SQL

Part A

```
select rider_id, trip_id
from uber_trip u
where (select count(trip_id) from uber_trip
where rider_id=u.rider_id and trip_status='completed'
and begintrip_timestamp_utc<u.begintrip_timestamp_utc)=4
```

Part B

```
Select a.trip_id as trip.id, a.rider_id as rider_id, a.driver_id as driver_id,a.timestamp as initiated_ts, b.timestamp as cancel_ts, d.timestamp as complete_ts
Insert into dispatch_events
from (trip_initiated a left join trip_cancel b
on a.trip_id = b.trip_id )c
left join trip_completed d
on c.trip_id=d.trip_id
```

Part C

Test query 1:

```
Select count(*)
From dispatch_events
Where cancel_ts is NULL
and complete_ts is NULL
```

For this query, I expect it returns 0 since all the trips should either result in cancel or complete status so no rips should miss both cancel_ts and complete_ts.

Test query 2:

```
Select count(*)
From dispatch_events
Where cancel_ts is NULL
And trip_id not in (select trip_id from trip_complete)
```

For this query, I expect it returns 0 since every trip that has NULL values in cancel_ts column should be completed trips so they should all be in the trip_complete table it the dispatch_events table are created correctly.

Test query 3:

```
Select count(*)
From dispatch_events
Where complete_ts is NULL
And trip_id not in (select trip_id from trip_cancel)
```

For this query, I expect it returns 0 since every trip that has NULL values in complete_ts column should be canceled trips so they should all be in the trip_cancel table it the dispatch_events table are created correctly.