**Objective**: At the end of this lab session you will be able to learn about the CREATE TABLE command used to create database tables and to specify Primary key, foreign key and other constraints on the tables created.

## Section 1

***Create table:***

The CREATE TABLE statement is used to create a new table in a database.

**Syntax for the CREATE TABLE**

CREATE TABLE <table\_name> (  
    <column\_name1> data\_type,  
    <column\_name2> data\_type,  
    <column\_name3> data\_type,

...

);

The following table lists few of the data types that is used in SQL

|  |  |
| --- | --- |
| **Data type** | **Definition** |
| Integer | For numbers. |
| Char() | Use to store textual information.   * The char data type is used to hold a fixed length text. |
| Varchar() | Used to store textual information.   * The varchar data type is similar to char but stores variable length text. * Eg: char(10) – to store the NIC values.   Varchar(50) – To store the address of a customer. |
| Datetime | For date and time. |
| Real | For floating point numbers |
| Money | For currency values. |

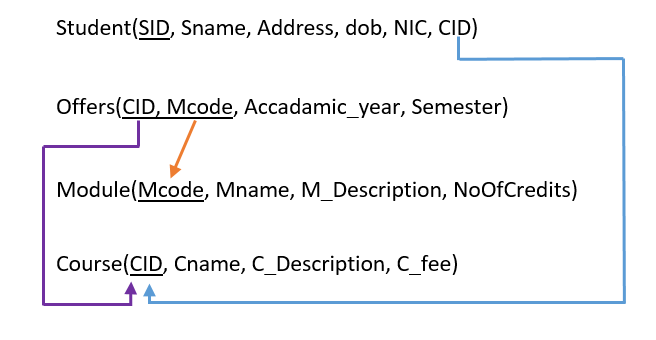
The following table shows the Syntaxes and the definitions for the Primary Key, Foreign Key and Check constraints.

|  |  |  |
| --- | --- | --- |
| **Constraint** | **Syntax** | **Definition** |
| Primary Key Constraint | 1. **Primary key (column name)** 2. **constraint** <constraint\_name> **primary** **Key** (<column \_name>) | The PRIMARY KEY constraint uniquely identifies each record in a table. Primary keys must contain UNIQUE values, and cannot contain NULL values. |
| Foreign Key Constraint | 1. **Foreign key (referncing column) refernces <refernced table> (refernced column)** 2. **constraint** <constraint\_name> **foreign** **key** (<this\_table\_column\_name>) **references** <reference\_table> (<reference\_table\_column\_name>) | A FOREIGN KEY is a key used to link two tables together.  A FOREIGN KEY is a field (or collection of fields) in one table that refers to the PRIMARY KEY in another table. |
| Check Constraint | 1. **Check <condition>** 2. **constraint** <constraint\_name> **check** (<put the condition here>) | The CHECK constraint is used to limit the value range that can be placed in a column. |

\*\*\* even though there are different methods available inform the students that method 2 which provides name for the constraint is the best option, since we can easily figure out the error using the constraint name.

## Section 2

1. Consider the following relational schema.



1. List the primary keys and foreign keys you have identified in each table?

*(Hint: In the above schema, Foreign keys (FK) are pointed by tail side of the arrow and the Primary Keys (PK) are pointed by the head side of the arrow.)*

|  |  |  |
| --- | --- | --- |
| **Table Name** | **Primary Key(PK)** | **Foreign Key(FK)** |
| Student | SID | CID |
| Offers | CID, Mcode | Mcode, CID |
| Module | Mcode | - |
| Course | CID | - |

1. What is the correct order of creating tables in the above mention relational schema?

*Hint: When you consider about the order of creating tables, as the first table you have to identify the table which are not having foreign keys.*

CREATE TABLE <table name> (

<column\_name > data type,

<column\_name2>data type,

);

Order of creating tables : (consider foreign key)

* + - 1. 🡪 Module, Couse
      2. 🡪 Student
      3. 🡪Offers

1. Consider the following data types for the above schema, create the following relational database using CREATE TABLE SQL statement. Ensure that appropriate referential integrity constraints (Foreign key) are met. Save the script as “CreateDB.sql”.

Student(SID:*CHAR(10)*, Sname:*VARCHAR(50)*, Address:*VARCHAR(50)*, dob:*DATE*, NIC:*CHAR(10)*, CID:*CHAR(6)*)

Offers (CID:*CHAR(6)* , Mcode:*CHAR(6)*, Accadamic\_year:*CHAR(2)*, Semester:*INTEGER*)

Module (Mcode:*CHAR(6)*, Mname:*VARCHAR(50)*, M\_Description:*VARCHAR(200)*, NoOfCredits:*INTEGER*)

Course (CID:*CHAR(6)*, Cname:*VARCHAR(50)*, C\_Description:*VARCHAR(200)*, C\_fee:*INTEGER*)



CREATE TABLE Course (

CID CHAR(6),

Cname VARCHAR(50),

C\_Description VARCHAR(200),

C\_fee INT

CONSTRAINT Couse\_PK PRIMARY KEY(CID)

);

CREATE TABLE Module (

Mcode CHAR(6),

Mname VARCHAR(50),

M\_Description VARCHAR(200),

NoOfCredits INT,

CONSTRAINT Module\_PK PRIMARY KEY (Mcode)

);

CREATE TABLE Students(

SID CHAR(10),

Sname VARCHAR(50),

Address VARCHAR(50),

dob DATE,

NIC CHAR(10),

CID CHAR(6),

CONSTRAINT Student\_PK PRIMARY KEY(SID),

CONSTRAINT Student\_FK FOREIGN KEY(CID) REFERENCES Course(CID)

);

CREATE TABLE Offers (

CID CHAR(6) ,

Mcode CHAR(6),

Accadamic\_year CHAR(2),

Semester INT

CONSTRAINT Offers\_PK PRIMARY KEY (CID, Mcode),

CONSTRAINT Offers\_FK\_1 FOREIGN KEY (CID) REFERENCES Course(CID),

CONSTRAINT Offers\_FK\_2 FOREIGN KEY (Mcode) REFERENCES Module(Mcode)

);

1. What is the difference between ALTER and DROP in SQL?

Alter is used to modify an existing table, for example by adding a column to it.

DROP is used to remove a table completely.

1. Use **CHECK** constraint to enforce the following rules stated by modifying existing ***Student*** and ***Module*** tables.

*Hint: Use*

**ALTER TABLE <table\_name>**

**ADD CONSTRAINT <constraint\_name> CHECK(<condition>)**

* Ensure that the Student’s NIC number contains 9 digits (0-9) and one character which is “V”or “v”.

ALTER TABLE Student

ADD CONSTRAINT CHK\_Student

CHECK (NIC LIKE '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][V-V]')

* Ensure that number of credits for module should be one of the following :1,2,3,4

ALTER TABLE Module

ADD CONSTRAINT CHK\_Module

CHECK (NoOfCredits IN (1,2,3,4))