point\_of\_retail(address, name)

ivm(serial\_number, manuf)

retailer(tin, name, instant, serial\_number, manuf, nr, ean)

* serial\_number, manuf, nr, ean: FK(replenishment\_event.serial\_number, replenishment\_event.manuf, replenishment\_event.nr, replenishment\_event.ean)
* UNIQUE(name)

installed\_at(serial\_number, manuf, address, nr)

* serial\_number, manuf: FK(ivm.serial\_number, ivm.manuf)
* address: FK(point\_of\_retail.address)

shelf(serial\_number, manuf, nr, height, name)

* serial\_number, manuf: FK(ivm.serial\_number, ivm.manuf)
* name: FK(category.name)
* RI-1: No shelf can exist at the same time in “Ambient Temp Shelf”, “Warm Shelf” and “Cold Shelf”
* RI-2: Every shelf must exist in “Ambient Temp Shelf”, “Warm Temp Shelf” and/or “Cold Shelf”

product(ean, descr)

* RI-1: Every “Product” (ean) must participate in the “has” association

planogram(ean, serial\_number, manuf, nr, height, faces, units, loc)

* ean: FK(product.ean)
* serial\_number, manuf, nr: FK(shelf.serial\_number, shelf.manuf, shelf.nr)

ambient\_temp\_shelf(serial\_number, manuf, nr)

* serial\_number, manuf, nr: FK(shelf.serial\_number, shelf.manuf, shelf.nr)

warm\_shelf(serial\_number, manuf, nr)

* serial\_number, manuf, nr: FK(shelf.serial\_number, shelf.manuf, shelf.nr)

cold\_shelf(serial\_number, manuf, nr)

* serial\_number, manuf, nr: FK(shelf.serial\_number, shelf.manuf, shelf.nr)

category(name, serial\_number, manuf, nr )

* serial\_number, manuf, nr: FK(shelf.serial\_number, shelf.manuf, shelf.nr)
* RI-1: No category can exist at the same time in “Simple Category” and “Super Category”
* RI-2: Every category must exist in “Simple Category” and/or “Super Category”
* RI-3: No category can contain itself
* RI-4: No cicles are allowed in the “Category” hierarchy

simple\_category(name)

* name: FK(category.name)

super\_category(name)

* name: FK(category.name)
* RI-1: Every “Super Category” (name) must participate in the “has\_other” association

has\_other(name, super\_name)

* name: FK(category.name)
* super\_name: FK(super\_category.name)

has(name, ean)

* name: FK(category.name)
* ean: FK(product.ean)

responsible\_for(tin, name, serial\_number, manuf)

* tin: FK(retailer.tin)
* name: FK(category.name)
* serial\_number, manuf: FK(ivm.serial\_number, ivm.manuf)

replenishment\_event(ean, serial\_number, manuf, nr, instant , tin, units)

* ean, serial\_number, manuf, nr : FK(planogram.ean, planogram.serial\_number, planogram.manuf, planogram.nr)
* tin : FK(Retailer.tin)
* RI-1: “units” cannot exceed planogram.units.

1.

T ← (σunits > '10' ∧ instant > '2021/12/31' (replenishment\_event))

πean,descr (σname='Barras Energéticas' (T ⨝ product ⨝ has ⨝ category))

2.

π serial\_number (planogram ⨝ (σean='9002490100070' (product)))

ou apenas,

π serial\_number (σean='9002490100070' (planogram))

3. preciso extrair count da tabela??

Gcount() (π name ((σ super\_name='Sopas Take-Away' (has\_other)) ÷ (σ name='Sopas Take-Away' (category))))

Ou,

π count (σ super\_name='Sopas Take-Away' (super\_name Gcount() (has\_other)) )

4.

T ← eanGsum(units)->tot\_units (replenishment\_event)

πean,descr(product ⨝ T ⨝ Gmax(tot\_units)(T))

ou,

πean,descr(Gmax(tot\_units)( product ⨝ T))

1. SELECT result.ean, result.descr
2. FROM(
3. SELECT \*
4. FROM replenishment\_event NATURAL JOIN product NATURAL JOIN category
5. WHERE replenishment\_event.units > 10 AND replenishment\_event.instant > ‘2021/12/21’
6. AND product.ean == replenishment\_event.ean
7. AND category.name == ‘Barras Energéticas’) AS result
8. SELECT plan.serial\_number
9. FROM planogram NATURAL JOIN product as plan
10. WHERE product.ean == ‘9002490100070’
11. SELECT COUNT(\*)
12. FROM(
13. SELECT result.name
14. FROM has\_other DIVISION category
15. WHERE has\_other.super\_name == ‘Sopas Take-Away’
16. AND category.name == ‘Sopas Take-Away’)
17. SELECT result.ean, result.descr
18. FROM product NATURAL JOIN
19. HAVING