

Modelo Relacional

ivm(serial_number, manuf)

point_of_retail(address, name)

retailer(tin, name)

- unique(name)

category(name)

- IC-1: name must exist in super_category and/or simple_category
- IC-2: name cannot exist in super_category and simple_category

simple_category(name)

- name: FK(category.name)

super_category(name)

- name: FK(category.name)
- IC-3: In order to super_category exist, name must participate in the relation has_other

shelve(nr, serial_number, manuf, height, name)

- serial_number, manuf: FK(ivm)
- name: FK(category)
- IC-4: nr, serial_number and manuf must exist in ambient_temp_shelf and/or warm_shelf and/or cold_shelf
- IC-5: nr, serial_number and manuf cannot exist in ambient_temp_shelf and warm_shelf
- IC-6: nr, serial_number and manuf cannot exist in ambient_temp_shelf and cold_shelf
- IC-7: nr, serial_number and manuf cannot exist in warm_shelf and cold_shelf
- IC-8: nr, serial_number and manuf cannot exist in ambient_temp_shelf and warm_shelf and cold_shelf

ambient_temp_shelf(nr, serial_number, manuf)

- name: FK(shelve.nr, shelve.serial_number, shelve.manuf)

warm_shelf(nr, serial_number, manuf)

- name: FK(shelve.nr, shelve.serial_number, shelve.manuf)

cold_shelf(nr, serial_number, manuf)

- name: FK(shelve.nr, shelve.serial_number, shelve.manuf)

product(ean, descr)

- IC-9: every product (ean) must participate in the 'has' association

planogram(nr, serial_number, manuf, ean, faces, units, loc)

- nr, serial_number, manuf: FK(shelve)
- ean: FK(product)

replenishment_event(instant, ean, nr, serial_number, manuf, units, tin)

- ean, nr, serial_number, manuf: FK(planogram)
- tin: FK(retailer)
- IC-10: units <= planogram.units
- IC-11: a product can only be replenished if its category equals shelf.name
- IC-12: a product can only be replenished by a retailer of its category

installed_at(serial_number, manuf, address, nr)

- serial_number, manuf: FK(ivm)
- address: FK(point_of_retail)

responsible_for(name, tin, serial_number, manuf)

- name: FK(category)
- tin: FK(retailer)
- serial_number, manuf: FK(ivm)

has(ean, name)

- ean: FK(product)
- name: FK(category)

has_other(category, super_category)

- category: FK(category)
- super_category: FK(super_category)
- IC-13: There cannot be cycles in categories hierarchies
- IC-14: category != super_category

Álgebra Relacional

1. $\pi_{\text{ean, descr}} (\sigma_{\text{Categoria} = \text{Barras Energéticas} \wedge \text{units} > 10 \wedge \text{instant} > 31/12/2021} (\text{product} \bowtie \text{shelf} \bowtie \text{replenishment_event}))$

2. $\pi_{\text{serial_number}} (\sigma_{\text{ean} = 9001490100070} (\text{planogram} \bowtie \text{shelf}))$

3. $G_{\text{COUNT}} (\sigma_{\text{super_category} = \text{Sopas Take-Away}} (\text{has_other}))$

4. $\text{sums} \leftarrow \text{ean, descr } G_{\text{SUM}(\text{units}) \rightarrow C} (\text{replenishment_event})$

$\pi_{\text{ean, descr}} (G_{\text{MAX}(C)} (\text{sums}) \bowtie \text{sums})$

SQL

1.

```
SELECT ean, descr
FROM product
NATURAL JOIN shelve
NATURAL JOIN replenishment_event
WHERE instante > 31/12/2021 AND units > 10
AND name = 'Barras Energéticas';
```

2.

```
SELECT serial_number
FROM shelve
NATURAL JOIN planogram
WHERE ean = 9002490100070;
```

3.

```
SELECT COUNT(*)
FROM has_other
WHERE super_category = 'Sopas Take-Away';
```

4.

```
SELECT ean, descr
FROM replenishment_event
GROUP BY ean, descr
HAVING SUM(units) >= ALL(
    SELECT SUM(units)
    FROM replenishment_event
    GROUP BY ean, descr);
```