
* Postal codes are 6 characters.
* Book identifiers can be up to 20 characters.
* Other identifiers are numeric, with up to 7 digits (no decimal places).
* Titles (books, articles, and so on) can be up to 50 characters.

You therefore decide to add appropriate domains that you can later use to specify data types for attributes. (These added domains will also be available after you exit Data Modeler and restart it later.)

### Click Tools, then Domains Administration.

### In the [Domains Administration](http://download.oracle.com/docs/cd/E15276_01/doc.20/e13677/dialogs_data_modeling.htm#BABBFGAA) dialog box, add domains with the following definitions. Click Add to start each definition, and click Apply after each definition.

| **Name** | **Logical Type** | **Other Information** |
| --- | --- | --- |
| Person Name | VARCHAR | Size: 25 |
| Address Line | VARCHAR | Size: 40 |
| City | VARCHAR | Size: 25 |
| Province | CHAR | Size: 2 |
| Postal Code | CHAR | Size: 6 |
| Book Id | VARCHAR | Size: 20 |
| Numeric Id | NUMERIC | Precision: 7, Scale: 0 |
| Title | VARCHAR | Size: 50 |

### Add the provincial codes for Quebec and Ontario to the list of possible values for Province:

* Select **Province** from the list of domains and click the **Modify** button.
* Click the **Value List** button.
* Click the **Add** button. Enter **QC** for **Value** and **Quebec** for **Description**.
* Repeat for **ON** and **Ontario**.
* Click **OK**.
* Click **Apply**.

### Modify the **Province** domain again. This time set the **Default Value** to **QC**.

### Click Save. This creates a file named defaultdomains.xml in your lab folder.

### Click Close to close the dialog box.

### In the main area (right side) of the SQL Developer Data Modeler window, click the **Logical** tab (or right click on Logical Model from the Browser window)

### Right click in drawing area and select **Show🡪Grid** and **Show🡪Labels**.

## **Creating the Book Entity**

The Book entity describes each book in the library.

### Click the **New Entity** icon.

### Click in the logical model pane in the main area to draw an entity box. The [Entity Properties](http://download.oracle.com/docs/cd/E15276_01/doc.20/e13677/dialogs_data_modeling.htm#BABHACHD) dialog box is displayed.

### Click General on the left, and specify as follows:

Name: Book

### Click Attributes on the left, and use the Add (+) icon to add the following attributes, one at a time. (For datatypes, select from the Domain types except for Rating, which is a Logical type.) If a description is given in the Notes column, click the Notes tab at the bottom of the Attribute Properties and copy the note.

| **Name** | **Data type** | **Other** | **Notes** |
| --- | --- | --- | --- |
| book\_id | Data type: Domain  Domain: Book Id | Primary UID | The Dewey code or other book identifier. |
| title | Data type: Domain  Domain: Title | Mandatory |  |
| author\_last\_name | Data type: Domain  Domain: Person Name | Mandatory |  |
| author\_first\_name | Data type: Domain  Domain: Person Name |  |  |
| rating | Data type: Logical  Type: NUMERIC (Precision=2, Scale= 0) |  | Librarian's personal rating of the book, from 1 (poor) to 10 (great). |

### Click OK to finish creating the Book entity.

## **Creating the Patron Entity**

The Patron entity describes each library patron (that is, each person who has a library card and is thus able to borrow books).

### Add a new entity called **Patron** to the diagram.

### Add the following attributes, one at a time. (For data types, select from the Domain types, except for location, which uses the structured type SDO\_GEOMETRY.)

| **Attribute Name** | **Type** | **Other** | **Notes** |
| --- | --- | --- | --- |
| patron\_id | Domain: Numeric Id | Primary UID | Unique patron ID number, also called the library card number. |
| last\_name | Domain: Person Name | Mandatory | Patron's last name. |
| first\_name | Domain: Person Name |  | Patron's first name. |
| street\_address | Domain: Address Line |  | Patron's street address. |
| city | Domain: City |  | City or town where the patron lives. |
| province | Domain: Province |  | 2-letter code for the province where the patron lives. |
| postal\_code | Domain: Postal Code |  | Postal code where the patron lives. |
| location | Structured type: SDO\_GEOMETRY |  | Oracle Spatial geometry object representing the patron's geocoded address. |

### Click OK to finish creating the **Patron** entity.

## **Creating the Transaction Entity**

The Transaction entity describes the transactions that involve a patron and a book, such as someone checking out or returning a book. Each record represents a transaction between one patron and one book. For example, a patron returning two books and checking out three books causes five transactions to be recorded (two returns and three checkouts). Create the Transaction entity as follows:

### Add New Entity called **Transaction** to the diagram.

### Add the following attributes, one at a time. (For datatypes, select from the Domain types, except for transaction\_date, which uses a Logical type.)

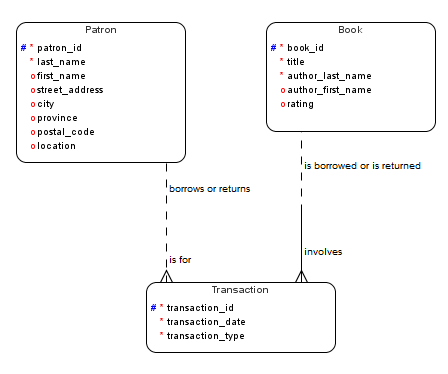
|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Other** | **Notes** |
| transaction\_id | Domain: Numeric Id | Primary UID | Unique transaction ID number |
| transaction\_date | Logical type: Date | Mandatory | Date and time of the transaction. |
| transaction\_type | Domain: Numeric Id | Mandatory | Numeric code indicating the type of transaction, such as 1 for checking out a book. |

### Click OK to finish creating the Transaction entity.

## **Creating Relations between Entities**

Relations show the relationships between entities: one-to-many, many-to-one, many-to-many or one-to-one. The following relationships exist between the entities:

* **Book and Transaction**: one-to-many. Each book can be involved in multiple sequential transactions. Each book can have zero or one active checkout transactions; a book that is checked out cannot be checked out again until after it has been returned.
* **Patron and Transaction**: one-to-many. Each patron can be involved in multiple sequential and simultaneous transactions. Each patron can check out one or many books in a visit to the library, and can have multiple active checkout transactions reflecting several visits; each patron can also return checked out books at any time.

 Create the relationships as follows. When you are done, the logical model pane in the main area should look like the following figure:

### In the logical model pane in the main area, arrange the entity boxes as follows: Patron on the left, Book on the right, and Transaction either between Book and Patron or under them and in the middle. (If the pointer is still cross-hairs, click the Select icon at the top left to change the pointer to an arrow.)

### Click the **New 1:N Relation** icon.

### Click first in the **Book** box, then in the **Transaction** box. A line with an arrowhead is drawn from Book to Transaction.

### The [Relation Properties](http://download.oracle.com/docs/cd/E15276_01/doc.20/e13677/dialogs_data_modeling.htm#BABDJCBE) information window should open:

#### Type "book transaction" in the **Name** field.

#### Type "is borrowed or is returned" in the **Name on Source** field.

#### Type "involves" in the **Name on Target** field.

#### Uncheck the **Target Optional** check box.

#### Click **OK**.

### Click the **New 1:N Relation** icon.

### Click first in the **Patron** box, then in the **Transaction** box. A line with an arrowhead is drawn from Patron to Transaction.

### Repeat steps d). Name the relationship the source name is "borrows or returns" and the target name "is for".

## Make any adjustments necessary so that the diagram looks like the one shown above. ***Paste in a screen capture of your diagram below.***

## Save the model as ***username*\_D10\_L05\_Tutorial** in your ***username*\_D10\_L05\_Data\_Modeler** folder.

### H:\Databases\Labs\pdumaresq_D10_L05\entities.PNG

# Drawing an Entity Relationship Diagram on Your Own

***Purpose:*** Draw different types of entity relationships using Oracle Data Modeler.

***To Do:***

## Right-click on **Designs** in the Navigation panel and select **New Design**. Right click on the background of the new design and select Show -> Labels, Show -> Grid.

## Draw the following Entity Relationship Diagrams in the drawing area. ***Paste in a copy of your screen capture below each corresponding diagram.***

### 

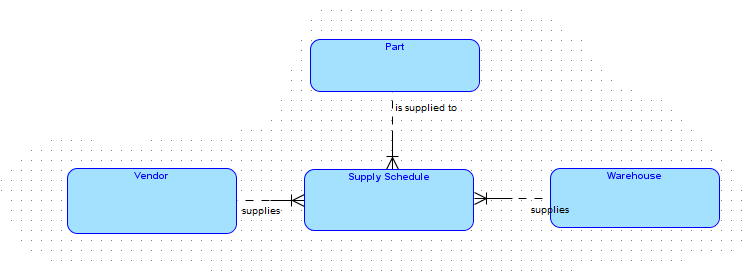
***Note***: Both sides of the relationship (Source and Target) should have the optional checkbox unchecked. ****

### 

### 

### 

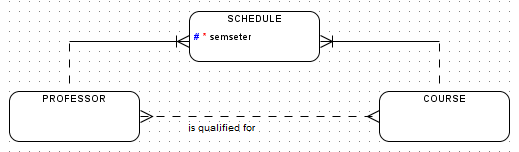
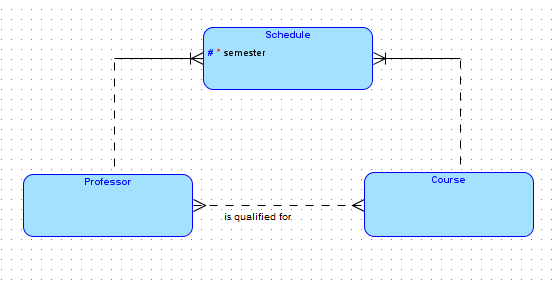
### 

******

***Note***:

SUPPLYSCHEDULE is a weak entity (identified by the short vertical bar beside the fork at the end of the relationships. For a weak entity, you must click the **Identifying** check box.

### 

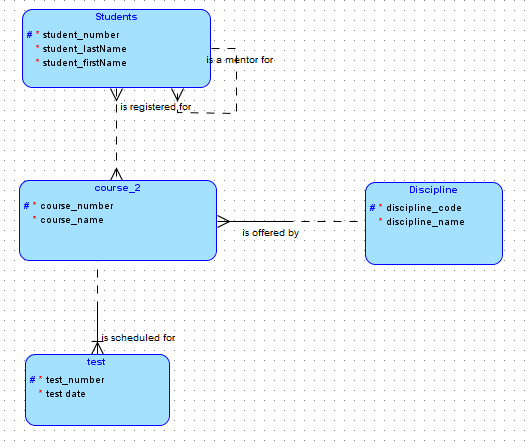


## Save the diagrams as ***username*\_D10\_L05\_Examples** and close the diagram. Draw the entity relationship diagram shown here.

The types for the attributes follow:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entity** | **Attribute name** | **Domain/**  **Logical type** | **Size/**  **Precision** | **Scale** |
| STUDENT | Student Number | Numeric Id |  |  |
| STUDENT | Student lastname | Person Name |  |  |
| STUDENT | Student firstname | Person Name |  |  |
| COURSE | Course number | char | 8 |  |
| COURSE | Course name | Title |  |  |
| DISCIPLINE | Discipline code | numeric | 3 | 0 |
| DISCIPLINE | Discipline name | Title |  |  |
| TEST | Test number | numeric | 2 | 0 |
| TEST | Test date | date |  |  |

## Save the diagram as ***username*\_D10\_L05\_College\_System** and close the diagram. ***Also, paste in a screen capture of the diagram below:***



## Exit from Oracle Data Modeler.

1. Adapted from the tutorial in **Oracle® SQL Developer Data Modeler User's Guide -** <http://download.oracle.com/docs/cd/E15276_01/doc.20/e13677/tut_data_modeling.htm#CBAHDFAF> [↑](#footnote-ref-1)