**Question1 (EXAM\_RESULTS table)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STUDENT\_ID** | **FIRST\_NAME** | **LAST\_NAME** | **EXAM\_ID** | **EXAM\_SCORE** |
| 10 | LAURA | LYNCH | 1 | 90 |
| 10 | LAURA | LYNCH | 2 | 85 |
| 11 | GRACE | BROWN | 1 | 78 |
| 11 | GRACE | BROWN | 2 | 72 |
| 12 | JAY | JACKSON | 1 | 95 |
| 12 | JAY | JACKSON | 2 | 92 |
| 13 | WILLIAM | BISHOP | 1 | 70 |
| 13 | WILLIAM | BISHOP | 2 | 100 |
| 14 | CHARLES | PRADA | 2 | 85 |

Write the SQL queries for the following statements:

|  |  |
| --- | --- |
| 1. SQL statement to find the average exam score for EXAM\_ID = 1? | |
| 2. SQL statement to find out number of students attended each exam? |  |

3. SQL statement to find the maximum score of EXAM\_ID=1 and FIRST\_NAME has the letter “E”.

4. SQL statement to find the maximum EXAM\_SCORE for EXAM\_ID=1 which is greater than the average

marks of that EXAM\_ID.

For e.g. for EXAM\_ID=1 available scores are 90,78,95,70 whose average is

(90+78+95+70)/4=**83.25**. So records selected will be 90 and 95 as both are >83.25. Out of these two **95** is the final score as it is maximum.

**Question2 (Users table)**

A website maintains two tables as below:

|  |  |
| --- | --- |
| **USERS** |  |
|  | userid |
|  | username |
|  | contactnumber |

And another table maintains the list of users visited the site:

|  |  |
| --- | --- |
|  |  |
|  | userid |
|  | date |
|  | task |

Where **“date”** is the date of visiting and **“task”** is the action done by the user in the site on visit.

Website owner wants to know the following information from the two above tables:

1. What are the users’ details who did not visited the site since a month? From recent date
2. All those users details who did not visited the site at all since starting? Not visited the site
3. How many users visited the site today i.e. current date but did not do any task?

**Question3: T-SQL Assignments**

Assume a person salary is 50000. He maintains a table **“tbl\_expense”** to store the daily expense records with columns **expense\_id, amount, date**. Whenever he does any expense, a record is added.

**1.** Create a stored procedure which will accept his expense amount and insert a record in the expense table with following details: **expense\_id, amount, date**.

**2.** Note that the stored procedure should monitor his expenses and reject adding those expenses which make his total sum of expenses (including the current expense being added) more than 50% of his salary.