SQL Queries:

Calculate the average rating given by students to each teacher for each session created. Also, provide the batch name for which session was conducted.

```
select ss.id session_id,ss.conducted_by, b.name,avg(a.rating) from sessions ss
join batches b on b.id=ss.batch_id
join attendances a on ss.id=a.session_id
group by ss.id,ss.conducted_by,b.name;
```

Find the attendance percentage for each session for each batch. Also mention the batch name and users name who has conduct that session

```
with t0 as (select batches.id batch_id , count(student_batch_maps.id) as total_std
from batches join student_batch_maps on
batches.id=student_batch_maps.batch_id group by batches.id),
select ss.id session_id,u.name, b.name,(count(a.student_id)/t0.total_std)*100
attendance_percentage from sessions ss
join batches b on b.id=ss.batch_id
join attendances a on ss.id=a.session_id
join users u on u.id=ss.conducted_by
join t0 on t0.batch_id=ss.batch_id
group by ss.id,u.name, b.name;
```

What is the average marks scored by each student in all the tests the student had appeared?

```
select test_scores.user_id, avg(score) from test_scores
group by test_scores.user_id;
```

A student is passed when he scores 40 percent of total marks in a test. Find out how many students passed in each test. Also mention the batch name for that test.

```
select tests.id, batches.name, count(ts.user_id) from test_scores ts join tests on
ts.test_id=tests.id
join batches on tests.batch_id=batches.id
where (ts.score/tests.total_mark)*100 > 40
group by ts.test_id,batches.name;
```

A student can be transferred from one batch to another batch. If he is transferred from batch a to batch b. batch b's active=true and batch a's active=false in student_batch_maps. At a time, one student can be active in one batch only. One Student can not be transferred more than four times. Calculate each student's attendance percentage for all the sessions created for his past batch. Consider only those sessions for which he was active in that past batch.

```
with t0 as(
select * from
(select user_id, batch_id, deactivated_at, row_number() over(partition by user_id,
batch_id order by updated_at) rn from student_batch_maps sbm
where active='false')where rn=1),
t1 as (
select t0.user_id,sessions.id from t0 join sessions on t0.batch_id=sessions.batch_id
and sessions.start_time < t0.deactivated_at
),
t2 as(select t1.user_id, count(*) as std_count from attendances join t1 on
t1.id=attendances.session_id and t1.user_id=attendances.student_id group by
t1.user_id ),
t3 as ( select t0.batch_id, count(*) sessions_ct from t0 join sessions on
t0.batch_id=sessions.batch_id group by t0.batch_id)

select t0.user_id, (t2.std_count/t3.sessions_ct)*100 std_attendance_percentage
from t0 join t2 on t2.user_id=t0.user_id
join t3 on t0.batch_id=t3.batch_id
```