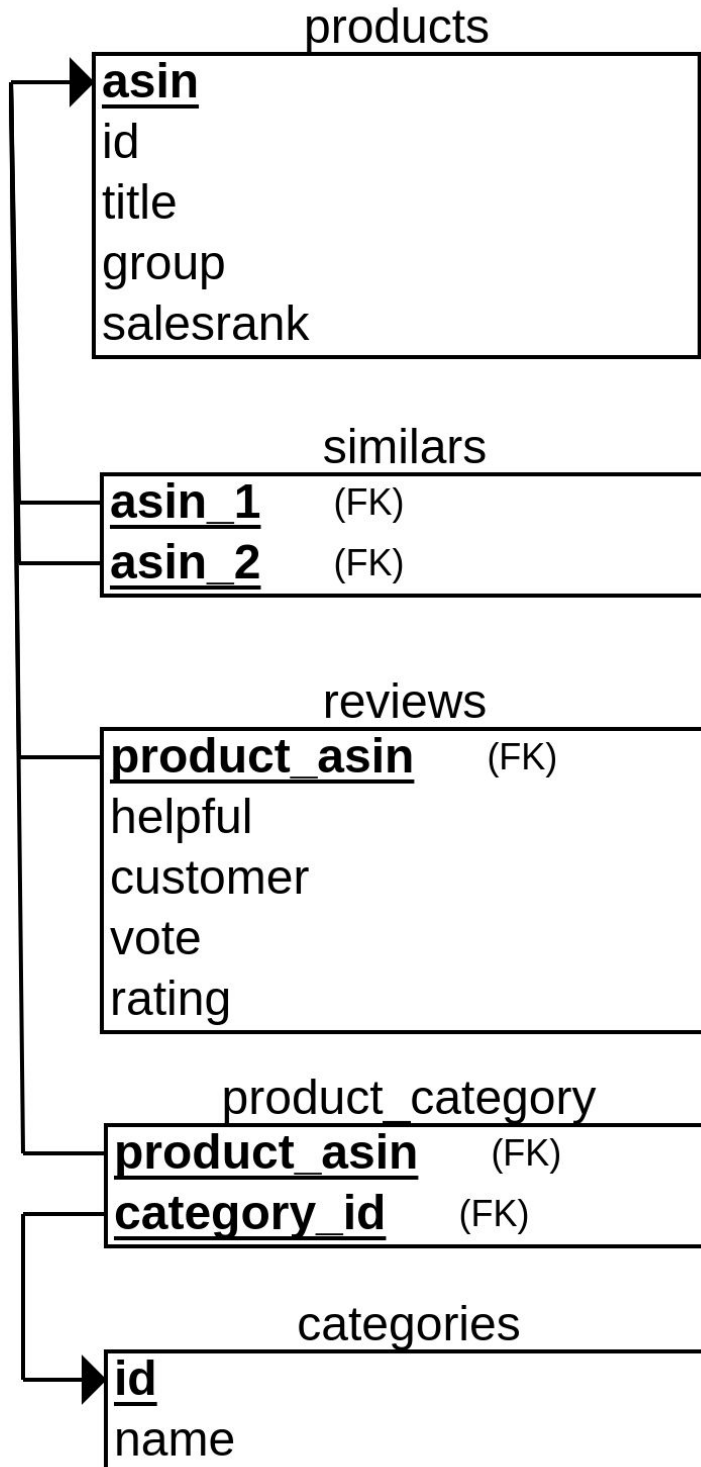


# TP4 - BD

## 1 - Esquema relacional



## 2 - Em SQL:

```
create table if not exists products
```

```

(
  id integer,
  asin varchar not null
    constraint products_pk
      primary key,
  title varchar,
  "group" varchar,
  salesrank integer
);

create unique index if not exists products_asin_uindex
on products (asin);

create table if not exists similars
(
  asin_1 varchar not null
    constraint similars_products_asin_fk
      references products,
  asin_2 varchar
);

create table if not exists categories
(
  id integer not null
    constraint categories_pk
      primary key,
  name varchar
);

create table if not exists product_category
(
  product_asin varchar
    constraint product_category_products_asin_fk
      references products,
  category_id integer
    constraint product_category_categories_id_fk
      references categories
);

```

```

create unique index if not exists
product_category_product_asin_category_id_uindex
on product_category (product_asin, category_id);

create table if not exists reviews
(
  id serial not null
    constraint reviews_pk
      primary key,
  date date,
  customer varchar,
  rating integer,
  votes integer,
  helpful integer,
  product_asin varchar
    constraint reviews_products_asin_fk
      references products
);

create unique index if not exists reviews_id_uindex
on reviews (id);

```

### 3 - Dicionário de dados:

#### 3.1 Mapeamento para as relações:

**products:** Foi criada a partir da entidade products do diagrama, utilizando o passo 1 descrito no algoritmo de mapeamento EER -> relacional. (Apenas a criação da relação a partir de uma entidade regular)

**similars:** Foi criada a partir do relacionamento M:N similars do diagrama, utilizando o passo 5 descrito no algoritmo de mapeamento EER -> relacional. (Criar uma tabela para o relacionamento). Como as **duas** entidades do relacionamento são a mesma (**products**), as duas chaves estrangeiras da relação **similars** são as chaves primárias de **products**.

**reviews:** Foi criada a partir do passo 2 descrito no algoritmo de mapeamento. O passo 2 se refere a como mapear entidades fracas (caso de reviews). Como a entidade review era **fraca**, foi criada uma relação para ela, utilizando a chave primária de **products** (a entidade **dona** de review) como chave estrangeira de **reviews**. Além disso, foram inseridos os atributos de entidade fraca reviews nessa relação.

**product\_category:** Essa relação foi criada a partir do passo 5 descrito no algoritmo de mapeamento (M:N). Foi criada uma relação para o relacionamento **product\_category**. Como as entidades da relação são **products** e **category**, as chaves **estrangeiras** de **product\_category** são as chaves primárias de **products** e **category**.

**categories:** Essa relação foi criada a partir do passo 1 do algoritmo de mapeamento, já que **categories** é uma entidade forte. Apenas foi criada a relação com os seus atributos.