James Satherley, Database Design, Assignment 2, 158.247

3)

```
ALTER TABLE appointments
ADD CHECK (apt_start >= '8:00' AND apt_end <= '17:00' AND apt_start + '00:15' <= apt_end);
```

4)

```
CREATE OR REPLACE FUNCTION
apt_count(

IN date_input date, IN time_input
time)

RETURNS INTEGER AS $$

BEGIN

   RETURN(

   SELECT COUNT(*)

   FROM appointments

   WHERE apt_start <= time_input

   AND   apt_end > time_input

   AND   date_input = apt_date);

END;

$$ LANGUAGE plpgsql;


SELECT apt_count('2021-05-01',
'12:00:00');

SELECT * FROM apt_count('2021-05-
01', '11:00')
```

5)

```
CREATE OR REPLACE VIEW fully_booked AS
        SELECT a1.apt_date, a1.apt_start, a1.apt_end
        FROM appointments a1
        WHERE apt_count(a1.apt_date, a1.apt_start) =
avl_count(a1.apt_date, a1.apt_start)
        AND a1.apt_end <=
                (SELECT MIN(a2.apt_end)
                FROM appointments a2
                WHERE a1.apt_start = a2.apt_start)
        GROUP BY a1.apt_date, a1.apt_start,
a1.apt_end;


SELECT *
FROM fully_booked;
```

```
CREATE OR REPLACE FUNCTION avl_count(
IN date_input date, IN time_input time)
RETURNS INTEGER AS $$
BEGIN
  RETURN(
  SELECT COUNT(*)
  FROM availability
  WHERE avl_start <= time_input
  AND   avl_end > time_input
  AND   date_input = avl_date);
END;
$$ LANGUAGE plpgsql;


SELECT avl_count('2021-05-01',
'12:00:00');
```

6)

```
CREATE OR REPLACE FUNCTION dates_between(
IN low date, IN high date)
RETURNS TABLE (
        booked_date date
) LANGUAGE plpgsql AS $$
BEGIN
  RETURN QUERY
                SELECT apt_date
                FROM fully_booked
                WHERE apt_date BETWEEN low AND high;
END;$$;
SELECT dates_between('2021-05-01', '2021-05-02');
```

```
CREATE OR REPLACE FUNCTION insert_check(
IN username varchar, doctor varchar, date_in date, start_time time, IN end_time
time)
RETURNS VOID
LANGUAGE plpgsql AS $$
BEGIN
        IF avl_count(date_in, start_time) > apt_count(date_in, start_time) AND
doctor = check_doctor(doctor, date_in) THEN
        INSERT INTO appointments VALUES(username, doctor, date_in, start_time,
end_time);
        END IF;
END;$$;
```

```
CREATE OR REPLACE FUNCTION check_doctor(
IN doctor_in varchar, date_in date)
RETURNS varchar
LANGUAGE plpgsql AS $$
BEGIN
        RETURN(
                SELECT doctor
                FROM availability
                WHERE doctor = doctor_in
                AND avl_date = date_in);
END;$$;
```