# Tablas básicas

Tabla **COUNTRIES**

* Esta table es para los países.
* Por ahora solo estará Colombia
* Se utiliza para saber a qué país pertenece una tienda
* La información del país se ingresa manualmente en la DB

Tabla **STORES**

* Esta table es para las tiendas.
* Se guarda información básica como el Nombre de la tienda.
* Se utiliza para identificar una prenda (CLOTHE) a qué tienda pertenece.
* La información de la tienda se ingresa manualmente en la DB a medida que vayamos agregando más

Tabla **CATEGORIES** y **SUBCATEGORIES**

* Se utiliza para guardar la categoría y subcategoría de una prenda.
* Ejemplo #1: Categoría ROPA subcategoría CAMISETAS
* Ejemplo #2: Categoría ACCESORIOS subcategoría GORRAS

Tabla **SIZES** y **COLORS**

* Se guarda tallas en una tabla y colores en otras.

Tabla **IMAGES**

* Se guarda la URL de una imagen por individual

Tabla **CLOTHES**

* La información de la prenda
* El atributo REFERENCE es el código único de cada tienda

# Tablas mapas

Tabla **CLOTHES\_COLORS\_SIZES\_MAP** (MUCHOS A MUCHOS)

* Es una tabla que permite mapear:
  + Prenda (CLOTHES)
  + Color de la prenda (COLORS)
  + Tamaño del color de la prenda (SIZES)
* Ejemplo:

|  |  |  |
| --- | --- | --- |
| **Reloj para Hombre #1** | **Amarillo** | **L** |
| **Reloj para Mujer #1** | **Verde** | **L** |
| **Reloj para Hombre #1** | **Amarillo** | **XL** |

Tabla **CLOTHES\_IMAGES\_MAP** (MUCHOS A MUCHOS)

* Es una tabla que permite mapear:
  + Prenda
  + Imagen de la prenda
* Ejemplo:

|  |  |
| --- | --- |
| **Reloj para Hombre #1** | **https://imagen1.com** |
| **Reloj para Mujer #1** | **https://imagen2.com** |
| **Reloj para Hombre #1** | **https://imagen3.com** |

# Lógica ETL

Para guardar **COLORES, TALLAS, IMÁGENES, CATEGORÍAS y SUBCATEGORÍAS**

* Verificar que el registro no exista en la DB
* Si existe, coger la referencia.
* Si no existe, crear el registro y coger la referencia.
* Con la referencia se guardaría la ropa y se pondrían las referencias en las TABLAS MAPA

Para guardar **PRENDAS**

* Si ya existe:
  + Actualizar sus atributos
  + Actualizar sus colores y tallas
  + Actualizar el atributo **SOLD\_OUT** (agotado) a **false**
* Si ya existe en la DB, pero no vino en la data scrapeada:
  + Setear el atributo **SOLD\_OUT** (agotado) como **true**
* Si no existe:
  + Guardar la prenda
  + Guardar las tallas y colores de la prenda
  + Guardar imágenes de la prenda
  + Guardar la referencia (REFERENCE) con el siguiente formato:
    - {tienda}-{reference}
    - Ejemplos:
      * DIESEL-A12394PD
      * ZARA-12384ASD34
      * ZARA-ASD8123F

# Atributos data Scrapeada

* **Uuid**: índice del registro scrapeado (se puede ignorar)
* **Product\_name**: Nombre de la prenda
* **Image\_urls**: URLs de las imágenes de la prenda
* **Images**: si las imágenes se guardan en AWS S3 o en local, tiene información de donde están (POR AHORA NO LAS GUARDAMOS)
* **Price**: precio de la prenda
* **Old\_price**: Si la prenda tiene descuento, este es el precio viejo y “PRICE” sería el precio con descuento
* **Disc**: Descuento de la prenda
* **URL**: Url de la prenda
* **Color\_SKU**: Tallas y colores de la prenda
* **Description**: Descripción de la prenda
* **Reference**: Código único de la prenda en la tienda.
* **Category**: Categoría de la prenda. Ejemplo: ACCESORIOS
* **Subcategory**. Subcategoría de la prenda. Ejemplo: GORRAS
* **Subcategory\_url**: URL de la subcategoría (OPCIONAL)
* **Genre**: Género de la prenda. (Masculino, Femenino o Unisex)
* **Store**: Nombre de la tienda de la prenda