**Remote Terminal Unit Database**

**Participants are instructed to create following tables with appropriate data types, primary key, foreign key, not null and check constraints.**

**Insert data as shown below into respective tables**

**Table - RTU**

|  |  |  |
| --- | --- | --- |
| **RTU\_ID** | **LOCATION** | **IP\_ADDRESS** |
| R101 | BENGALURU | 192.168.0.1 |
| R102 | MUMBAI | 192.168.0.2 |
| R103 | CHENNAI | 192.168.0.3 |
| R104 | NEW DELHI | 192.168.0.4 |
| R105 | KOLKATA | 192.168.0.5 |

**Table – SENSORS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SENSOR\_ID** | **RTU\_ID** | **STATUS** | **SENSOR\_TYPE** | **REQ\_VAL** | **CURR\_VAL** |
| S801 | R101 | ENABLED | PRESSURE | 34 | 33 |
| S802 | R101 | ENABLED | TEMPERATURE | 98 | 90 |
| S803 | R101 | DISABLED | TEMPERATURE | 96 |  |
| S804 | R102 | ENABLED | PRESSURE | 30 | 32 |
| S805 | R102 | ENABLED | PRESSURE | 36 | 32 |
| S806 | R102 | DISABLED | TEMPERATURE | 108 |  |
| S807 | R102 | ENABLED | BAROMETER | 750 | 660 |
| S808 | R103 | ENABLED | BAROMETER | 780 | 690 |
| S809 | R103 | ENABLED | ULTRASONIC | 510 | 450 |
| S810 | R104 | DISABLED | ULTRASONIC | 530 |  |
| S811 | R104 | ENABLED | PRESSURE | 60 | 50 |

**Table – USERS**

|  |  |  |
| --- | --- | --- |
| **USER\_ID** | **USER\_NAME** | **ROLE** |
| U501 | PRAKASH | MONITOR |
| U502 | PARTHIV | ADMINISTER |
| U503 | SMITHA | ADMINISTER |

**Table – CONTROLLER**

|  |  |
| --- | --- |
| **USER\_ID** | **SENSOR\_ID** |
| U501 | S801 |
| U501 | S804 |
| U501 | S806 |
| U502 | S802 |
| U502 | S804 |
| U503 | S805 |
| U503 | S806 |
| U501 | S809 |
| U502 | S810 |
| U503 | S811 |
| U501 | S807 |
| U503 | S808 |
| U502 | S801 |
| U501 | S808 |

**Write following queries based on above tables**

Simple Queries

1. Fetch details of all the sensors which are disabled
2. List sensors which are having required value less than current value
3. List sensors which are of type TEMPERATURE and required value greater than 96. Sort on required value field.
4. List sensors which has no current values
5. List all sensors which are of type TEMPERATURE & PRESSURE.
6. List sensors with rtu\_id 102 and sensor type pressure and required value greater than 30.

**Single Row Functions**

1. Write a query which will display username in upper case, role first letter capitalized and all other letters lower case and length of the username.
2. Display percentage of difference between req\_value and curr\_value as a whole number.

**Group Functions**

1. List how many sensors are present of type PRESSURE.
2. Display number of sensors enabled and disabled. Sort on number of sensors
3. List how many sensors are present in each sensor type.
4. List number of sensors under each RTU\_ID.
5. Display how many sensors are controlled by each user. Exclude users who are controlling less than 3 sensors and also restrict users with id “U503”. Sort on user\_id.

**JOINS**

1. Display RTU details alongwith sensor details for all RTU's
2. Display user details and sensor details controlled by each user.
3. Display user details and sensor details controlled by each user alongwith RTU details.
4. List how many sensors are controlled by each user. Display user\_name and count. Sort on count.
5. Display user details who are controlling sensor\_type TEMPERATURE
6. Display user details that is controlling sensor S801 along with sensor details and location.

**SUB-QUERIES**

1. Display sensor details located in CHENNAI
2. Display user details controlling sensors in MUMBAI location
3. Display sensor details controlled by SMITHA
4. Display user details who are controlling sensors with id between S803 and S808
5. Display user details that are controlling most number of sensors.
6. Display RTU details which has most number of sensors.