DT228/2, DT282/2 Databases I
Data Modelling

Objectives

The objectives of this lab are to:

- To provide you with an opportunity to become very familiar with putting together a data model.
- To allow you the opportunity to become familiar with using ERWin.
- To allow you to start thinking the assignment.

Contents

1. Build a data model

For this model you need to choose datatypes you think are appropriate. Choose from number (with scale at least and precision if needed), varchar2 (with scale and precision if needed) and Date.

Galleries keep information about artists, their names (which are unique), birthplaces, age, and style of art.

For each piece of artwork, the artist, the year it was made, its unique title, its types of art (e.g., painting, lithograph, sculpture, photograph), and its price must be stored. It is assumed the each piece of art is by one artist but that artist may have many pieces of art in the gallery.

Pieces of artwork are also classified into styles of art e.g. portrait, landscape, impressionism, cubism, sculpture etc. ; a given piece may belong to more than one style. Any artist may paint or practice in more than one style.

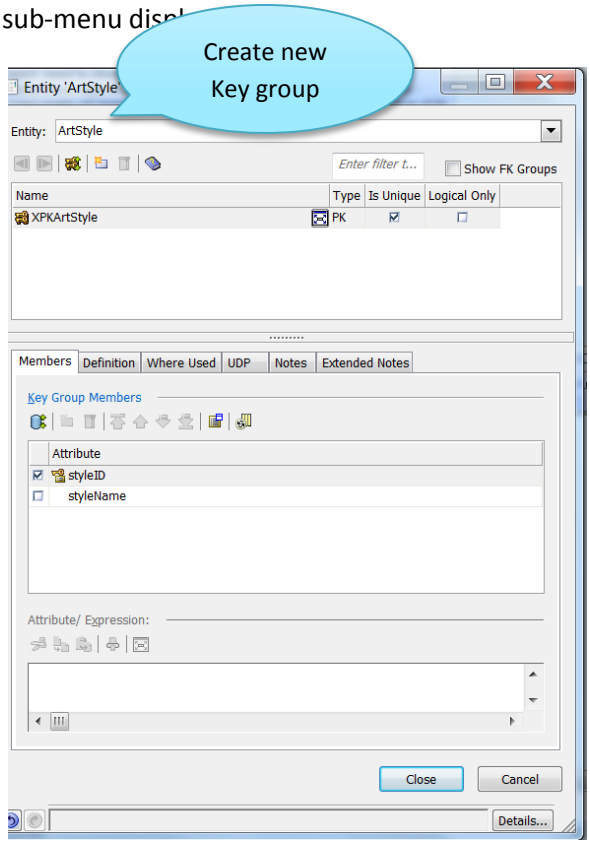
Each style is identified by a unique identifier and a name (like those just given) that describes the style.

Finally, galleries keep information about customers. For each customer, galleries keep that person's unique name, address, total amount of dollars spent in the gallery (very important!), and the artists and styles of art that the customer tends to like.

2. Adding UNIQUE constraints to a model

Suppose we want to introduce some value constraints to the models we created.

For our Gallery we want to make the name of painting style unique. To do this in ERWIN we need to add a UNIQIUE constraint to the attribute of the entity we created to represent art style. Right click on the entity and select KEY GROUP PROPERTIES. This will open the dialog below. Create a new key group and select New Alternate key from the sub-menu disp



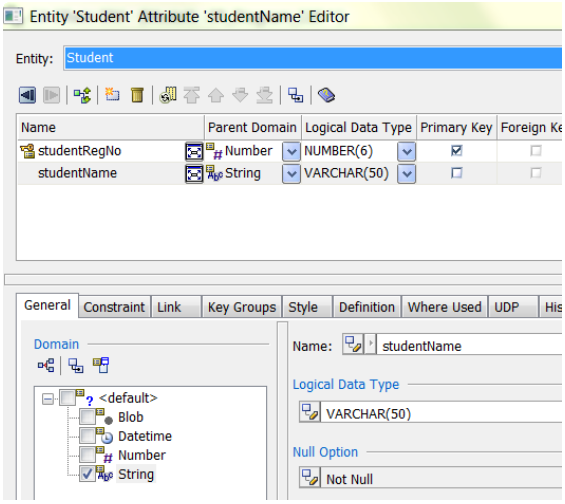
We need to add a new key group. In the bottom dialog click the name of the attribute you created for style name (make sure it is the only one selected) and then close the dialog.

The reason we are creating an alternate key is that any attribute that has unique values is a potential primary key. If we also add the constraint NOT NULL that is a candidate for primary key (primary keys need always to have a value, not change and be unique).

3. Adding NOT NULL constraints to a model

Again for our gallery we want to make sure that all customers provide us with an address. To make sure it is not possible to create a customer with no value for this attribute we need to add a NOT NULL constraint.

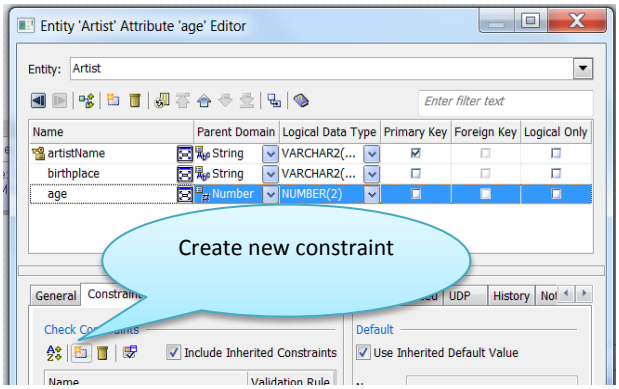
To set NOT NULL, right click on the entity, open the attribute dialog, select the General Tab, select the attribute you are interested in and choose Not Null from the **Null option** drop down list.



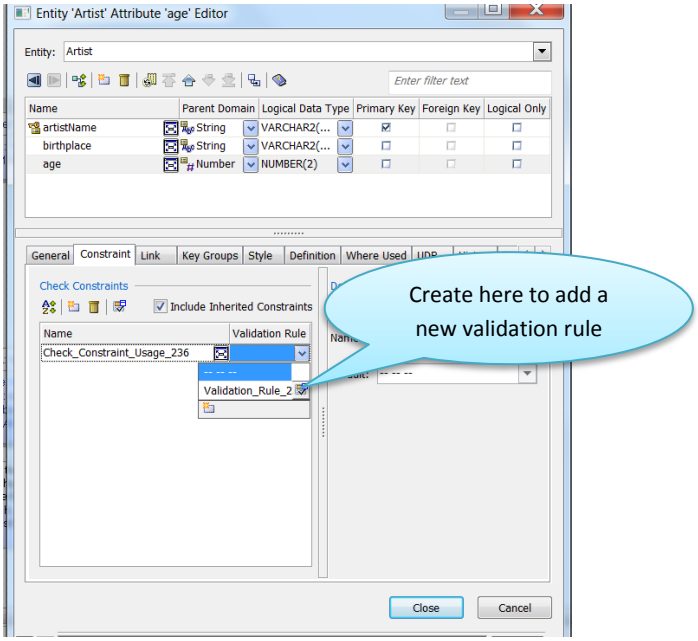
4. Adding VALUE constraints to a model

Again for our gallery we want to make sure that all artists are 18 or older. To make sure it is not possible to create an artist with value less than 18 for this attribute we need to add a value constraint.

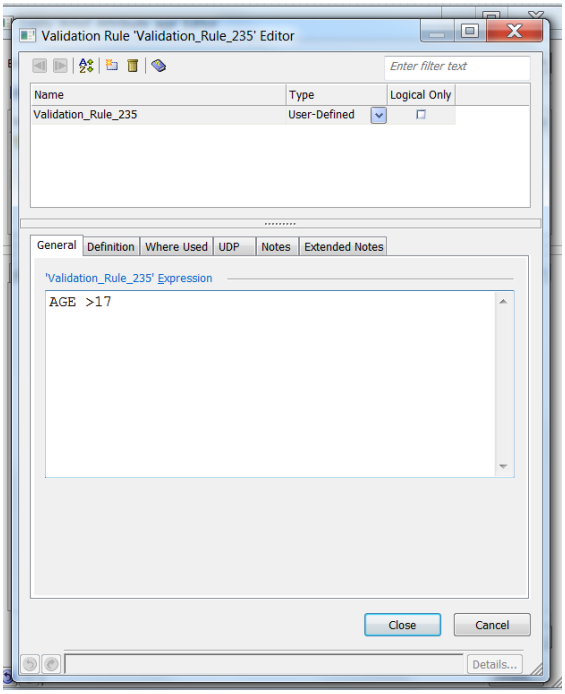
To do this we open the attributes dialog and click on the attribute we created for age. Then select the CONSTRAINTS TAB and click on create new constraint.



This will add a constraint, change the name to check_age then open the constraint editor under Validation Rule to create the constraint.



This is a check constraint so we need to add the bit that needs to be checked e.g. if our attribute is called age and we want it to 18 or over then we can include as a rule age >17.



5. Changing what is shown on the ERWin canvas

If you Right-click anywhere on the ERWin canvas you can choose properties from the sub-menu that appears.

To change what is shown for entities, under the entity tab click on what you want to be show e.g. you might want to show the datatype of the attributes.

To change what is shown for relationships, under the Relationship (singular) tab click on what you want to show.

When you change to physical view again right-click on the canvas and you can choose what you want to be displayed.

6. Another Data Model

You have been hired to design an RDBMS for the Luxury Limousines Inc. which operates a number of vehicles. The relevant information is given below.

Every vehicle has a registration number and each vehicle is of a specific model. Each model is identified by a model number (e.g. LIN-2000) and has a capacity and weight. In addition, the model also has a range (eg. 100 km, 1000 km) associated with it. A value is required for all these attributes (i.e. null values are not allowed).

A number of technicians work for the company. You need to store the name, employee no., address, phone number and salary of each technician. You can assume that the employee no. uniquely identifies each employee. A name, address, phone number and salary must be provided for each technician.

Each technician specializes in one or more vehicle models. This expertise may overlap with that of other technicians. The company has employees working as controllers who control the incoming and outgoing vehicle traffic in the vehicle areas.

The company performs a number of checks periodically to ensure that the vehicles are in good condition. These tests are standardized by the Bureau of Motor Vehicles (BMV). Each test is identified by a BMV test number. The BMV number should be unique. The test also has a name and a maximum possible score. The name of the test should be unique and the maximum possible score must have a value and should be less than 2000.

These BMV tests are conducted by the company technicians. The BMV requires the company to keep track of each time a given vehicle is checked by a given technician using a given test. The information for each testing event is the date, the number of hours spent in testing and the score the vehicle received on the test as well as the vehicle and technician involved.

7. Read the case study specification provided for the assignment.