USE UNIVERSITY\_DATABASE

-- STEP 3: HOW TO DEFINE "COURSES" TABLE?

CREATE TABLE tblCourses

(

CourseID INT UNIQUE, -- THIS COLUMN DOES NOT ALLOW DUPLICATES

CourseName VARCHAR(30) NOT NULL, -- THIS COLUMN VALUE IS MANDATORY

CourseDur INT NULL -- THIS COLUMN VALUE IS OPTIONAL

) ON [PRIMARY]

-- STEP 4: INSERT DATA INTO ABOVE TABLE

INSERT INTO tblCourses VALUES (101, 'COMPUTERS', 120)

INSERT INTO tblCourses VALUES (102, 'ROBOTICS', 180)

INSERT INTO tblCourses VALUES (103, 'CIVIL', 120)

-- STEP 5: HOW TO REPORT OR VERIFY FROM ABOVE TABLE

SELECT \* FROM tblCourses

-- STEP 6: HOW TO TEST UNIQUE (KEY) CONSTRAINT?

INSERT INTO tblCourses VALUES (103, 'ELECTRONICS', 120) -- ERROR. REASON : COURSE\_ID IS DUPLICATED.

INSERT INTO tblCourses VALUES (104, 'ELECTRONICS', 120) -- NO ERROR

-- STEP 7: HOW TO TEST NULL VALUES & NULL CONSTRAINT?

INSERT INTO tblCourses VALUES (105, NULL, 120) -- ERROR. REASON : COURSE\_NAME CANNOT BE NULL.

-- STEP 8: HOW TO DEFINE A "STUDENTS" TABLE, MAKE IT RELATED TO THE ABOVE "COURSES" TABLE?

CREATE TABLE tblStudents

(

STD\_ID INT PRIMARY KEY, -- THIS COLUMN DOES NOT ALLOW DUPLICATES AND DOES NOT ALLOW NULL VALUES

STD\_NAME VARCHAR(30) NOT NULL,

STD\_AGE TINYINT CHECK (STD\_AGE >= 18),

STD\_GENDER CHAR(6) CHECK (STD\_GENDER= 'M' OR STD\_GENDER = 'F'),

STD\_CRS\_ID INT REFERENCES tblCourses(CourseID)

)

-- STEP 9: HOW TO INSERT DATA INTO ABOVE TABLE?

INSERT TBLSTUDENTS VALUES (1001, 'SAI', 34, 'M', 101)

INSERT TBLSTUDENTS VALUES (1002, 'JON', 34, 'F', 101)

INSERT TBLSTUDENTS VALUES (1003, 'JOHNE', 34, 'M', 102)

INSERT tblStudents VALUES (1004, 'JOHNY', 34, 'M', 102)

INSERT tblStudents VALUES (1005, 'AMIN', 34, 'M', 101)

INSERT tblStudents VALUES (1006, 'AMINI', 34, 'F', 101)

-- STEP 10: HOW TO TEST PRIMARY KEY?

INSERT tblStudents VALUES (1006, 'AMINISHA', 20, 'F', 101)

-- STEP 11: HOW TO TEST FORIEGN KEY?

INSERT tblStudents VALUES (1007, 'AMINISHA', 20, 'F', 105) -- ERROR. REASON: INVALID COURSE ID

INSERT tblStudents VALUES (1007, 'AMINISHA', 20, 'F', 104)

-- STEP 12: HOW TO DEFINE STAFF TABLE?

CREATE TABLE TBLSTAFF

(

STF\_ID INT IDENTITY(100001,1), -- START VALUE = SEED. INCREMENT VALUE APPLIES FOR EVERY INSERT.

STF\_NAME VARCHAR(30) NOT NULL, -- THIS IS MANDATORY COLUMN

STF\_AGE TINYINT CHECK (STF\_AGE >= 28) DEFAULT 28,

STF\_GENDER CHAR CHECK (STF\_GENDER IN ('M','F')) DEFAULT 'F',

STF\_COURSE\_ID INT REFERENCES tblcourses(CourseID) -- MEANS, STAFF SHOULD BE RECRUITED FOR AVALABLE COURSES ONLY

)

-- STEP 13: HOW TO INSERT DATA INTO ABOVE TABLE?

INSERT TBLSTAFF VALUES ('SAI', 34, 'M', 101)

INSERT TBLSTAFF VALUES ('JON', 34, 'F', 101)

INSERT TBLSTAFF VALUES ('JOHNE', 34, 'M', 102)

-- STEP 14: HOW TO VERIFY DATA FROM ABOVE TABLE?

SELECT \* FROM TBLSTAFF

-- STEP 15: HOW TO INSERT DEFAULT VALUES?

INSERT TBLSTAFF(STF\_NAME,STF\_COURSE\_ID) VALUES ('AMINI', 102)

INSERT TBLSTAFF(STF\_NAME,STF\_COURSE\_ID,STF\_GENDER) VALUES ('AMIN', 102, 'M')

-- STEP 16: HOW TO VERIFY ABOVE TABLE DATA?

SELECT \* FROM TBLSTAFF

-- STEP 17: HOW TO TEST FOREIGN KEY?

INSERT TBLSTAFF VALUES ('JEFFREY', 34, 'M', 666) -- ERROR. REASON : INVALID COOURSE ID

INSERT TBLSTAFF VALUES ('JEFFREY', 34, 'M', 101)

-- STEP 18: HOW TO DEFINE INDEXES ON A TABLE?

CREATE CLUSTERED INDEX IND1 ON TBLSTAFF (STF\_ID)

CREATE NONCLUSTERED INDEX IND2 ON TBLSTAFF (STF\_COURSE\_ID)

CREATE NONCLUSTERED INDEX IND3 ON TBLSTAFF (STF\_GENDER)

SELECT \* FROM TBLSTAFF WHERE STF\_ID >= 100001

SELECT \* FROM TBLSTAFF WHERE STF\_GENDER = 'M'

/\*

WHILE EXECUTING ABOVE QUERIES:

"QUERY OPTIMIZER" COMPONENT DECIDES THE BEST INDEX FROM THE AVAILABLE LIST OF INDEXES IN THE TABLE.

MEANS: FROM ABOVE, THE TABLE HAS TOTAL OF 3 INDEXES.

WHENEVER WE EXECUTE ANY QUERY THEN QUERY OPTIMIZER (QO) WILL AUTOMATICALLY CHECK FOR THE BEST INDEX THAT CAN BE USED

TO PROCESS THE QUERY

-- IT IS RECOMMENDED TO ENSURE PRIMARY KEY DEFINED ON THE TABLE. SO THAT A CLUSTERED INDEX IS AUTO CREATED.

-- ADVANTAGES OF PRIMARY KEY : FASTER TABLE DATA ACCESS

EASY REFERENCE TO FOREIGN KEY CONSTRAINT

\*/

-- IN REAL-TIME, INDEXES ARE RECOMMENDED TO BE CREATED(DEFINED) ON SUCH COLUMNS WHICH ARE FREQUENTLY BEING USED IN SEARCH CONDITIONS