Average mark greater than 80

Grade settings: Maximum grade: 20
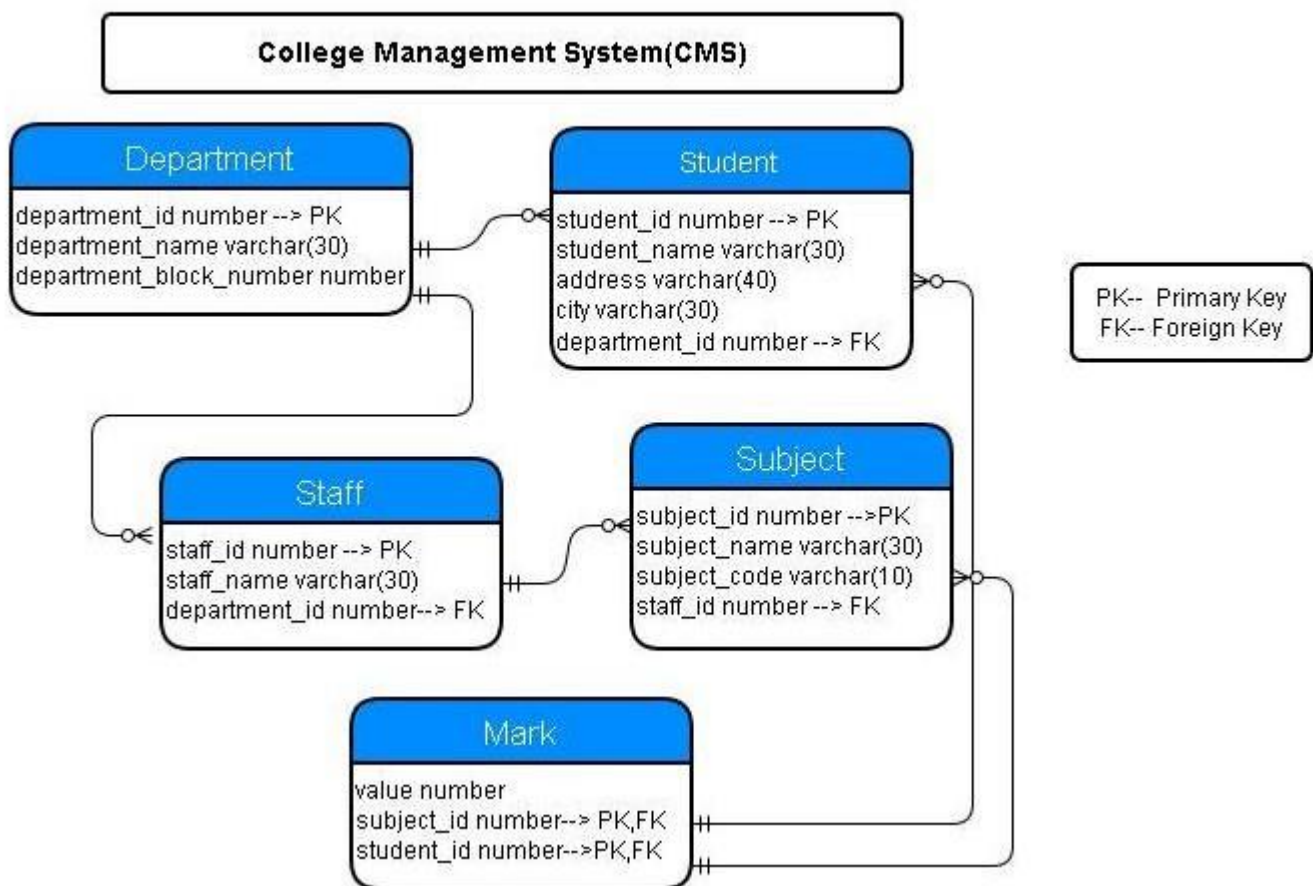**Based on**: DRL - Oracle
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

Write a query to display list of student ids and average mark in 2 decimal places if their average mark is greater than 80. Give an alias to average mark as avg_mark. Sort the result based on average mark.

Refer the below schema diagram:

Change Case of Guest Name
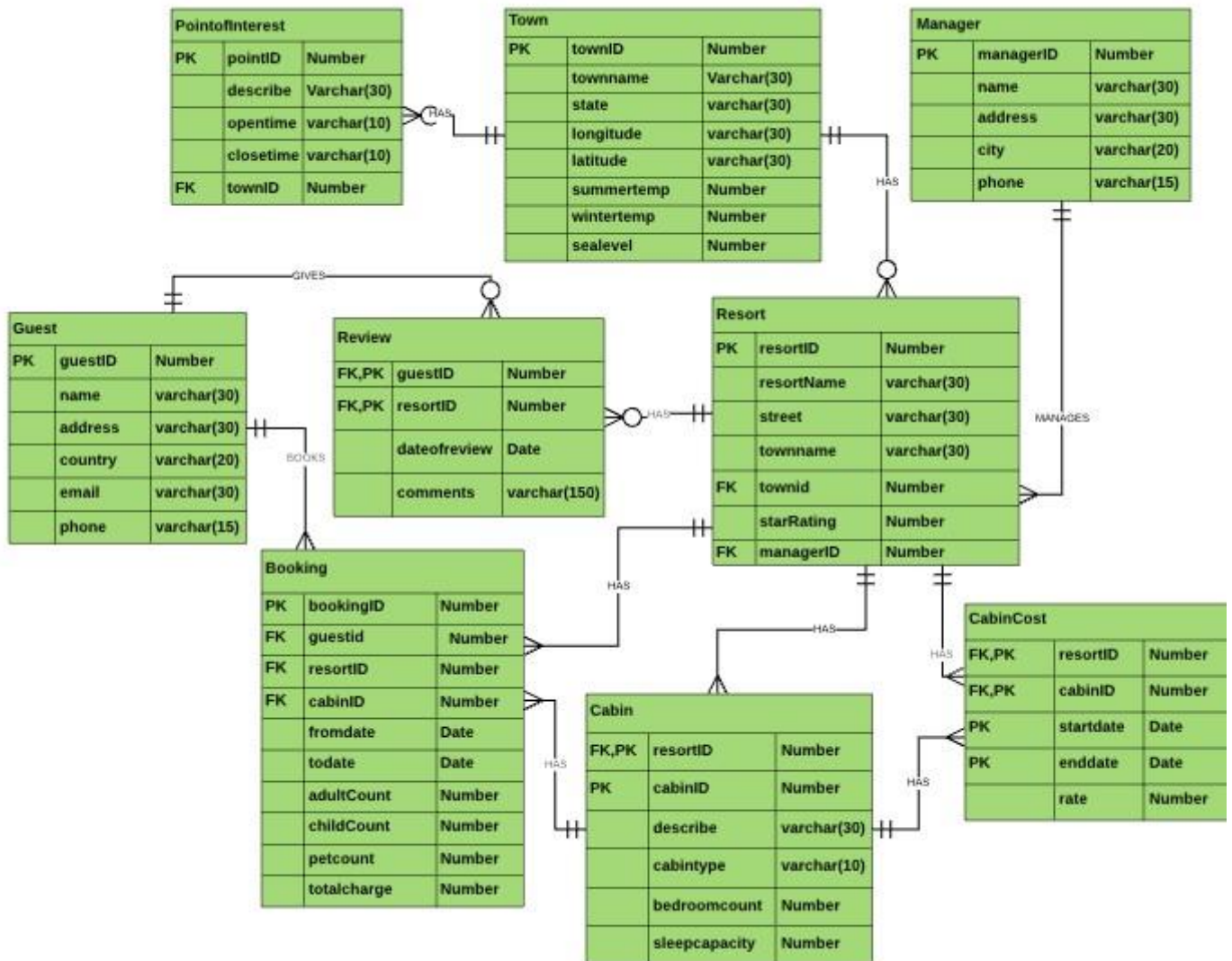
Grade settings: Maximum grade: 20
Based on: Meta - Select -Oracle
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

Write a query to display the guest name in proper case. For instance the guest name "geeta govind" should be displayed as "Geeta Govind". Give alias name as GUESTNAME. Sort the result based on Guest Name.

RESORT MANAGEMENT SYSTEM
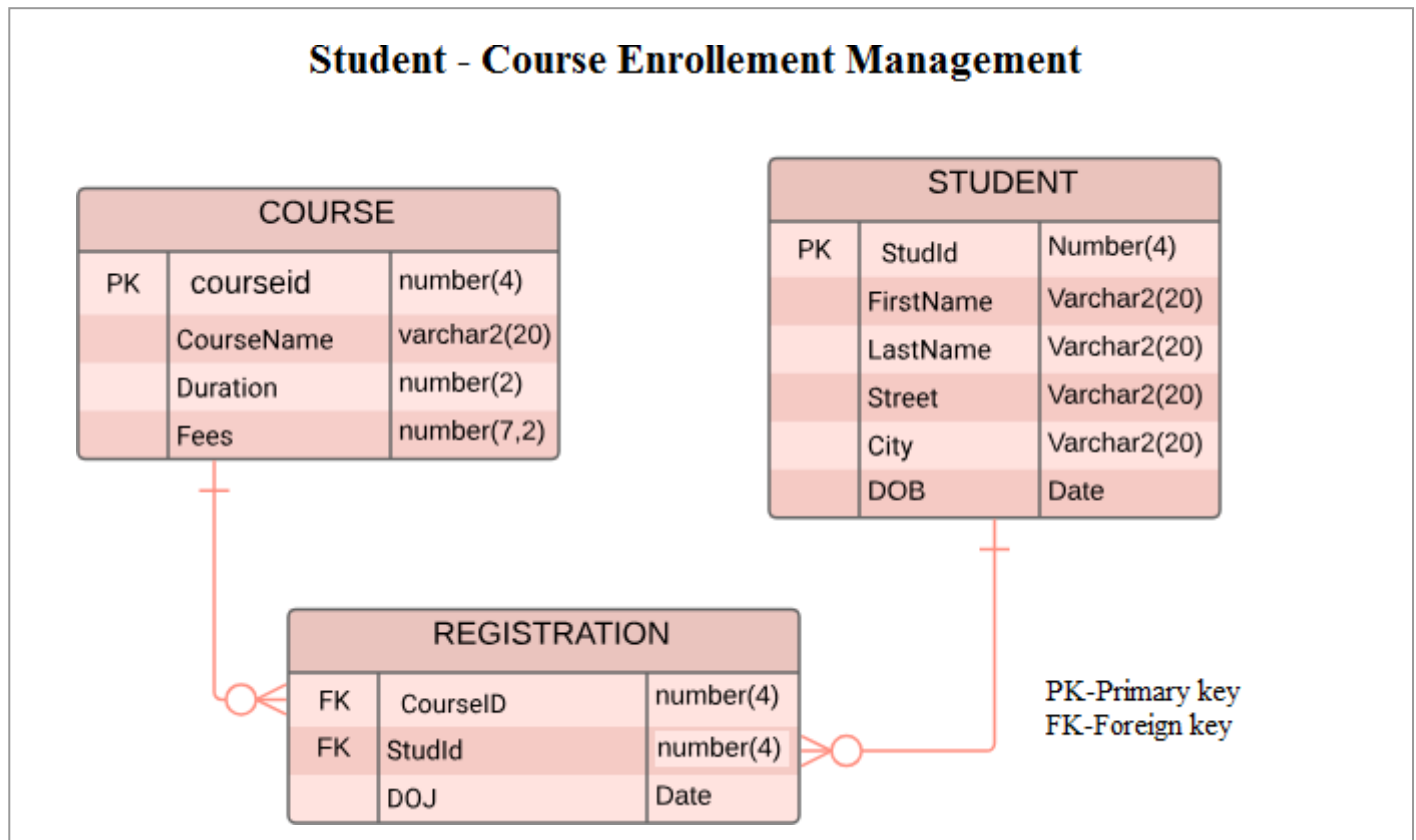
Latest date of registration

Grade settings: Maximum grade: 20
Based on: Meta - Select -Oracle
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

Write a query to display the Latest date with alias name as LATESTDATE in which registration happened.
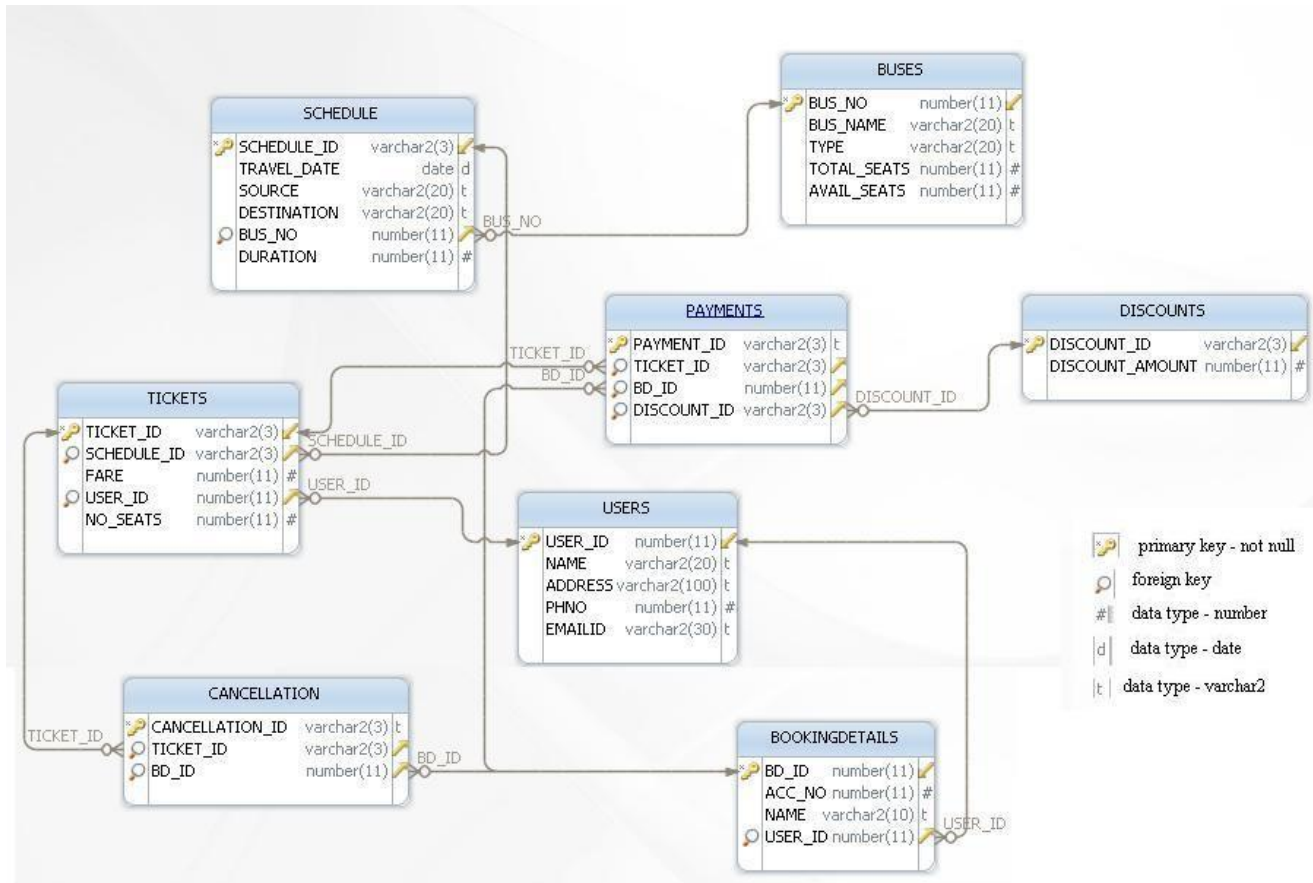
Maximum Bus Tickets Fare

Grade settings: Maximum grade: 20
Based on: DRL - Oracle
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

Write a query to display maximum bus tickets fare and give an alias name as Maximum_Amount

Number of courses Registered
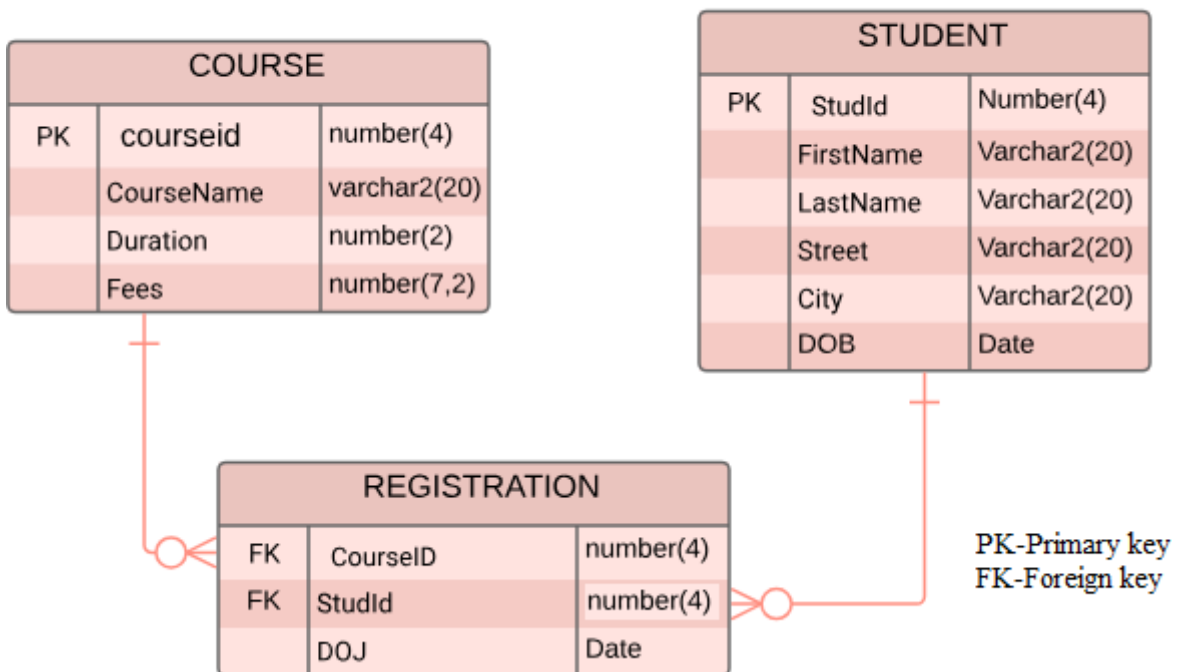
Grade settings: Maximum grade: 20
Based on: Meta - Select -Oracle
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

Write a query to display student ID and number of course registered by students. Display student ID only if student has registered for a course. Give alias name for the count as NOOFCOURSES. Sort the result based on count in descending and student ID ascending.

Average mark greater than 80

```sql
select student_id,round(avg(value),2) as avg_mark
from mark
group by student_id
having avg(value)>80
order by avg_mark;
```

Change Case of Guest Name

```sql
select INITCAP(name) as
GUESTNAME
from Guest
order by GUESTNAME;
```

Latest date of registration

```sql
select (MAX(DOJ)) as latestdate from registration;
```

Maximum Bus Tickets Fare

```sql
select max(fare) as Maximum_Amount
from tickets;
```

Number of courses Registered

```sql
select StudId,count(CourseId) as NOOFCOURSES
from registration
group by studId
order by NOOFCOURSES desc,StudId asc;
```