

Examen BD - SQL

-- Subject nr. 145

-- Tudor Raluca, Grupa 135

--1

```
SELECT u.ume, u.prenume,
       c.denumire AS curs,
       md.denumire AS desfasurare,
       p.oras
FROM curs c JOIN sesiune_curs sc ON (c.id = sc.id_curs)
  LEFT JOIN mod_desfasurare md ON (md.id = sc.id_mod_desfasurare)
  JOIN se_inscrie si ON (sc.id = si.id_sesiune_curs)
  JOIN utilizator u ON (u.id = si.id_utilizator)
  JOIN profil p ON (p.id_utilizator = u.id)
WHERE (si.data_inscriere + c.durata_zile) >= SYSDATE;
```

--2

```
SELECT u.email, NVL(p.cale_poza, 'Fara poza'),
-- durata totala a cursurilor pe care utilizatorul
-- le-a urmat in modul de desfasurare respectiv
SUM(CASE
    WHEN md.denumire='online' THEN c.durata_zile
    ELSE 0
    END) "Durata totala online",
SUM(CASE
    WHEN md.denumire='sediu client' THEN c.durata_zile
    ELSE 0
    END) "Durata totala sediu client",
SUM(CASE
    WHEN md.denumire='sediu propriu' THEN c.durata_zile
    ELSE 0
    END) "Durata totala sediu propriu"
FROM utilizator u
  JOIN profil p ON (p.id_utilizator = u.id)
  JOIN se_inscrie si ON (u.id = si.id_utilizator)
  JOIN sesiune_curs sc ON (si.id_sesiune_curs = sc.id)
  JOIN mod_desfasurare md ON (md.id = sc.id_mod_desfasurare)
  JOIN curs c ON (c.id = sc.id_curs)
GROUP BY u.id, u.email, p.cale_poza;
```

--3

-- Rezolv simuland DIVISION cu COUNT

```
SELECT c0.denumire, sc0.cost
FROM curs c0 JOIN sesiune_curs sc0 ON (c0.id = sc0.id_curs)
WHERE sc0.id_curs IN (SELECT sc.id_curs
                     FROM curs c JOIN sesiune_curs sc ON (c.id = sc.id_curs)
                     WHERE sc.data_start + c.durata_zile >= SYSDATE
                     )
```

```
GROUP BY c0.denumire, sc0.cost
HAVING COUNT(sc0.id_curs) = (SELECT COUNT(sc.id_curs)
                             FROM curs c JOIN sesiune_curs sc ON (c.id = sc.id_curs)
                             WHERE sc.data_start + c.durata_zile >= SYSDATE
                             );
-- folosesc COUNT(id) si nu COUNT(*) pentru a nu numara si null-urile

--4
-- adaug in sesiune_curs
INSERT INTO sesiune_curs(id, data_start, id_curs, cost)
VALUES (
    NVL((SELECT MAX(id) + 1 FROM sesiune_curs), 0),
    SYSDATE + 2,
    -- cursul care nu are o sesiune in desfasurare la data curenta
    (SELECT c0.id -- aflu id ul cursului
     FROM curs c0
     WHERE NOT EXISTS (
         SELECT c.id
         FROM curs c JOIN sesiune_curs sc ON (c.id = sc.id_curs)
         WHERE sc.data_start + c.durata_zile >= SYSDATE
         AND c0.id = c.id
     )
    ),
    (SELECT min(cost)
     FROM sesiune_curs sc
     WHERE id_curs = (SELECT c0.id
                     FROM curs c0
                     WHERE NOT EXISTS (
                         SELECT c.id
                         FROM curs c JOIN sesiune_curs sc ON (c.id = sc.id_curs)
                         WHERE sc.data_start + c.durata_zile >= SYSDATE
                         AND c0.id = c.id
                     )
                    )
    )
);

--5
-- Aflu denumirea astfel:
SELECT constraint_name
FROM user_constraints
WHERE LOWER(table_name) = 'profil' AND constraint_type = 'R';
--Denumirea este FK_PROFIL_UTILIZATOR

--Suprim constrangerea:
ALTER TABLE profil
DROP CONSTRAINT FK_PROFIL_UTILIZATOR;

--O readaug cu ON DELETE CASCADE:
ALTER TABLE profil
```

```
ADD CONSTRAINT FK_PROFIL_UTILIZATOR FOREIGN KEY (id_utilizator) REFERENCES  
utilizator(id) ON DELETE CASCADE;
```

```
--Verific:
```

```
SELECT *
```

```
FROM user_constraints
```

```
WHERE LOWER(table_name) = 'profil';
```

```
-- La delete rule se poate observa acum "CASCADE"
```