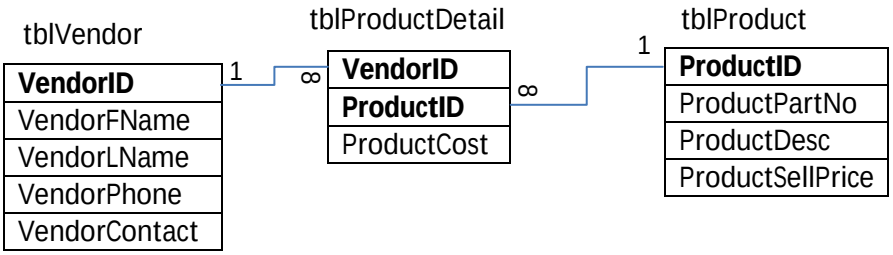
DDL – Data Definition Language

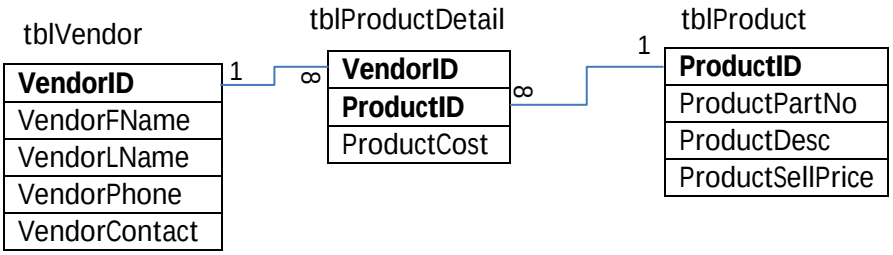
CREATE ALTER DROP RENAME	**Data Definition Language (DDL)** Creates, changes, and removes data structures.

Examples for the CREATE TABLE statement:



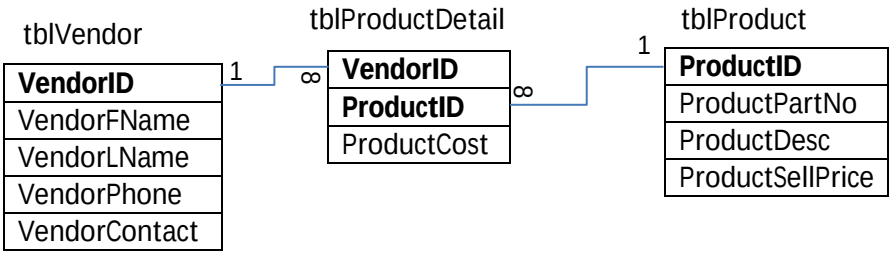
Create a table - And give the primary key constraint a name.	**CREATE TABLE** *tblProduct* **(** *ProductID* **NUMBER(***38***) CONSTRAINT** tblProduct_pk **PRIMARY KEY,** *ProductPartNo* **VARCHAR2(***15***),** *ProductDesc* **VARCHAR2(***35***),** *ProductSellPrice* **Number(***5,2***)** **);**
Create a table – Same as above but do not give the PK constraint a name.	**CREATE TABLE** *tblProduct* **(** *ProductID* **NUMBER(***38***) PRIMARY KEY,** *ProductPartNo* **VARCHAR2(***15***),** *ProductDesc* **VARCHAR2(***35***),** *ProductSellPrice* **Number(***5,2***)** **);**
Create a table with a composite primary key.	**CREATE TABLE** *tblProductDetail* **(** *ProductID* **NUMBER(***38***),** *VendorID* **NUMBER(***38***),** *ProductCost* **Number(***5,2***),** **PRIMARY KEY (***ProductID, VendorID***)** **);**
Create a table – Same as above but now add in the foreign key constraints.	**CREATE TABLE** *tblProductDetail* **(** *ProductID* **NUMBER(***38***),** *VendorID* **NUMBER(***38***),** *ProductCost* **Number(***5,2***),** **PRIMARY KEY (***ProductID, VendorID***),** **FOREIGN KEY (***ProductID***) REFERENCES** *tblProduct***(***ProductID***),** **FOREIGN KEY (***VendorID***) REFERENCES** *tblVendor***(***VendorID***)** **);**
Create a table and set the ProductSellPrice to a default value of 0.	**CREATE TABLE** *tblProduct* **(** *ProductID* **NUMBER(***38***) PRIMARY KEY,** *ProductPartNo* **VARCHAR2(***15***),** *ProductDesc* **VARCHAR2(***35***),** *ProductSellPrice* **Number(***5,2***) DEFAULT** *0* **);**
Create a table with a check constraint on the SellPrice field – this check constraint ensures the selling price is between 0 and 100 dollars.	**CREATE TABLE** *tblProduct* **(** *ProductID* **NUMBER(***38***) PRIMARY KEY,** *ProductPartNo* **VARCHAR2(***15***),** *ProductDesc* **VARCHAR2(***35***),** *ProductSellPrice* **Number(***5,2***),** **CONSTRAINT** *check_SellPrice* **CHECK (***ProductSellPrice* **Between** *0* **AND** *100***)** **);**

Examples for the ALTER TABLE statement:



Add a column to an existing table	**ALTER TABLE** *tblVendor* **ADD** *VendorCity* **VARCHAR2(***20***);**
Add more than one column to an existing table	**ALTER TABLE** *tblVendor* **ADD** *VendorCity* **VARCHAR2(***20***),** **ADD** *VendorProvince* **VARCHAR2(***25***);**
Change the size of an existing column or change the data type Note: You cannot modify multiple columns in one statement.	**ALTER TABLE** *tblProduct* **MODIFY** *ProdDesc* **VARCHAR2(***55***);**
Changing the default value of a column.	**ALTER TABLE** *tblProductDetail* **MODIFY** *ProductCost* **DEFAULT** *0*;
Removing (dropping) a column	**ALTER TABLE** *tblVendor* **DROP COLUMN** *VendorContact*;

Working with Indexes:



Create a non-unique index on the PartNo field and give the index a name of ndx_Product_PartNo	**CREATE INDEX** *ndx_Product_PartNo* **ON** *tblProduct*(*ProductPartNo*);
Same as above but now a unique index	**CREATE UNIQUE INDEX** *ndx_Product_PartNo* **ON** *tblProduct*(*ProductPartNo*);
Create a composite index on the first and last name	**CREATE INDEX** *ndx_Vendor_Name* **ON** *tblVendor*(*VendorFName*,*VendorLName*);
Creates a function based index on the PartNo	**CREATE INDEX** *ndx_Product_PartNo* **ON** *tblProduct*(**UPPER**(*ProductPartNo*));
Renaming an index	**ALTER INDEX** *ndx_Product_PartNo* **RENAME TO** *ndx_Product_Part*;
Remove (drop) an index	**DROP INDEX** *ndx_Product_PartNo*;