

1. Present a report listing the Manager's name and telephone number for each hall of residence.

```
select Hall_Manager, Phone_Number
from Residence_Hall;
```

2. Present a report listing the names and student ids of students with the details of their lease agreements.

```
select s.first_name || ' ' || s.last_name "Student Name", s.student_number "Student Number",
l.lease_number "Lease Number", l.duration "Duration", l.rental_address "Rental Address",
l.move_out_date "Move Out Date", l.move_in_date "Move In Date", l.room_number "Room
Number", l.phone_number "Phone Number"
from student s, lease l, rents r
where s.student_number = r.student_number AND r.lease_number = l.lease_number;
```

3. Display the details of lease agreements that include the summer quarter.

```
select lease_number "Lease Number", duration "Duration", rental_address "Rental Address",
move_out_date "Move Out Date", move_in_date "Move In Date", room_number "Room
Number", phone_number "Phone Number"
from lease
where duration = '4';
```

4. Display the details of the total rent paid by a given student.

```
select s.first_name, s.last_name, sum(r.rent)
from student s, invoice i, rents rt, room r
where s.student_number = i.student_number AND i.payment_received_date < sysdate
      AND s.student_number = rt.student_number
      AND rt.room_number = r.room_number
group by s.first_name, s.last_name;
```

5. Present a report on students who have not paid their invoices by a given date.

```
select s.first_name || ' ' || s.last_name "Student Name", s.student_number
from student s, invoice i
where s.student_number = i.student_number AND i.payment_received_date > '01-FEB-13';
```

6. Display the details of apartment inspections where the property was found to be in an unsatisfactory condition.

```
select inspection_date, satisfactory, comments, apartment_number
from inspection_report
where satisfactory = 'Unsatisfactory';
```

7. Present a report of the names and student ids of students with their room number and place number in a particular hall of residence.

```
select s.first_name || ' ' || s.last_name "Student Name", s.student_number, r.room_number,
r.place_number
from student s, rents r
where s.student_number = r.student_number AND s.residence_type = 'Residence Hall';
```

8. Present a report listing the details of all students currently on the waiting list for accommodation; that is, who were not placed.

```
select first_name || ' ' || last_name "Student Name", student_number
from student
where current_status = 'Waiting';
```

9. Display the total number of students in each student category.

```
select class, count(*)
from student
group by class;
```

10. Present a report of the names and ids for all students who have not supplied details of their next-of-kin.

```
select s.first_name || ' ' || s.last_name "Student Name", s.student_number
from student s
where not exists (select *
                  from nok_relationship r
                  where s.student_number = r.student_number);
```

11. Display the name and internal telephone number of the Adviser for a particular student.

```

select m.first_name || ' ' || m.last_name "Advisor Name", a.phone_number
from student s, staff_member m, advisor a
where s.student_number = '600001' AND m.staff_number = a.advisor_number AND
s.advisor_number = a.advisor_number;

```

12.Display the minimum, maximum, and average monthly rent for rooms in residence halls.

```

select ro.min(rent), ro.max(rent), ro.avg(rent)
from room ro,rented_residence rr, residence_hall rh
where ro.residence_number = rr.residence_number AND rr.residence_number =
rh.residence_number;

```

13.Display the staff number, name, age, and current location of all members of the residence staff who are over 60 years of age today.

```

select staff_number, first_name,last_name, to_number(to_char(sysdate,'YYYY'))-
to_number(to_char(19) || to_char(date_of_birth,'YY')) Age, location
from staff_member
where to_number(to_char(sysdate,'YYYY'))-to_number(to_char(19) ||
to_char(date_of_birth,'YY')) >= 60;

```

Complex Queries:

CQ1:

A student requests to move into an apartment with his friend with the last name Vaughn, do a query seeing if that apartment has available rooms.

```

select sa.available_rooms
from student_apartment sa, room rm, rents rts, student s
where sa.residence_number = rm.residence_number
and rm.room_number = rts.room_number
and rts.student_number = s.student_number
and s.student_number = (select student_number
                        from student
                        where last_name = 'Vaughn');

```

CQ2: Get a list of all apartments and residence halls and their rent-per-room, listing apartments by apartment number and residence halls by name

to get rent for each apartment:

```
select apartment_number "Apt #", rent "Rent"
from student_apartment sa join rented_residence rr on sa.residence_number =
rr.residence_number, room r
where rr.residence_number = r.residence_number
group by apartment_number, rent
order by apartment_number asc
```

to get rent for each residence hall:

```
select rh_name "Name", rent "Rent"
from residence_hall rh join rented_residence rr on rh.residence_number = rr.residence_number,
room r
where rr.residence_number = r.residence_number
group by rh_name, rent
order by rh_name asc
```

I dont know how to get these together on one table!! - Danny

Do a query that will give you all the NOK in a certain zip code whose student is currently not enrolled in any courses but that lives in a residence for the last month.

Display the student's name, student number, next of kin name, relationship and phone number for all renting students who are under 21 and have not paid their rent yet.

invoice not paid:

```
select s.first_name || ' ' || s.last_name "Student Name", s.student_number
from student s, invoice i
where s.student_number = i.student_number AND i.payment_received_date > '01-FEB-13';
```

students under 21:

```
select s.last_name
from student s
where to_number(to_char(sysdate, 'YYYY')) - to_number(to_char(19) ||
to_char(date_of_birth, 'YY')) < 21
```

```
select s.first_name || ' ' || s.last_name "Student", s.student_number "SID", n.nok_name "Next of
Kin", n.nok_phone_number "Phone #", nr.nok_relationship
from next_of_kin n join nok_relationship nr on n.nok_name = nr.nok_name, lease l, rents rts,
student s, invoice i
where nr.student_number = s.student_number
```

```
and s.student_number = i.student_number  
and
```