Database Management Systems Lab
Assignment 1

Name: Aryan Mahawar
Registration Number: 23BDS1095
Course Code: BCSE302P (Lab)
Slot: L49+L50 (Lab)

1. Create a table students and set sname column with NOT NULL constraint
Students(sname, regno, dept, year,ccode,
Mark,average, grade)
Sname – NOT NULL

1.
```
SQL> create table students_23BDS1095 (sname varchar2(20) NOT NULL, regno varchar2(20), dept varchar2(20)
, year number, ccode varchar2(20), Mark number, average number, grade varchar2(1));

Table created.

SQL>
```

2. Display the structure of the table students

2.
```
SQL> desc students_23BDS1095;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 SNAME                                     NOT NULL VARCHAR2(20)
 REGNO                                              VARCHAR2(20)
 DEPT                                               VARCHAR2(20)
 YEAR                                               NUMBER
 CCODE                                              VARCHAR2(20)
 MARK                                               NUMBER
 AVERAGE                                            NUMBER
 GRADE                                              VARCHAR2(1)

SQL>
```

3. Insert a record in the students table with 23BDS1001

3.
```
SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average, grade) values
 ('Raj', '23BDS1001', 'SCOPE', 2, 'BCSE210P', 80, 80, 'A');

1 row created.

SQL>
```

4. Insert a record with empty value in sname column and identify the kind of error

4.
```
SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average,
grade) values ('', '23BDS4545', 'SAS', 2, 'BMAT454K', 90, 90, 'S');
insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average, grade
) values ('', '23BDS4545', 'SAS', 2, 'BMAT454K', 90, 90, 'S')

          *
ERROR at line 1:
ORA-01400: cannot insert NULL into ("SYSTEM"."STUDENTS_23BDS1095"."SNAME")


SQL> -- the error is there due to NOT NULL constraint on the sname column because an
empty value is being inserted...
SQL>
```

5. Add primary key constraint - Regno column as Primary Key

5.
```
SQL> alter table students_23BDS1095 add CONSTRAINT pk_regno PRIMARY KEY (regno);

Table altered.

SQL>
```

6. Insert a record with same reg.no 23BDS1001 and identify the violation

6.
```
SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average,
grade) values ('Kanishk', '23BDS1001', 'MECH', 2, 'BMCK4454', 90, 90, 'S');
insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average, grade
) values ('Kanishk', '23BDS1001', 'MECH', 2, 'BMCK4454', 90, 90, 'S')
*
ERROR at line 1:
ORA-00001: unique constraint (SYSTEM.PK_REGNO) violated


SQL> -- here we get an error due to violation of PRIMARY KEY constraint | because the
regno 23BDS1001 already exists as a primary key
```

7. Add check constraint for Mark column with constraint name and mark between
1 and 100

7.
```
SQL> alter table students_23BDS1095 add CONSTRAINT marks_valid_range CHECK (Mark between 1 AND 100);

Table altered.

SQL>
```

8. Insert two records in students table with mark as -5 in one record and 110 in another record. Find the appropriate violation.

8.
```
SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average,
grade) values ('Shree', '23BDS7878', 'MECH', 2, 'BMCK4454', -5, -5, 'S');
insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average, grade
) values ('Shree', '23BDS7878', 'MECH', 2, 'BMCK4454', -5, -5, 'S')
*
ERROR at line 1:
ORA-02290: check constraint (SYSTEM.MARKS_VALID_RANGE) violated


SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average,
grade) values ('Sahay', '23BDS4578', 'MECH', 2, 'BMCK4454', 110, 110, 'S');
insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, average, grade
) values ('Sahay', '23BDS4578', 'MECH', 2, 'BMCK4454', 110, 110, 'S')
*
ERROR at line 1:
ORA-02290: check constraint (SYSTEM.MARKS_VALID_RANGE) violated


SQL> -- error because Marks are not in range as specified for the CHECK constraint
SQL>
```

9. Add default constraint (0 as default value) to the column Average in the students table

9.

```
SQL> alter table students_23BDS1095 modify average DEFAULT 0;

Table altered.

SQL>
```

10. Insert a record into students without giving any value in Average column

10.

```
SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, grade) values
 ('Sanjeev', '23BDS1111', 'AIML', 2, 'BAI54744', 95, 'S');

1 row created.

SQL>
```

11. Display the students detail and to know the functionality of default constraint
with default value

11.

```
SQL> select * from students_23BDS1095;

SNAME                REGNO                DEPT                      YEAR CCODE                    MARK    AVERAGE G
-------------------- -------------------- -------------------- ---------- -------------------- ---------- ---------- -
Raj                  23BDS1001            SCOPE                         2 BCSE210P                     80         80 A
Sanjeev              23BDS1111            AIML                          2 BAI54744                     95          0 S

SQL>
```

12. Create another table course as per the schema,
Course(cname,ccode)
Cname- Not null constraint
Ccode – Primary key constraint with constraint name

12.

```
SQL> create table course_23BDS1095 (cname varchar2(20) NOT NULL, ccode varchar2(20) CONSTRAINT pk_code PRIMARY KEY);

Table created.

SQL>
```

13. Display the structure of the course table

13.

```
SQL> desc course_23BDS1095;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------------------------------------------------------
 CNAME                                     NOT NULL VARCHAR2(20)
 CCODE                                     NOT NULL VARCHAR2(20)

SQL>
```

14. Insert 5 records in the course table

```
SQL> insert into course_23BDS1095 (cname, ccode) values ('DSA', 'BCSE101L');

1 row created.

SQL> insert into course_23BDS1095 (cname, ccode) values ('OS', 'BCSE201L');

1 row created.

SQL> insert into course_23BDS1095 (cname, ccode) values ('OOPS', 'BCSE301L');

1 row created.

SQL> insert into course_23BDS1095 (cname, ccode) values ('Compilers', 'BCSE401L');

1 row created.

SQL> insert into course_23BDS1095 (cname, ccode) values ('Game Engines', 'BCSE501L');

1 row created.

SQL>
```

14.

15. Display the records from course table

15.

```
SQL> select * from course_23BDS1095;

CNAME                CCODE
-------------------- --------------------
DSA                  BCSE101L
OS                   BCSE201L
OOPS                 BCSE301L
Compilers            BCSE401L
Game Engines         BCSE501L

SQL>
```

16. Now, add foreign key constraint to the column Ccode in the table students.

16.

```
SQL> alter table students_23BDS1095 add CONSTRAINT fk_ccode FOREIGN KEY (ccode) REFERENCES course_23BDS1095 (ccode);
alter table students_23BDS1095 add CONSTRAINT fk_ccode FOREIGN KEY (ccode) REFERENCES course_23BDS1095 (ccode)
                                              *
ERROR at line 1:
ORA-02298: cannot validate (SYSTEM.FK_CCODE) - parent keys not found


SQL> -- the output is an error message to this query
SQL>
```

17. Identify the type of error from the previous query and find the solution to add
foreign key

17.

```
SQL> -- there will be an error in the previous query because t
he foreign key will fail if there are any existing values in t
he students table where the ccode doesn't match any value in t
he course table
SQL> -- we need to make sure that all the values in the ccode
column in students match a value in the ccode column of course
SQL> -- a solution to the error could be to insert valid ccode
 values into the students table before adding the foreign key
SQL>
```

After solution

```
SQL> insert into course_23BDS1095 (cname, ccode) values ('Game Engines', 'BCSE210P');

1 row created.

SQL> insert into course_23BDS1095 (cname, ccode) values ('Game Engines', 'BAI54744');

1 row created.

SQL> alter table students_23BDS1095 add CONSTRAINT fk_ccode FOREIGN KEY (ccode) REFERENCE
S course_23BDS1095 (ccode);

Table altered.

SQL>
```

18. Insert 2 records in the students table. Enter the value of ccode of student table
that matches the value of ccode of Course table.

18.

```
SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, grade) values ('Sanjeev', '23BDS7415', 'AIML', 2,
 'BCSE101L', 95, 'S');

1 row created.

SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, grade) values ('Sanjeev', '23BDS1097', 'AIML', 2,
 'BCSE201L', 95, 'S');

1 row created.

SQL>
```

19. Insert 1 record in the students table. Enter the value of ccode of student table
that does not match the value of ccode of Course table. Recognize the type of
violation

19.

```
SQL> insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, grade) va
lues ('Sanjeev', '23BDS1095', 'AIML', 2, 'BCSE444L', 95, 'S');
insert into students_23BDS1095 (sname, regno, dept, year, ccode, Mark, grade) values
('Sanjeev', '23BDS1095', 'AIML', 2, 'BCSE444L', 95, 'S')
*
ERROR at line 1:
ORA-02291: integrity constraint (SYSTEM.FK_CCODE) violated - parent key not found


SQL> -- this is a foreign key constrain violation since the ccode 'BCSE444l' does not
 previously exits in the course table
```

20. Drop the check constraint

20.

```
SQL> alter table students_23BDS1095 drop CONSTRAINT marks_valid_range;

Table altered.

SQL>
```

21. Delete all the records from course table.

21.

```
SQL> delete from course_23BDS1095;
delete from course_23BDS1095
*
ERROR at line 1:
ORA-02292: integrity constraint (SYSTEM.FK_CCODE) violated - child record found


SQL> -- an error message is the output to the above query
SQL>
```

22. Identify the type of violation from the previous query and find the solution to

delete all the records from course table

Error:

- A foreign key violation will occur because there are records in the `students` table that reference the `ccode` in the `course` table.

Solution:

- You must either delete the dependent records in `students` first or modify the foreign key constraint to `ON DELETE CASCADE`, which automatically deletes the dependent records in `students` when a record in `course` is deleted.

```
SQL> alter table students_23BDS1095 drop CONSTRAINT fk_ccode;

Table altered.
```

```
SQL> alter table students_23BDS1095 add CONSTRAINT fk_ccode FOREIGN
 KEY (ccode) REFERENCES course_23BDS1095 (ccode) ON DELETE CASCADE;

Table altered.

SQL> delete from course_23BDS1095;

7 rows deleted.

SQL>
```