

1	2	3	Σ
/10	/5	/5	/20

Korrigiert am: _____

Aufgabe 5.1 (Punkte: /10)

(a)

```
SELECT name
FROM medikament
WHERE preis > 20
```

(b)

```
SELECT name, arzt_id
FROM arzt A
WHERE NOT EXISTS
(SELECT arzt
FROM behandlung
WHERE arzt = A.arzt_id)
```

(c)

```
SELECT name, preis
FROM medikament
ORDER BY preis ASC
```

(d)

```
CREATE VIEW Behandelte_Kunden AS
SELECT name
FROM patient P
WHERE EXISTS
(SELECT patient
FROM behandlung
WHERE patient = P.patient_id)
```

```
SELECT * FROM Behandelte_Kunden
```

(e)

```
SELECT patient_id
FROM patient P
INNER JOIN behandlung ON P.patient_id = behandlung.patient
GROUP BY patient_id
HAVING COUNT(*) >= ALL
(SELECT COUNT(patient_id)
```

```
FROM patient P
INNER JOIN behandlung ON P.patient_id = behandlung.patient
GROUP BY patient_id)
```

(f)

```
SELECT name
FROM medikament M
WHERE EXISTS
(SELECT medikament
FROM behandlung
WHERE medikament = M.medikament_id
GROUP BY medikament
HAVING COUNT(*) > 3)
```

(g)

```
SELECT name, COUNT(DISTINCT medikament) AS anzahlVerschiedenerMedikamente
FROM arzt A JOIN behandlung B
ON A.arzt_id = B.arzt
GROUP BY arzt
ORDER BY name ASC
```

Aufgabe 5.2 (Punkte: /5)

Die Anfrage gibt Namen und durchschnittliche Anzahl Sterne der Filme aus. Die durchschnittliche Anzahl der Sterne eines Films errechnet sich aus seinen Rezensionen, wobei nur die Rezensionen der Kritiker, die mindestens 200 Rezensionen verfasst und dabei höchstens 5% der Filme mit der besten Bewertung von 5 Sternen ausgezeichnet haben, berücksichtigt werden. Gibt es für einen Film keine Rezensionen, die den genannten Kriterien entsprechen, wird er nicht mit ausgegeben.

Die Aufgabenstellung würde so lauten:

„Erstellen Sie einen Query, der Namen und durchschnittliche Anzahl der Sterne aller Filme ausgibt. Es sollen bei der Ermittlung der durchschnittlichen Anzahl Sterne nur Rezensionen berücksichtigt werden, die von einem Kritiker stammen, der in höchstens 5% seiner Kritiken mit der höchstmöglichen Anzahl Sterne vergeben hat, und der schon mindestens 200 Kritiken abgegeben hat. Gibt es für einen Film keine solchen Rezensionen, soll er nicht mit ausgegeben werden.“

Subquery *m*: Gibt eine Relation aus, die für jeden Kritiker, der mindestens 200 Kritiken geschrieben hat, einmal seine ID und die genaue Anzahl seiner Kritiken enthält.

Subquery *n*: Gibt eine Relation aus, die für jeden Kritiker, der mindestens eine Kritik geschrieben hat, einmal seine ID und die Anzahl der Kritiken, in denen er 5 Sterne vergeben hat, enthält.

Subquery *l*: Gibt eine Relation aus, die für jeden Kritiker, der mindestens 200 Kritiken geschrieben

hat, und dabei in höchstens 5% der Rezensionen 5 Sterne vergeben hat, seine ID, die Anzahl seiner Rezensionen und die Anzahl seiner Rezensionen, in denen er 5 Sterne vergeben hat, enthält.

Aufgabe 5.3 (Punkte: /5)

```
SELECT DISTINCT name, title, location, filename, pictures.timestamp, COUNT(commentID)
AS commentcount
FROM (pictures NATURAL JOIN users)
LEFT JOIN comments ON pictures.pictureID=comments.pictureID
WHERE current_timestamp - '1 day'::interval <= pictures.timestamp AND current_timestamp
>= pictures.timestamp
GROUP BY name, title, location, filename, pictures.timestamp
ORDER BY pictures.timestamp DESC
```