

Interogări

Reuniune:

- `SELECT nume FROM localuri WHERE id_local IN (SELECT id_local FROM cafenele) UNION DISTINCT SELECT nume FROM localuri WHERE id_local IN (SELECT id_local FROM cofetarii);`

Diferență:

- `SELECT l.nume FROM localuri l WHERE NOT EXISTS (SELECT NULL FROM localuri_firme_livrare lf JOIN firme_livrare f ON lf.id_firma = f.id_firma WHERE lf.id_local = l.id_local AND f.nume = l.nume);`

Selectie:

- `SELECT nume from localuri where animale_de_companie = 1;`

Proiecție:

- `SELECT DISTINCT l.nume, c.mail FROM localuri l JOIN contacte c ON l.id_local = c.id_local;`

Joncțiune simplă:

- `SELECT l.nume nume_local, f.nume nume_firma, lf.cost_standard cost FROM localuri l JOIN localuri_firme_livrare lf ON l.id_local = lf.id_local JOIN firme_livrare f ON lf.id_firma = f.id_firma ORDER BY l.nume;`

Joncțiune compusă 1:

- `CREATE TEMPORARY TABLE firme_bolt AS SELECT id_firma FROM firme_livrare WHERE nume = 'Bolt Food';`
- `CREATE TEMPORARY TABLE livrari_ieftine AS SELECT id_local, id_firma FROM localuri_firme_livrare WHERE cost_standard < 17;`

- `SELECT l.nume FROM localuri l JOIN restaurante r ON l.id_local = r.id_local JOIN livrari_jeftine lf ON l.id_local = lf.id_local JOIN firme_bolt f ON lf.id_firma = f.id_firma;`

Joncțiune compusă 2:

- `SELECT DISTINCT l.nume FROM (SELECT id_local FROM programe WHERE zi = 'Duminica') pl JOIN localuri l ON l.id_local = pl.id_local JOIN localuri_firme_livrare lf ON l.id_local = lf.id_local JOIN (SELECT id_firma FROM programe WHERE zi = 'Duminica') pf ON lf.id_firma = pf.id_firma;`

Mai multe detalii referitoare la logica operațiilor se pot găsi în documentația de la pagina 14 la pagina 19.