Query 1 Testing

Code tested:

SELECT STAFF.NAME AS STAFF\_NAME, STAFF.WORKPHONENO AS WORK\_PHONE, STAFF.HOMEPHONENO AS HOME\_PHONE, Building.buildingname AS BUILDING

FROM STAFF

INNER JOIN LOCATION

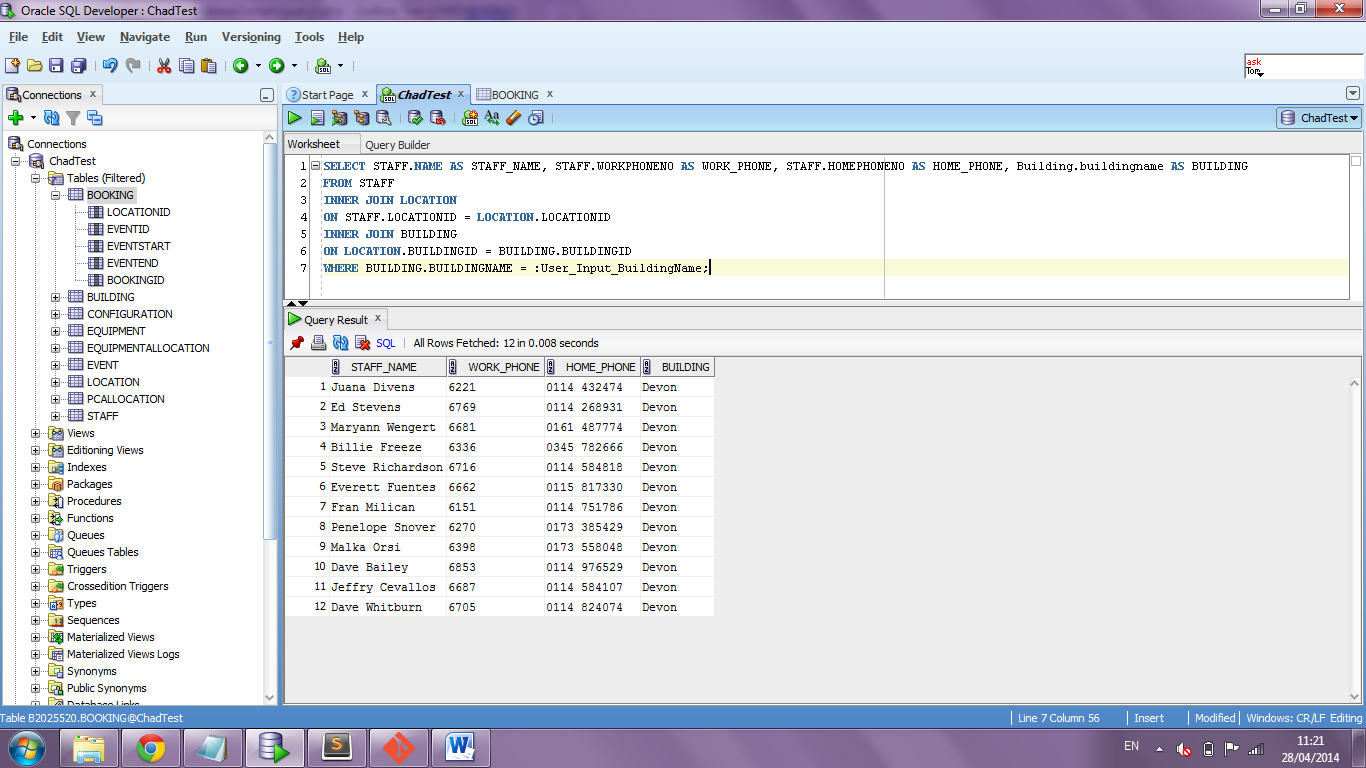
ON STAFF.LOCATIONID = LOCATION.LOCATIONID

INNER JOIN BUILDING

ON LOCATION.BUILDINGID = BUILDING.BUILDINGID

WHERE BUILDING.BUILDINGNAME = :User\_Input\_BuildingName;

:User\_Input\_BuildingName = ‘Devon’

Result:

When tested in SQL Developer the returned results are as expected meeting the critera of the query. For testing perposes the bulding name is returned, this allows incorrect data to be visible and ensure all the staff returned meet the building criteria

Code tested:

SELECT STAFF.NAME AS STAFF\_NAME, STAFF.WORKPHONENO AS WORK\_PHONE, STAFF.HOMEPHONENO AS HOME\_PHONE, Building.buildingname AS BUILDING

FROM STAFF

INNER JOIN LOCATION

ON STAFF.LOCATIONID = LOCATION.LOCATIONID

INNER JOIN BUILDING

ON LOCATION.BUILDINGID = BUILDING.BUILDINGID

WHERE BUILDING.BUILDINGNAME = 12345;

Result:

An error is returned as expected; “Invalid number”.

Query 2 Testing:

Code tested:

SELECT EQUIPMENT.NAME AS EQUIPMENT\_NAME, EQUIPMENTALLOCATION.QUANTITY, LOCATION.ROOMNO AS ROOM\_NUMBER, LOCATION.FLOOR, LOCATION.TYPE AS ROOM\_TYPE, BUILDING.BUILDINGNAME AS BUILDING

FROM EQUIPMENTALLOCATION

INNER JOIN EQUIPMENT

ON EQUIPMENTALLOCATION.EQUIPMENTID = EQUIPMENT.EQUIPMENTID

INNER JOIN LOCATION

ON EQUIPMENTALLOCATION.LOCATIONID = LOCATION.LOCATIONID

INNER JOIN BUILDING

ON LOCATION.BUILDINGID = BUILDING.BUILDINGID

WHERE BUILDING.BUILDINGNAME = :User\_Input\_BuildingName

AND

LOCATION.ROOMNO IS NOT NULL

AND

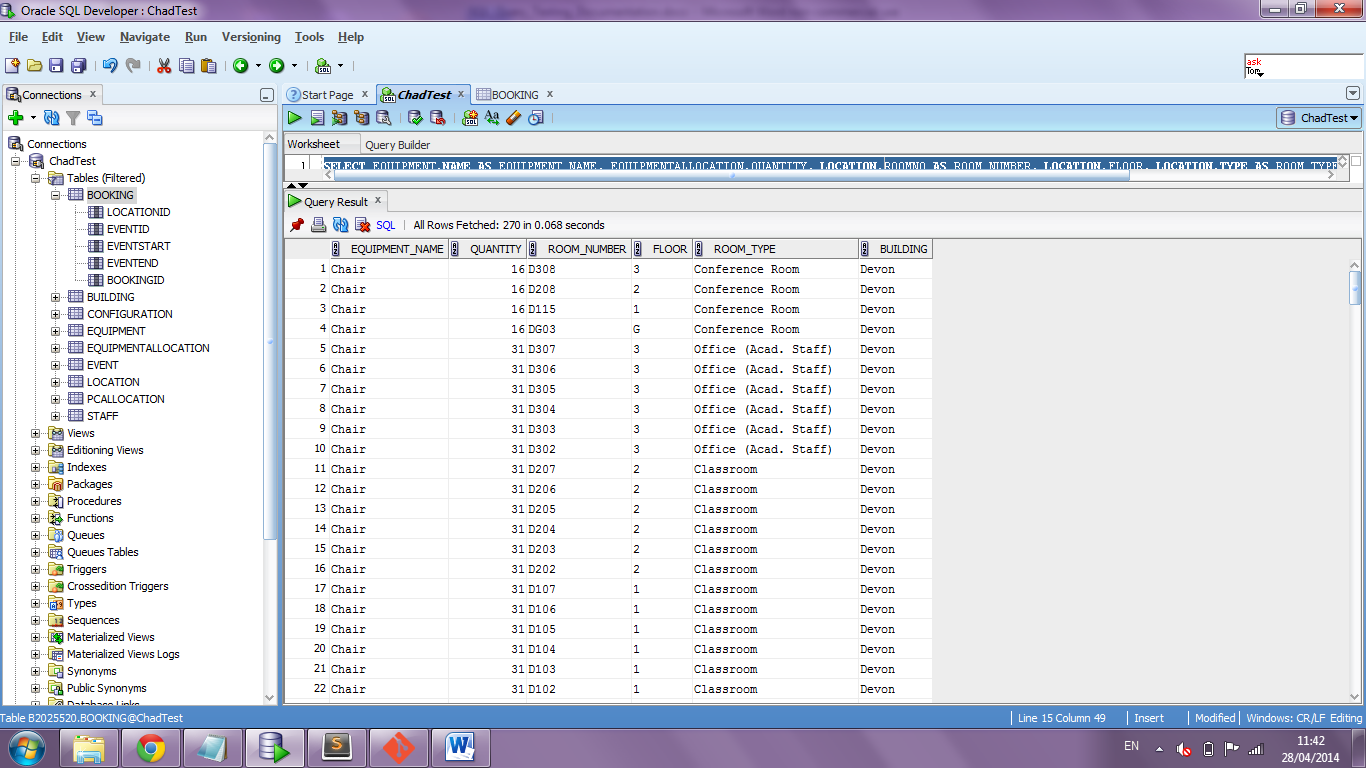
LOCATION.CURRENTCONFIG IS NOT NULL

AND

EQUIPMENTALLOCATION.LOCATIONID IS NOT NULL;

:User\_Input\_BuildingName = ‘Devon’

Result:

The data returned is correct and as expected. The buildingname column shows that the data is related to the correct building.

Query 3 Testing:

Code tested:

SELECT LOCATION.LOCATIONID, LOCATION.ROOMNO, LOCATION.TYPE, LOCATION.FLOOR, BUILDING.BUILDINGNAME, LOCATION.CURRENTCONFIG,

CONFIGURATION.ISLANDCAPACITY, CONFIGURATION.HORSESHOECAPACITY, CONFIGURATION.ROWCAPACITY

FROM LOCATION

INNER JOIN CONFIGURATION

ON LOCATION.LOCATIONID = CONFIGURATION.LOCATIONID

INNER JOIN BUILDING

ON LOCATION.BUILDINGID = BUILDING.BUILDINGID

WHERE LOCATION.LOCATIONID NOT IN

(

SELECT DISTINCT LOCATIONID

FROM BOOKING

WHERE

(

(TO\_DATE(:User\_Input\_Event\_Start, 'dd-mm-rrrr hh24:mi') BETWEEN EVENTSTART AND EVENTEND)

OR

(TO\_DATE(:User\_Input\_Event\_End, 'dd-mm-rrrr hh24:mi') BETWEEN EVENTSTART AND EVENTEND)

OR

(TO\_DATE(:User\_Input\_Event\_Start, 'dd-mm-rrrr hh24:mi') <= EVENTSTART AND TO\_DATE(:User\_Input\_Event\_End, 'dd-mm-rrrr hh24:mi') >= EVENTEND)

)

)

AND

BUILDING.BUILDINGNAME = :User\_Input\_BuildingName

AND

(

CONFIGURATION.ROWCAPACITY >= :User\_Input\_Room\_Capacity

OR

CONFIGURATION.ISLANDCAPACITY >= :User\_Input\_Room\_Capacity

OR

CONFIGURATION.HORSESHOECAPACITY >= :User\_Input\_Room\_Capacity

)

AND

LOCATION.ROOMNO IS NOT NULL

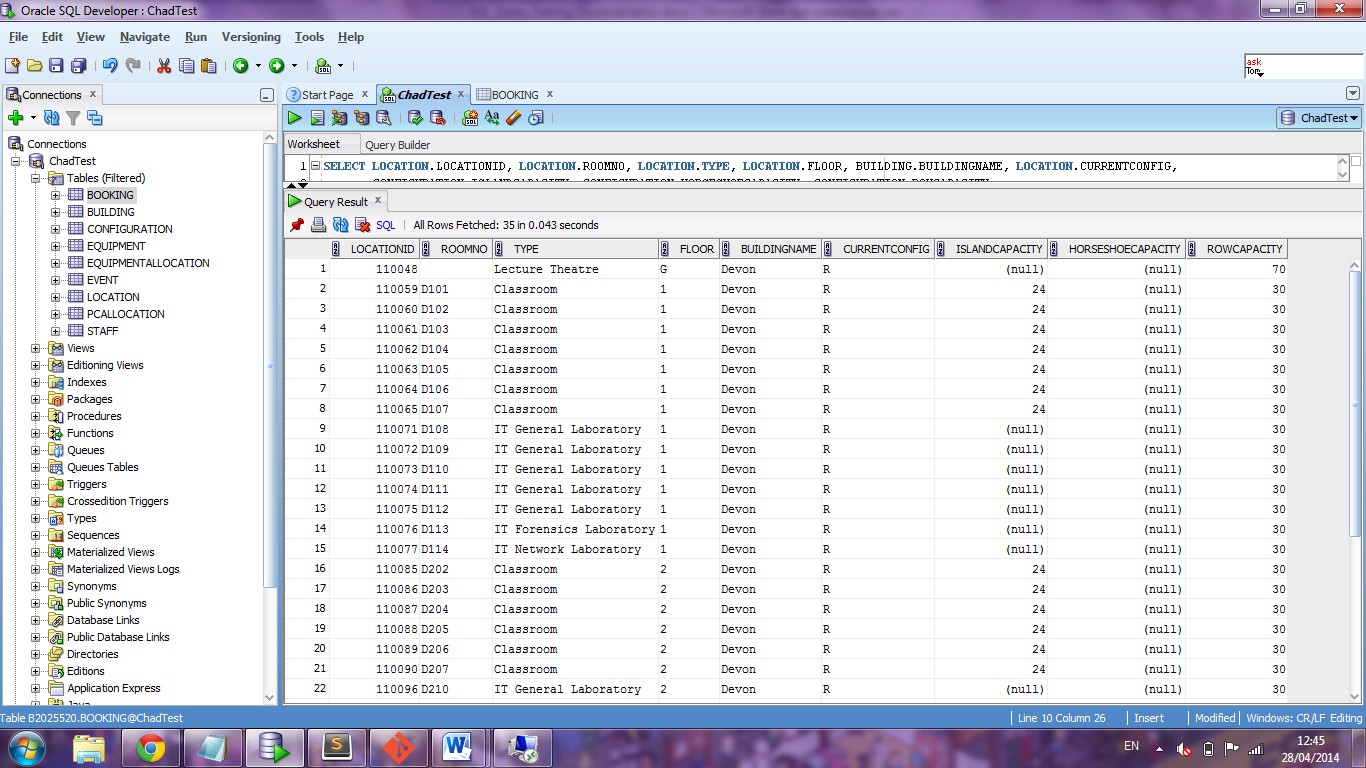
ORDER BY LOCATION.LOCATIONID

:User\_Input\_BuildingName = ‘Devon’

:User\_Input\_Room\_Capacity = 30

:User\_Input\_Event\_End = 04/10/2013 13:00

:User\_Input\_Event\_Start = 04/10/2013 12:00

Result:

The rows returned are correct fitting the critera of the requested query.

Query 4:

Code tested:

INSERT INTO EVENT (EVENTNAME, EVENTPURPOSE, STAFFID)

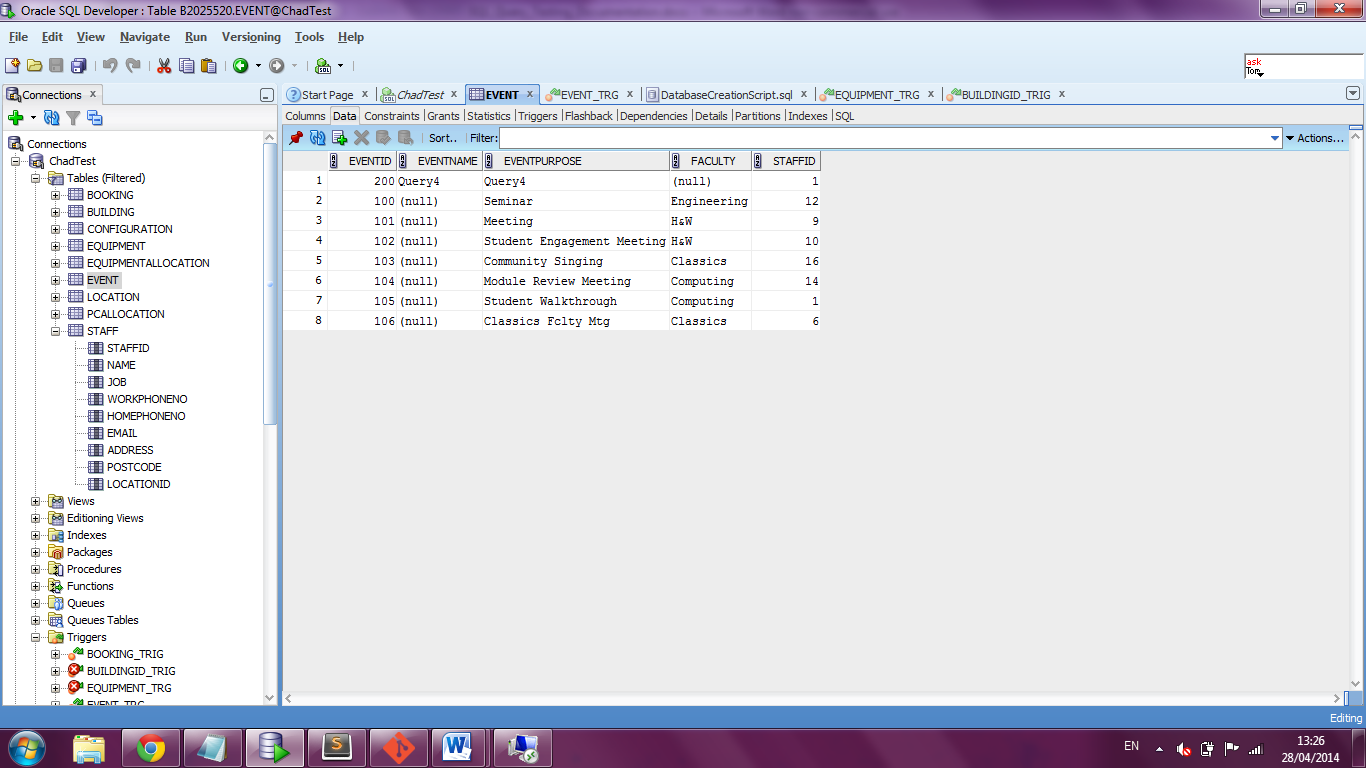
VALUES (:User\_Input\_EventName, :User\_Input\_EventPurpose, (SELECT STAFFID FROM STAFF WHERE STAFF.NAME = :User\_Input\_Organiser));

:User\_Input\_EventName = ‘Query4’

:User\_Input\_EventPurpose = ‘Query4’

:User\_Input\_Organiser = ‘Dave Whitburn’

Result:



The row has been correctly inserted and the staff id is correctly selected from the staff table.

Query 5:

Code tested:

INSERT INTO BOOKING (LOCATIONID, EVENTID, EVENTSTART, EVENTEND)

VALUES ((SELECT LOCATIONID

FROM LOCATION

WHERE ROOMNO = :User\_Input\_RoomNo), (SELECT EVENTID

FROM EVENT

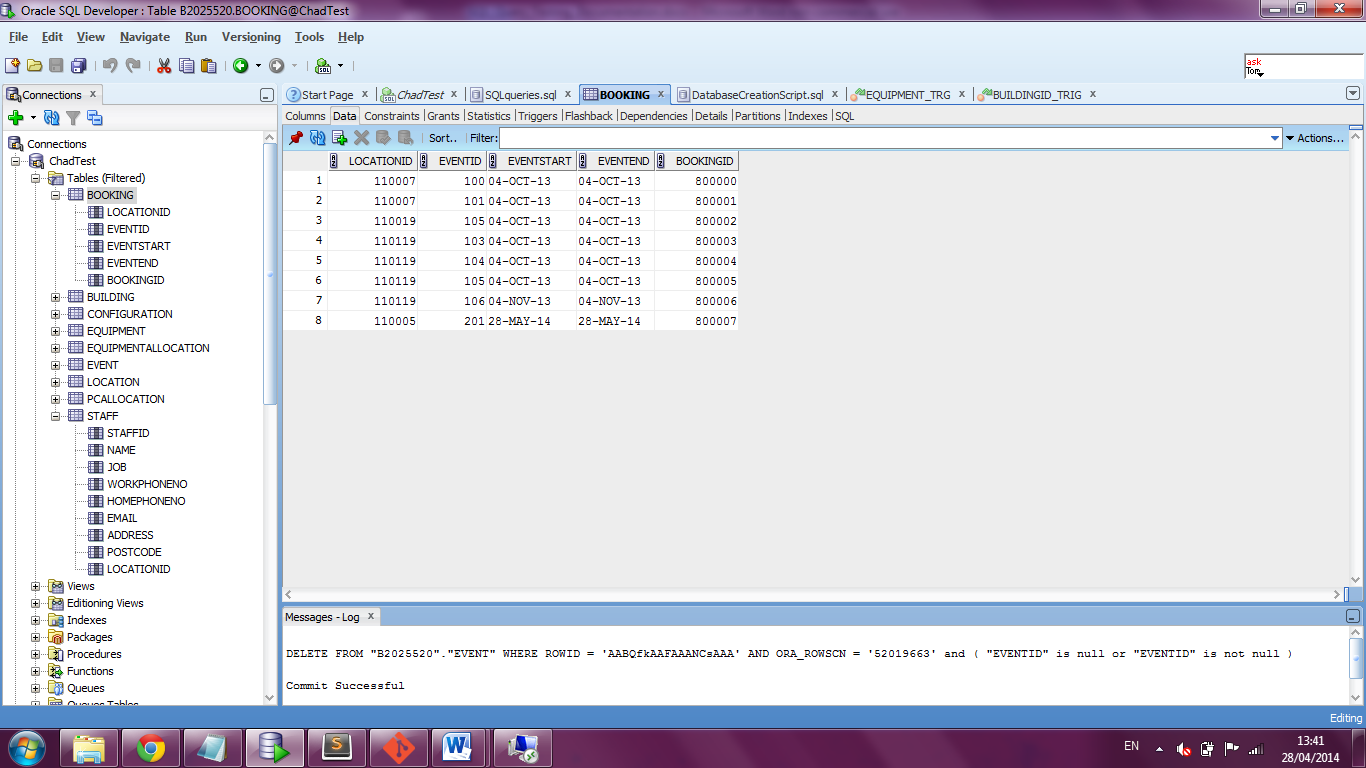
WHERE EVENTNAME = :User\_Input\_EventName), TO\_DATE(:User\_Input\_EventStart, 'dd-mm-rrrr hh24:mi'), TO\_DATE(:User\_Input\_EventEnd, 'dd-mm-rrrr hh24:mi'));

:User\_Input\_EventName = ‘Query4’

:User\_Input\_EventStart = ‘28/05/2014 12:00’

:User\_Input\_EventEnd = ‘28/05/2014 13:00’

Result:



As seen on row 8 the data has been correctly inserted. The selected foreign keys from the Location and Event tables are correct.