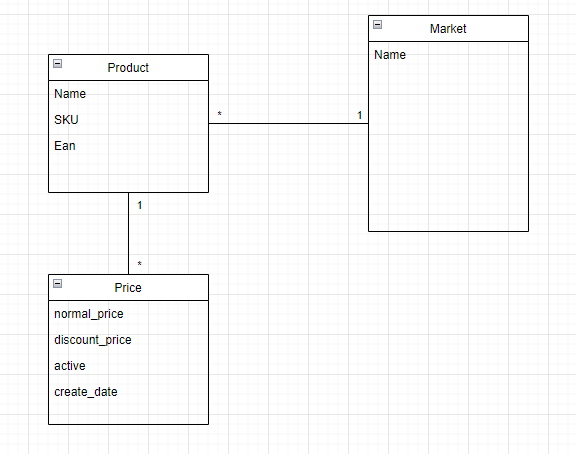
Prueba técnica



**Imagen 1**: Diagrama sobre precios de los productos. SKU es un valor único que lo relaciona con el mercado, Ean es un identificador único que lo provee el proveedor del producto a nivel mercado. Un producto en diferentes mercados puede tener el mismo Ean, pero diferente SKU.

# Preguntas:

1. Mediante una query SQL, obtener los productos indicando su último menor precio activo, EAN, SKU y su mercado (market).

Supuestos:

* La frase: ‘su último menor precio activo’ hace referencia a que existen más de un precio activo histórico, es decir, obtener de los precios activos el último, por lo que bastaría con filtrar por activo, obtener la última fecha y obtener (en caso de empate) el menor precio.
* El atributo active será de tipo booleano (True o False).
* El precio activo o real será obteniendo la resta entre normal\_price y discount\_price, en donde este último nunca será mayor a normal\_price.
* El identificado o Primary Key del producto será su SKU brindado por el mercado.
* Dada la relación entre productos y precio, para que sea de 1 a 0 o muchos, es necesario que ‘Price’ tenga una FK que lo relacione a productos, en este caso será su SKU.
* Para establecer la relación de 1 a 0 o muchos que tiene productos y el mercado, es necesario definir una FK que apunte a la PK del mercado, por lo que se define un atributo ID\_Mercado que será único para cada mercado

SELECT Product.SKU, Product.Name, Product.Ean, Market.name, MIN(normal\_price-discount\_price) as price

FROM Product

INNER JOIN Market ON Product.ID\_Market = Market.ID\_Market

INNER JOIN Price ON Product.SKU = Price.SKU

WHERE Price.active = true

GROUP BY Product.SKU

ORDER BY MAX(Price.create\_data) DESC

1. Teniendo en cuenta el modelo de la *imagen 1*, describa con palabras un proceso que tenga que ser ejecutado cada cierto tiempo (automatización) y que obtenga información “relevante” diferente a la pedida en la *pregunta 1*.

#Seguir pensando

1. Basándose en los datos obtenidos en la respuesta de la *pregunta 1*.
   1. Escribir una función en Python que recorra los datos y agrupe los productos mediante su Ean en el siguiente arreglo de diccionarios

[{

“Ean”: {

“nombre producto (asumir que los productos con mismo Ean tienen el mismo nombre)”,

“datos de la query”:[“listado de los datos que vienen en la query”,],

“cantidad de markets diferentes”,

“rango de precios (Mayor precio - Menor precio)”

},

},]

* 1. El diccionario anterior debe ser recorrido mediante React, en donde se debe crear componentes HTML que contendrán el nombre del producto, el rango de precios y en cuántos mercados diferentes se encuentra. Poner un input en donde se puede filtrar los productos que se ven por su nombre y utilizando hooks (useEffect) hacer que los productos que no se deben ver vayan desapareciendo 1 a 1 cada segundo.

Consideraciones:

1. No es necesario que el HTML tenga CSS o algún tipo de estilo.
2. El arreglo de diccionarios puede estar definido como constante dentro de React y su data debe ser inventada, al menos 3 Eans diferentes.
3. Al cargar la página se debe poder ver el HTML creado