```
/* Select Statements */
/* Query 1
NAME OF QUERY: Amount a Student Paid
SHOW student_id, amount_paid, date_paid, email
FOR [user-chosen student and year]
QUERY WILL USE: JOIN: yes // SUBQUERY: no // AGGREGATION: yes */

/* we hard coded the year and the student id */

SELECT o.student_id, SUM(t.amount_paid) as amount_paid, YEAR(t.date_paid) AS
date_paid, o.email FROM occupant as o
JOIN transaction as t on t.studID = o.student_id
WHERE YEAR(t.date_paid) = 2020 AND student_id = 76;

/* Query 2:
NAME OF QUERY: Rooms in Building by Price
SHOW b.neighborhood, r.type, r.price
FOR [user-chosen neighborhood]
QUERY WILL USE: JOIN: yes // SUBQUERY: no // AGGREGATION: no */

/* we hard coded the neighborhood */

SELECT b.neighborhood, r.type, r.price FROM building as b
JOIN room as r on r.room_building = b.building_id
WHERE b.neighborhood = "Northwest"
ORDER BY r.type DESC, r.price DESC;

/* Query 3
NAME OF QUERY: Work Where You Live
SHOW building_name, address, and how many employees for a specific building
FOR [user-chosen building]
QUERY WILL USE: JOIN: yes //  SUBQUERY: no // AGGREGATION: yes */

/* we hard coded the building id */

SELECT b.building_name, b.address, COUNT(*) FROM building as b
JOIN occupant as o on o.occupant_building = b.building_id
JOIN employees as e on e.employee_building = b.building_id
WHERE o.occupant_building = 12 AND e.employee_building = 12;

/* Query 4
NAME OF QUERY: March Birthdays
SHOW occupant-name, birthday
FOR [user-chosen month]
QUERY WILL USE: JOIN: yes // SUBQUERY: no // AGGREGATION: no */

/* we hard coded the month */

SELECT CONCAT(o.fname, ' ', o.lname) AS name, o.dob FROM occupant AS o
JOIN building AS b on b.building_id = o.occupant_building
WHERE MONTH(o.dob) = 3;

/* Query 5
NAME OF QUERY: Incomplete Payments
SHOW occupant-name and amount-due
FOR [user-chosen 'late' or 'paid']
QUERY WILL USE: JOIN: yes // SUBQUERY: yes // AGGREGATION: yes */

/* we hard coded the 'late' (1) */
```

```sql
SELECT CONCAT(o.fname, ' ', o.lname) AS name, p.amount_due AS payment FROM occupant
AS o
JOIN transaction AS t ON t.studID = o.student_id
JOIN payment AS p ON p.stud_id = o.student_id
WHERE t.transID NOT IN
      (SELECT t.transID FROM transaction AS t
       WHERE SIGN(p.due_date - t.date_paid) = 1);

/* Query 6: Profitability
Display the most profitable or least profitable buildings (allow the user to select
whether most or least)
in a given year based per occupant based on actual income (not cost). */

/* we hard coded the year and max */

SELECT b.building_name, SUM(t.amount_paid) AS sum FROM building AS b
JOIN occupant AS o ON b.building_id = o.occupant_building
JOIN transaction AS t ON o.student_id = t.studID
WHERE YEAR(t.date_paid) = 2021
GROUP BY b.building_name
ORDER BY sum ASC
LIMIT 1;
```