

## Laboratory 4

Title of the Laboratory Exercise: implement constraints and built in functions

### 1. Introduction and Purpose of Experiment

Constraints are the rules which are enforced on the data being stored in a table. There are constraints that can be applied to a table such as NOT NULL, UNIQUE, PRIMARY KEY and FOREIGN KEY. SQL has many built-in functions for performing calculations on data. In SQL, a built-in function is a piece for programming that takes zero or more inputs and returns a value. By doing this lab, students will be able to implement constraints and built in functions on the database.

### 2. Aim and Objectives

Aim

- To design and implement constraints on the data using SQL commands
- To implement built in functions in SQL

Objectives

At the end of this lab, the student will be able to

- Identify different types of constraints on the data
- Apply constraints on the data in different ways
- Implement built-in functions in SQL

### 3. Experimental Procedure

- i. Analyse the problem statement
- ii. Design SQL commands using appropriate constraints
- iii. Execute the SQL commands
- iv. Test the executed commands
- v. Document the Results
- vi. Analyse and discuss the outcomes of your experiment

### 4. Questions

Consider the relational schema given below.

PLAYER\_DETAILS (PCode, PName, DOB, City, Score)

MATCH\_DETAILS (MatchID, MatchName, PCode)

a

- a. Apply the following constraints on the given database schema. Enter appropriate tuples to show the purpose of each constraint.
- Not NULL
  - Default
  - Unique
  - Primary key
  - Foreign key
- b. Execute the following built-in functions in SQL using NetBeans IDE
- String functions
  - Date functions
  - Numeric functions

## 5. Presentation of Results

### Creating tables for the given problem statement

#### 1. Player Details

```
mysql> create table player_details(PCode int not null auto_increment,
-> PName varchar(40) not null,
[
-> DOB Date,
[
-> City varchar(40) default "Bangalore",
[
-> score int,
[
-> primary key (Pcode));
Query OK, 0 rows affected (0.02 sec)

mysql> desc player_details;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| PCode | int           | NO   | PRI | NULL    | auto_increment |
| PName | varchar(40)   | NO   |     | NULL    |                |
| DOB   | date          | YES  |     | NULL    |                |
| City  | varchar(40)   | YES  |     | Bangalore |                |
| score | int           | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
```

Figure 1 MySQL Query to Create a table "Player Details" and Describing the table.

#### 2. Match Details

```
mysql> create table match_details(
[
-> MatchID int unique,
[
-> MatchName varchar(40) not null,
[
-> PCode int, foreign key (PCode) references player_details (Pcode) on delete cascade);
Query OK, 0 rows affected (0.02 sec)

mysql> desc match_details;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| MatchID    | int           | YES  | UNI | NULL    |                |
| MatchName  | varchar(40)   | NO   |     | NULL    |                |
| PCode      | int           | YES  | MUL | NULL    |                |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

Figure 2 MySQL Query to Create a table "Match Details" and Describing the table.

**a. Not NULL**

```
[mysql> insert into player_details(DOB,Score,city) values ('2000-03-10',1000,"Delhi");
ERROR 1364 (HY000): Field 'PName' doesn't have a default value
```

Figure 3 MySQL Constraint Error for "Not Null"

**b. Default**

```
[mysql> insert into player_details(PName,DOB,Score) values ("Supraja",'1999-03-17',1000);
Query OK, 1 row affected (0.00 sec)
```

Figure 4 MySQL Query for Inserting data into table with default column

```
mysql> select * from player_details;
+-----+-----+-----+-----+-----+
| PCode | PName      | DOB       | City      | score |
+-----+-----+-----+-----+-----+
| 1     | K Srikanth  | 2000-01-23 | Bangalore | 1000  |
| 2     | Naveen Kumar | 2000-03-10 | Bangalore | 1000  |
| 3     | Naveen Kumar | 2000-03-10 | Delhi     | 1000  |
| 4     | Sushanth    | 2000-03-10 | Bombay    | 1000  |
| 5     | Supraja     | 1999-03-17 | Bangalore | 1000  |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Figure 5 MySQL Query to print the table "Players details" and showing an default value from the query figure 4

**c. Unique**

```
mysql> insert into match_details values (1,'Bangalore Stadium',4);
ERROR 1062 (23000): Duplicate entry '1' for key 'match_details.MatchID'
```

Figure 6 MySQL Constraint Error for "Unique"

**d. Primary key**

Field	Type	Null	Key	Default	Extra
PCode	int	NO	PRI	NULL	auto_increment

Figure 7 MySQL Table displaying "PCode" as a primary Key.

**e. Foreign key**

```
mysql> insert into match_details values (10,'Bangalore Stadium',10);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`lab_4`.`match_details`, CONSTRAINT `match_details_ibfk_1` FOREIGN KEY (`PCode`) REFERENCES `player_details` (`PCode`) ON DELETE CASCADE)
```

Figure 8 MySQL Constraint Error for "Foreign Key"

b. Execute the following built-in functions in SQL using NetBeans IDE but I used VS Code.

Making a connection on VS Code

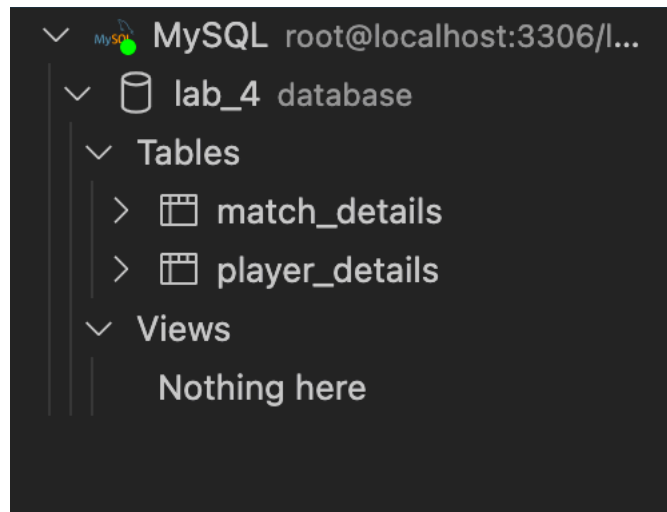


Figure 9 Making a MySQL Server Connection on VS Code Editor and connecting to database "lab\_4"

#### a. String functions

- Upper ()

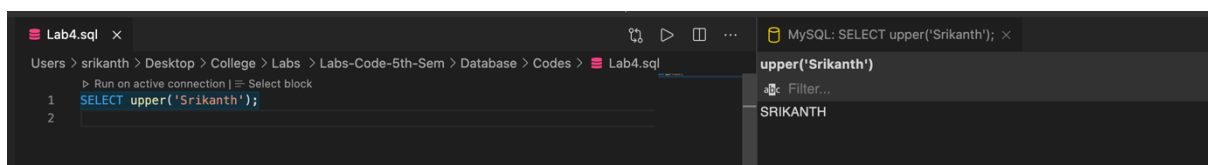


Figure 10 MySQL Query using Upper() inbuilt function.

- Lower ()

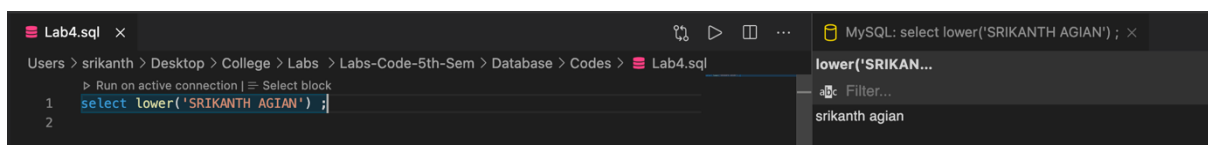


Figure 11 MySQL Query using Lower() inbuilt function

- Concat ()

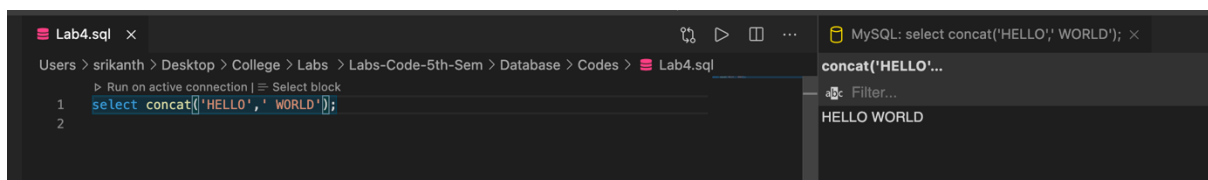


Figure 12 MySQL Query using Concat() inbuilt function

- Replace ()

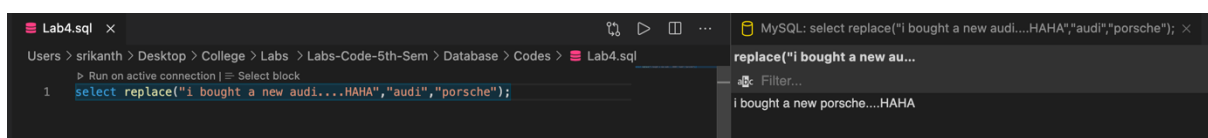
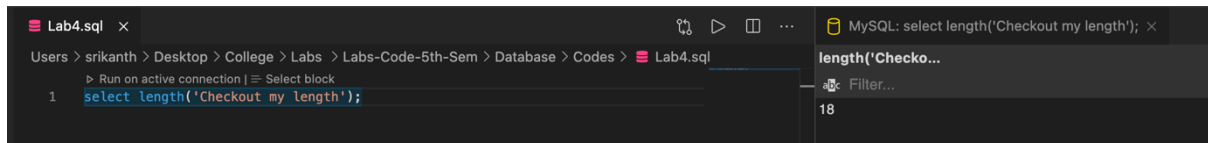


Figure 13 MySQL Query using Replace() inbuilt function

- **Length ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the following SQL statement:

```
1 select length('Checkout my length');
```

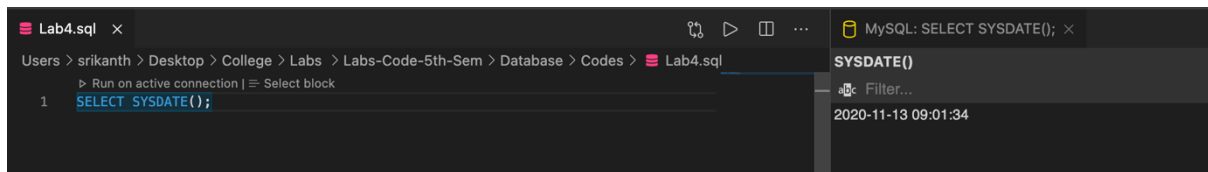
The output pane on the right shows the result of the query:

length('Checko...
18

Figure 14 MySQL Query using Length() inbuilt function

**b. Date functions**

- **Sysdate ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the following SQL statement:

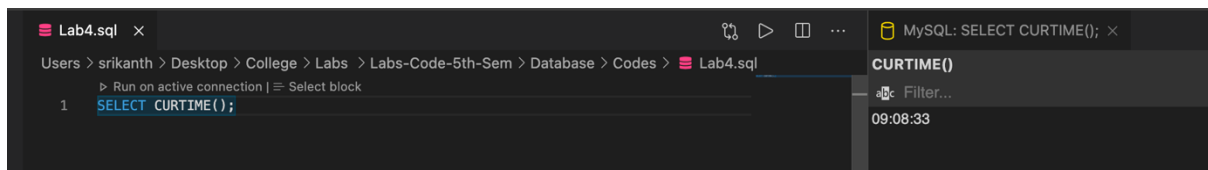
```
1 SELECT SYSDATE();
```

The output pane on the right shows the result of the query:

SYSDATE()
2020-11-13 09:01:34

Figure 15 MySQL Query using Sysdate () inbuilt function

- **Curtime ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the following SQL statement:

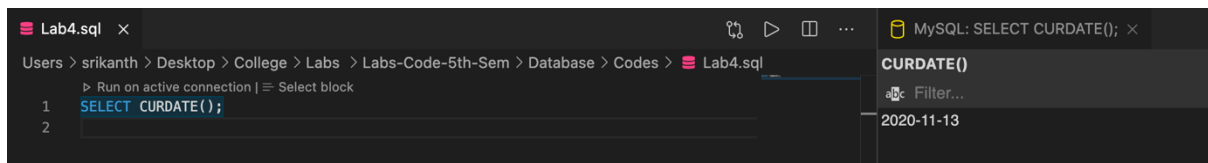
```
1 SELECT CURTIME();
```

The output pane on the right shows the result of the query:

CURTIME()
09:08:33

Figure 16 MySQL Query using Curtime() inbuilt function

- **Curdate ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the following SQL statement:

```
1 SELECT CURDATE();
```

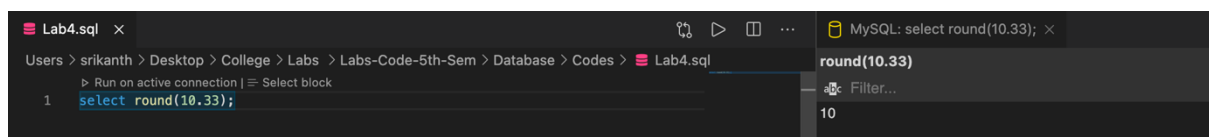
The output pane on the right shows the result of the query:

CURDATE()
2020-11-13

Figure 17 MySQL Query using Curdate () inbuilt function

**c. Numeric functions**

- **Round ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the following SQL statement:

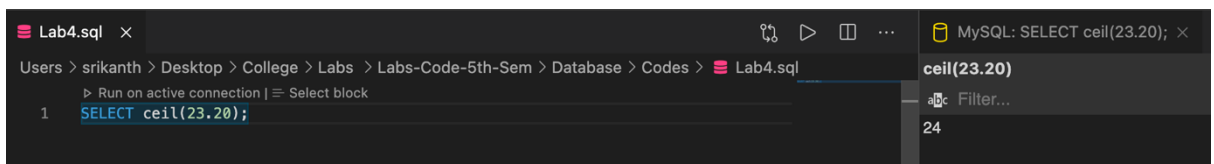
```
1 select round(10.33);
```

The output pane on the right shows the result of the query:

round(10.33)
10

Figure 18 MySQL Query using Round () inbuilt function

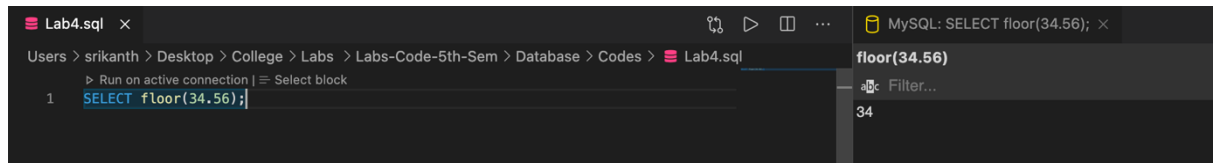
- **Ceil ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the SQL statement `SELECT ceil(23.20);`. The output pane on the right displays the result of the query, which is `ceil(23.20)` followed by the value `24`.

Figure 19 MySQL Query using Ceil () inbuilt function

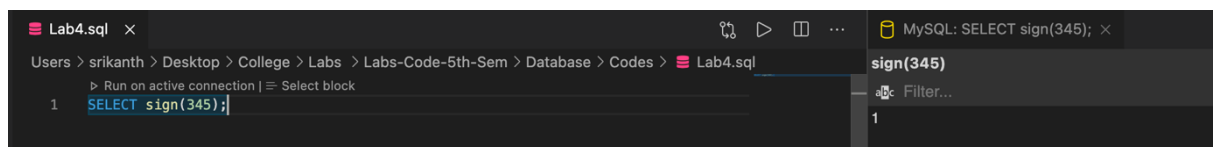
- **Floor ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the SQL statement `SELECT floor(34.56);`. The output pane on the right displays the result of the query, which is `floor(34.56)` followed by the value `34`.

Figure 20 MySQL Query using Floor () inbuilt function

- **Sign ()**

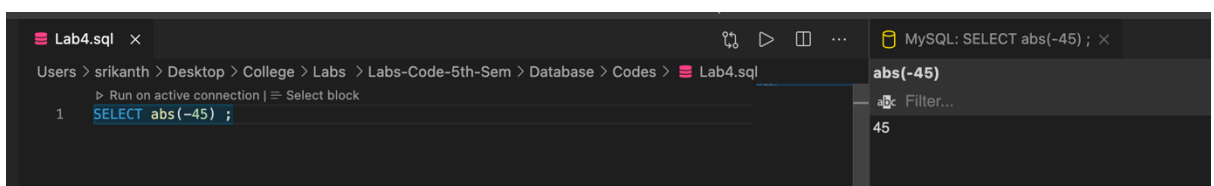


The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the SQL statement `SELECT sign(345);`. The output pane on the right displays the result of the query, which is `sign(345)` followed by the value `1`.

Figure 21 MySQL Query using Sign () inbuilt function

## Math Functions

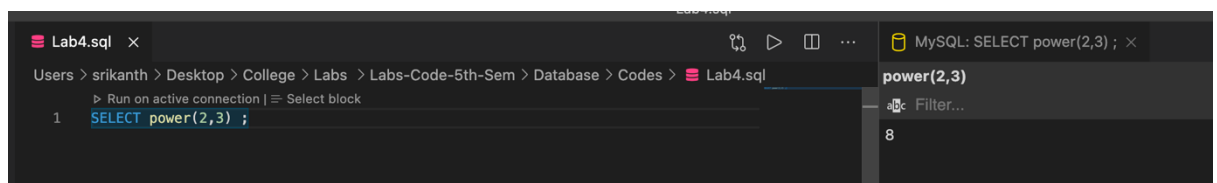
- **Abs ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the SQL statement `SELECT abs(-45);`. The output pane on the right displays the result of the query, which is `abs(-45)` followed by the value `45`.

Figure 22 MySQL Query using abs () inbuilt function

- **Power ()**



The screenshot shows a MySQL IDE window with a file named 'Lab4.sql'. The query editor contains the SQL statement `SELECT power(2,3);`. The output pane on the right displays the result of the query, which is `power(2,3)` followed by the value `8`.

Figure 23 MySQL Query using Power () inbuilt function

- **Mod ()**

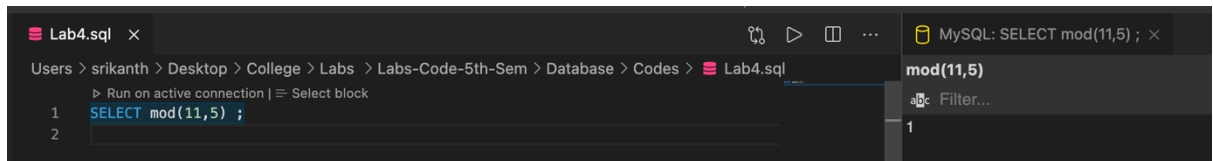


Figure 24 MySQL Query using Mod () inbuilt function

- **Sqrt ()**

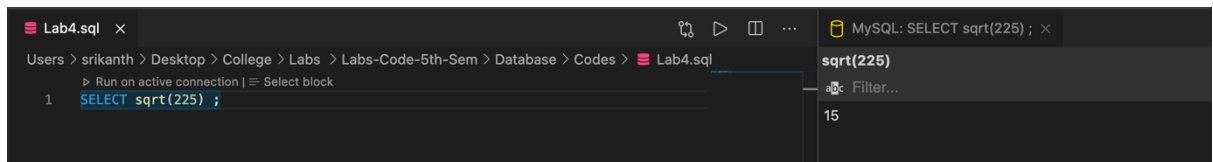


Figure 25 MySQL Query using Sqrt () inbuilt function

## 6. Analysis and Discussions

### Sub Question – A

#### a. Not NULL

This is a constraint used when a data field is required to have a data and can't be empty

**Reference Image : Image 3**

#### b. Default

This is a constraint used when we have a default value while creating a table so when the user doesn't insert any value into this particular column then it will consider the default value from when we create the table

**Reference Image : Image 4 and Image 5**

#### c. Unique

This is a constraint used when the data has to be unique and also when we don't want any duplicity of the data in our table.

**Reference Image : Image 6**

#### d. Primary key

This is key constraint used to define a key for an entire data set of a row in a table because when the data has to be called in other table for a reference then it can be called using this key

**Reference Image : Image 7**

#### e. Foreign key

This is key constraint used to define a key that was a primary key from table 1 and wants to be used in a table 2 so the primary key from table 1 acts a foreign key in table 2 but if the key doesn't exist then it will give you an error

**Reference Image : Image 8**

## **Sub Question – B**

### **a. String functions**

- **Upper ()**

This is a inbuilt function used to change all the lowercase letters to uppercase letters .

**Reference Image : Image 10**

- **Lower ()**

This is a inbuilt function used to change all the uppercase letters to lowercase letters .

**Reference Image : Image 11**

- **Concat ()**

This is a inbuilt function used to concatenate strings.

**Reference Image : Image 12**

- **Replace ()**

This is a inbuilt function used to replace strings and this function takes three arguments firstly the string followed by the string for changing and the string that its being changing to .

**Reference Image : Image 13**

- **Length ()**

This is a inbuilt function is used to find the length of a string.

**Reference Image : Image 14**

### **b. Date functions**

- **Sysdate ()**

This is a inbuilt function which returns the system date in format YYYY/MM/DD along with time in format H/M/S/MS

**Reference Image : Image 15**

- **Curtime ()**

This is a inbuilt function which returns the system current time in format H/M/S/MS

**Reference Image : Image 16**

- **Curdate ()**

This is a inbuilt function which returns the system current date in format YYYY/MM/DD

**Reference Image : Image 17**



### c. Numeric functions

- **Round ()**

This is a inbuilt function which returns the rounded value of float or double .

**Reference Image : Image 18**

- **Ceil ()**

This is a inbuilt function which returns the ceiling value of a number.

**Reference Image : Image 19**

- **Floor ()**

This is a inbuilt function which returns the Floor value of a number.

**Reference Image : Image 20**

- **Sign ()**

This is a inbuilt function which returns the sign of the number if the number is positive then it will returns "1" and if it's not then "-1"

**Reference Image : Image 21**

### Math Functions

- **Abs ()**

This is a inbuilt function which returns the absolute value of a number.

**Reference Image : Image 22**

- **Power ()**

This is a inbuilt function which returns the power of integer and this function takes two arguments one is the base followed by power

**Reference Image : Image 23**

- **Mod ()**

This is a inbuilt function which returns the remainder of a number divided by another number this takes two arguments firstly the value which is going to be divided by the divisor and secondly the divisor

**Reference Image : Image 24**

- **Sqrt ()**

This is a inbuilt function which returns the square root of a integer.

**Reference Image : Image 25**

## **7. Conclusions**

In a relational database, digital information is organized into rows, columns and tables which are indexed to make it easier to find relevant information through SQL. Constraints are specified to limit the data in the tables. In-built functions provide extended set of operations on data.

## **8. Comments**

### **Learning happened**

- To design and implement constraints on the data using SQL commands
- To implement built in functions in SQL.