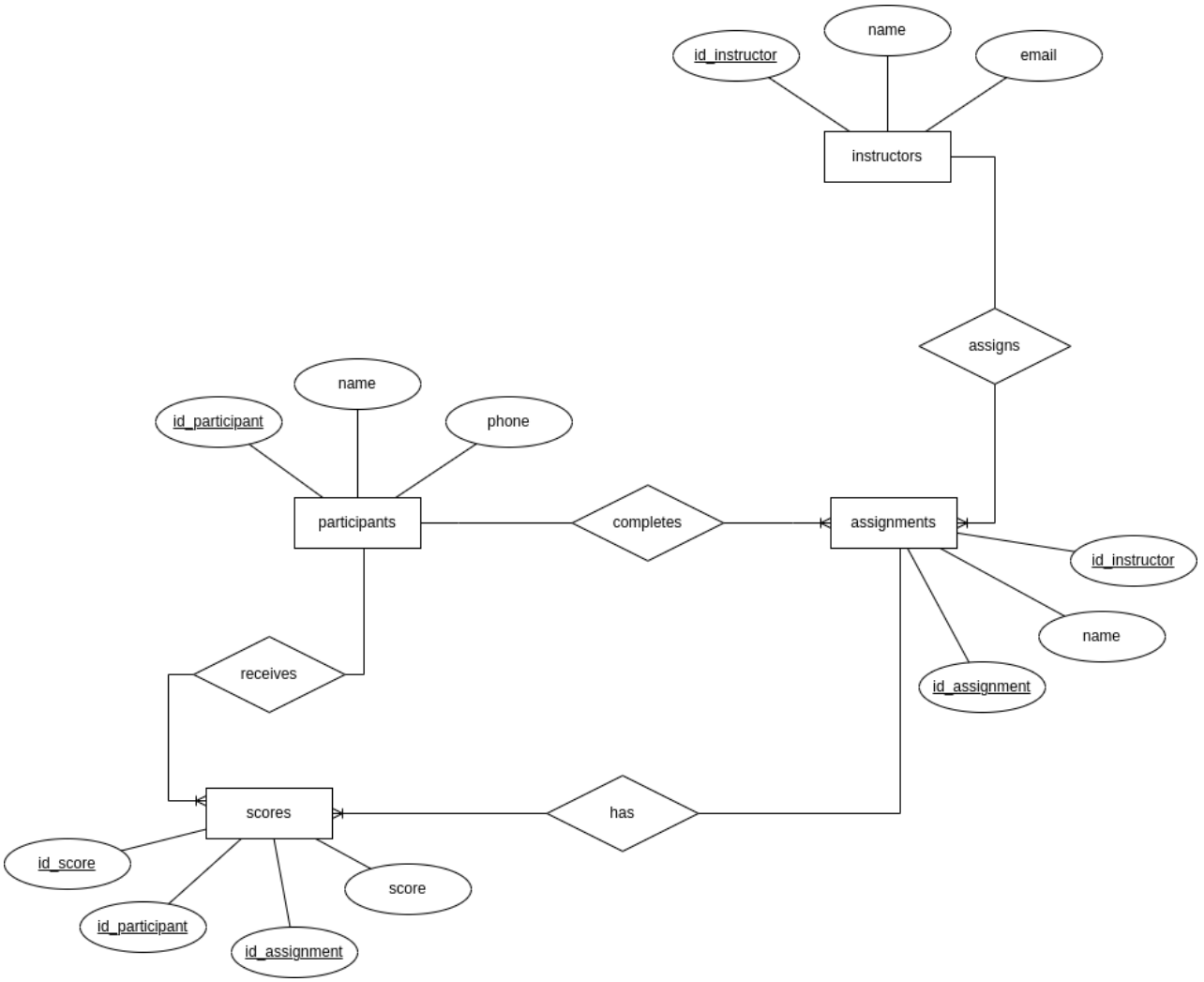
1.



2.

```
CREATE TABLE participants (
  id_participant SERIAL PRIMARY KEY,
  name VARCHAR(55),
  phone VARCHAR(13)
);

CREATE TABLE assignments (
  id_assignment SERIAL PRIMARY KEY,
  name VARCHAR(255),
  id_instructor INT,
);

CREATE TABLE scores (
    id_score SERIAL PRIMARY KEY,
    id_participant INT,
    id_assignment INT,
    score DECIMAL(5, 2),
```

```
    created_at DATE
    FOREIGN KEY (id_participant) REFERENCES participant(id_participant),
    FOREIGN KEY (id_assignment) REFERENCES assignment(id_assignment)
);

CREATE TABLE instructor (
    id_instructor SERIAL PRIMARY KEY,
    name VARCHAR(30),
    email VARCHAR(20)
);
```

3.

a.
```
SELECT participants.name AS participant_name,
assignments.name AS assignment_name,
scores.score
FROM scores
JOIN participants ON scores.id_participant = participants.id_participant
JOIN assignments ON scores.id_assignment = assignments.id_assignment
WHERE participants.id_participant = 1;
```

b.
```
SELECT participants.name AS participant_name,
AVG(scores.score) AS average_score
FROM participants
LEFT JOIN scores ON participants.id_participant = scores.id_participant
GROUP BY participants.name;
```

c.
```
SELECT
  CASE
    WHEN score >= 90 THEN 'A'
    WHEN score >= 80 THEN 'B'
    WHEN score >= 70 THEN 'C'
    WHEN score >= 60 THEN 'D'
    ELSE 'F'
  END AS nilai_range,
  COUNT(*) AS jumlah_peserta
FROM scores
GROUP BY nilai_range
ORDER BY nilai_range;
```

d.
```
SELECT
  nilai_range,
  string_agg(participants.name, ', ') AS peserta_dalam_rentang
FROM (
  SELECT
    participants.name,
    CASE
```

```
      WHEN scores.score >= 90 THEN 'A'
      WHEN scores.score >= 80 THEN 'B'
      WHEN scores.score >= 70 THEN 'C'
      WHEN scores.score >= 60 THEN 'D'
      ELSE 'F'
    END AS nilai_range
  FROM participants
  LEFT JOIN scores ON participants.id_participant = scores.id_participant
) AS nilai_peserta
GROUP BY nilai_range
ORDER BY nilai_range;
```

e.
```
SELECT
  tanggal_penilaian,
  COUNT(*) AS jumlah_penilaian
FROM (
  SELECT
    created_at::DATE AS tanggal_penilaian
  FROM scores
      WHERE created_at BETWEEN '2023-08-30' AND '2023-09-01') AS
tanggal_penilaian_count
GROUP BY tanggal_penilaian
ORDER BY tanggal_penilaian;
```