11 de mayo de 2022

Valentina valenzuela morgado

minería de datos y big data

Universidad San Sebastián

Taller 1

inteligencia de negocios

Contenido

[Introducción 2](#_Toc103189811)

[KPI 3](#_Toc103189812)

[Modelo Lógico Multidimensional 4](#_Toc103189813)

[Base de Datos 5](#_Toc103189814)

[Modelo físico 5](#_Toc103189815)

[Carga Inicial 8](#_Toc103189816)

[Dimensión Marca de Productos 8](#_Toc103189817)

[Dimensión Categoría de Productos 8](#_Toc103189818)

[Dimensión Producción de Productos 2016 9](#_Toc103189819)

[Dimensión Información de Tiendas 10](#_Toc103189820)

[Carga Mensual 11](#_Toc103189821)

[Cuadro de Mando 13](#_Toc103189822)

[Conclusión 15](#_Toc103189823)

[Referencias 16](#_Toc103189824)

# Introducción

Este taller de Minería de Datos y Big Data corresponde a un proyecto de Inteligencia de Negocios que permitirá generar un cuadro de mando para obtener información de ventas de bicicletas a través de KPIs corporativos. Para lograr este objetivo, se utilizará un modelo relacional de la base de datos previamente entregado por el profesor a cargo, el cual posee información de ventas, detalle de ventas, tienda, clientes y los productos principalmente.

# KPI

Los KPI (indicadores clave de rendimiento) son medidas cuantitativas y cualitativas que se utilizan para revisar el progreso de una organización en relación con sus objetivos. Son medidas que un sector u organización usa para definir el éxito y hacer un seguimiento del progreso en el cumplimento de sus objetivos estratégicos.

Representan un conjunto de medidas que se centran en los aspectos del desempeño organizacional que son los más críticos para el éxito actual y el futuro de la organización, estos muestran las tendencias a lo largo del tiempo y también miden el desempeño para comprar los resultados con los estándares u otras organizaciones similares.

Para gestionar la información de ventas de bicicletas y en base a la definición de un KPI, formulando las preguntas de gestión debidas y de acuerdo con estas se establecieron los siguientes KPIs:

* Ventas por localidad: Se puede observar que, en la base de datos sobre las tiendas, estas tienen distintas ubicaciones, por lo que es óptimo encontrar cual es la localidad que tiene más ventas y/o demanda.
* Modelo más vendido: En base a la producción de bicicletas, creo debido tomar este KPI ya que se puede obtener más información acerca de la oferta/demanda de los productos y poder conocer cuáles son los más vendidos en el mercado.

# Modelo Lógico Multidimensional

Diagrama

Descripción generada automáticamente

# Base de Datos

Se quiere garantizar una correcta legibilidad de datos, así como la implementación precisa y la posterior carga de estos datos, por lo que, se utilizará base de datos Staging para los distintos requisitos junto con la implementación de un modelo físico. Generalmente Data Staging se utiliza para la extracción de datos de una base de datos mayor y/o cargar otras bases de datos. Todo esto para asegurar que no se produzcan errores durante la creación y la posterior carga del Data WareHouse.

## Modelo físico

Para realizar el modelo físico de la base de datos se utilizará el lenguaje SQL junto con el programa Microsoft SQL Server. El código del modelo físico tiene en cuenta la arquitectura del modelo lógico y los datos que hay que extraer para las variables, lo que da lugar a varias tablas que sirven de conducto entre los datos y el Data WareHouse.

CREATE TABLE production.categories (

category\_id INT IDENTITY (1, 1) PRIMARY KEY,

category\_name VARCHAR (255) NOT NULL

);

CREATE TABLE production.brands (

brand\_id INT IDENTITY (1, 1) PRIMARY KEY,

brand\_name VARCHAR (255) NOT NULL

);

CREATE TABLE production.products (

product\_id INT IDENTITY (1, 1) PRIMARY KEY,

product\_name VARCHAR (255) NOT NULL,

brand\_id INT NOT NULL,

category\_id INT NOT NULL,

model\_year SMALLINT NOT NULL,

list\_price DECIMAL (10, 2) NOT NULL,

FOREIGN KEY (category\_id) REFERENCES production.categories (category\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (brand\_id) REFERENCES production.brands (brand\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE sales.customers (

customer\_id INT IDENTITY (1, 1) PRIMARY KEY,

first\_name VARCHAR (255) NOT NULL,

last\_name VARCHAR (255) NOT NULL,

phone VARCHAR (25),

email VARCHAR (255) NOT NULL,

street VARCHAR (255),

city VARCHAR (50),

state VARCHAR (25),

zip\_code VARCHAR (5)

);

CREATE TABLE sales.stores (

store\_id INT IDENTITY (1, 1) PRIMARY KEY,

store\_name VARCHAR (255) NOT NULL,

phone VARCHAR (25),

email VARCHAR (255),

street VARCHAR (255),

city VARCHAR (255),

state VARCHAR (10),

zip\_code VARCHAR (5)

);

CREATE TABLE sales.staffs (

staff\_id INT IDENTITY (1, 1) PRIMARY KEY,

first\_name VARCHAR (50) NOT NULL,

last\_name VARCHAR (50) NOT NULL,

email VARCHAR (255) NOT NULL UNIQUE,

phone VARCHAR (25),

active tinyint NOT NULL,

store\_id INT NOT NULL,

manager\_id INT,

FOREIGN KEY (store\_id) REFERENCES sales.stores (store\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (manager\_id) REFERENCES sales.staffs (staff\_id) ON DELETE NO ACTION ON UPDATE NO ACTION

);

CREATE TABLE sales.orders (

order\_id INT IDENTITY (1, 1) PRIMARY KEY,

customer\_id INT,

order\_status tinyint NOT NULL,

/\* Estado de la orden: 1 = Pendiente; 2 = Procesando; 3 = Rechazada; 4 = Completada \*/

order\_date DATE NOT NULL,

required\_date DATE NOT NULL,

shipped\_date DATE,

store\_id INT NOT NULL,

staff\_id INT NOT NULL,

FOREIGN KEY (customer\_id) REFERENCES sales.customers (customer\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (store\_id) REFERENCES sales.stores (store\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (staff\_id) REFERENCES sales.staffs (staff\_id) ON DELETE NO ACTION ON UPDATE NO ACTION

);

CREATE TABLE sales.order\_items(

order\_id INT,

item\_id INT,

product\_id INT NOT NULL,

quantity INT NOT NULL,

list\_price DECIMAL (10, 2) NOT NULL,

discount DECIMAL (4, 2) NOT NULL DEFAULT 0,

PRIMARY KEY (order\_id, item\_id),

FOREIGN KEY (order\_id) REFERENCES sales.orders (order\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (product\_id) REFERENCES production.products (product\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE production.stocks (

store\_id INT,

product\_id INT,

quantity INT,

PRIMARY KEY (store\_id, product\_id),

FOREIGN KEY (store\_id) REFERENCES sales.stores (store\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (product\_id) REFERENCES production.products (product\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

## Carga Inicial

La carga comienza con la información importante que permanecerá sin cambios y marcará los primeros pasos para verificar que la base de datos está construida correctamente. Para el inicio de esta carga se considerará las marcas de los productos y su categoría, junto con el año del modelo filtrado por 2016 y su precio, también se incluirá información de las tiendas tales como ubicación, nombre y métodos de contacto. Dando como resultado los siguientes códigos:

### Dimensión Marca de Productos

BEGIN

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(1,'Electra')

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(2,'Haro')

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(3,'Heller')

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(4,'Pure Cycles')

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(5,'Ritchey')

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(6,'Strider')

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(7,'Sun Bicycles')

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(8,'Surly')

INSERT INTO production.brands(brand\_id,brand\_name) VALUES(9,'Trek')

END

### Dimensión Categoría de Productos

BEGIN

INSERT INTO production.categories(category\_id,category\_name) VALUES(1,'Children Bicycles')

INSERT INTO production.categories(category\_id,category\_name) VALUES(2,'Comfort Bicycles')

INSERT INTO production.categories(category\_id,category\_name) VALUES(3,'Cruisers Bicycles')

INSERT INTO production.categories(category\_id,category\_name) VALUES(4,'Cyclocross Bicycles')

INSERT INTO production.categories(category\_id,category\_name) VALUES(5,'Electric Bikes')

INSERT INTO production.categories(category\_id,category\_name) VALUES(6,'Mountain Bikes')

INSERT INTO production.categories(category\_id,category\_name) VALUES(7,'Road Bikes')

END

### Dimensión Producción de Productos 2016

BEGIN

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(1,'Trek 820 - 2016',9,6,2016,379.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(2,'Ritchey Timberwolf Frameset - 2016',5,6,2016,749.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(3,'Surly Wednesday Frameset - 2016',8,6,2016,999.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(4,'Trek Fuel EX 8 29 - 2016',9,6,2016,2899.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(5,'Heller Shagamaw Frame - 2016',3,6,2016,1320.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(6,'Surly Ice Cream Truck Frameset - 2016',8,6,2016,469.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(7,'Trek Slash 8 27.5 - 2016',9,6,2016,3999.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(8,'Trek Remedy 29 Carbon Frameset - 2016',9,6,2016,1799.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(9,'Trek Conduit+ - 2016',9,5,2016,2999.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(10,'Surly Straggler - 2016',8,4,2016,1549)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(11,'Surly Straggler 650b - 2016',8,4,2016,1680.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(12,'Electra Townie Original 21D - 2016',1,3,2016,549.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(13,'Electra Cruiser 1 (24-Inch) - 2016',1,3,2016,269.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(14,'Electra Girl''s Hawaii 1 (16-inch) - 2015/2016',1,3,2016,269.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(15,'Electra Moto 1 - 2016',1,3,2016,529.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(16,'Electra Townie Original 7D EQ - 2016',1,3,2016,599.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(17,'Pure Cycles Vine 8-Speed - 2016',4,3,2016,429)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(18,'Pure Cycles Western 3-Speed - Women''s - 2015/2016',4,3,2016,449)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(19,'Pure Cycles William 3-Speed - 2016',4,3,2016,449)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(20,'Electra Townie Original 7D EQ - Women''s - 2016',1,3,2016,599.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(21,'Electra Cruiser 1 (24-Inch) - 2016',1,1,2016,269.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(22,'Electra Girl''s Hawaii 1 (16-inch) - 2015/2016',1,1,2016,269.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(23,'Electra Girl''s Hawaii 1 (20-inch) - 2015/2016',1,1,2016,299.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(24,'Electra Townie Original 21D - 2016',1,2,2016,549.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(25,'Electra Townie Original 7D - 2015/2016',1,2,2016,499.99)

INSERT INTO production.products(product\_id, product\_name, brand\_id, category\_id, model\_year, list\_price) VALUES(26,'Electra Townie Original 7D EQ - 2016',1,2,2016,599.99)

END

### Dimensión Información de Tiendas

BEGIN

INSERT INTO sales.stores(store\_name, phone, email, street, city, state, zip\_code)

VALUES('Santa Cruz Bikes','(831) 476-4321','santacruz@bikes.shop','3700 Portola Drive', 'Santa Cruz','CA',95060),

('Baldwin Bikes','(516) 379-8888','baldwin@bikes.shop','4200 Chestnut Lane', 'Baldwin','NY',11432),

('Rowlett Bikes','(972) 530-5555','rowlett@bikes.shop','8000 Fairway Avenue', 'Rowlett','TX',75088);

END

## Carga Mensual

Para la carga mensual de información se utilizarán los datos de órdenes de venta del año 2017, cargando mensualmente la fecha en la cual se ordenó, la fecha de pedido y la fecha de envío por lo que el código quedaría de la siguiente forma:

BEGIN

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(636,532,4,'20170103','20170104','20170105',1,2);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(637,512,4,'20170103','20170105','20170105',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(638,1302,4,'20170104','20170105','20170106',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(639,550,4,'20170106','20170107','20170107',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(640,1093,4,'20170106','20170108','20170107',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(641,673,4,'20170107','20170110','20170108',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(642,847,4,'20170107','20170109','20170110',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(643,901,4,'20170108','20170109','20170111',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(644,439,4,'20170108','20170111','20170110',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(645,1199,4,'20170108','20170111','20170110',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(646,975,4,'20170109','20170112','20170111',1,2);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(647,283,4,'20170109','20170110','20170111',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(648,684,4,'20170109','20170111','20170111',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(649,1153,4,'20170110','20170111','20170113',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(650,729,4,'20170111','20170114','20170112',1,2);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(651,190,4,'20170111','20170113','20170113',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(652,985,4,'20170112','20170115','20170113',3,9);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(653,1249,4,'20170114','20170117','20170115',1,2);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(654,1129,4,'20170114','20170115','20170117',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(655,347,4,'20170116','20170117','20170118',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(656,949,4,'20170116','20170117','20170118',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(657,349,4,'20170117','20170119','20170120',1,2);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(658,1051,4,'20170117','20170118','20170118',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(659,1391,4,'20170117','20170119','20170118',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(660,383,4,'20170118','20170119','20170119',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(661,626,4,'20170118','20170119','20170120',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(662,1125,4,'20170119','20170120','20170120',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(663,344,4,'20170120','20170122','20170123',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(664,681,4,'20170120','20170123','20170122',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(665,29,4,'20170121','20170122','20170123',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(666,770,4,'20170121','20170122','20170124',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(667,858,4,'20170122','20170124','20170125',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(668,297,4,'20170122','20170123','20170124',3,8);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(669,745,4,'20170123','20170124','20170126',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(670,361,4,'20170124','20170125','20170125',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(671,922,4,'20170125','20170128','20170126',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(672,908,4,'20170126','20170129','20170129',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(673,64,4,'20170126','20170128','20170127',3,9);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(674,733,4,'20170127','20170128','20170129',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(675,291,4,'20170128','20170131','20170130',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(676,790,4,'20170128','20170130','20170131',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(677,809,4,'20170128','20170131','20170130',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(678,250,4,'20170128','20170129','20170131',3,9);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(679,81,4,'20170129','20170130','20170130',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(680,245,4,'20170129','20170201','20170130',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(681,499,4,'20170129','20170201','20170131',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(682,614,4,'20170129','20170130','20170130',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(683,741,4,'20170129','20170131','20170201',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(684,762,4,'20170131','20170201','20170202',2,6);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(685,976,4,'20170131','20170203','20170202',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(686,380,4,'20170201','20170203','20170204',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(687,530,4,'20170202','20170205','20170204',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(688,343,4,'20170202','20170203','20170205',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(689,884,4,'20170202','20170203','20170205',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(690,1097,4,'20170203','20170204','20170206',1,2);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(691,1081,4,'20170204','20170206','20170206',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(692,2,3,'20170205','20170205',NULL,1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(693,1233,4,'20170205','20170207','20170206',1,3);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(694,58,4,'20170205','20170206','20170208',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(695,606,4,'20170205','20170208','20170207',2,7);

INSERT INTO sales.orders(order\_id, customer\_id, order\_status, order\_date, required\_date, shipped\_date, store\_id,staff\_id) VALUES(696,1242,4,'20170205','20170208','20170208',2,7);

END

# Cuadro de Mando

Los cuadros de mando permiten obtener una mejor visualización de los KPIs para poder apreciar mejor los resultados de esta y poder sacar conclusiones acerca de la gestión de ventas de bicicletas.

En cuanto al primer KPI de ventas por localidad se puede observar que de la tabla órdenes de venta, la tienda número 2, la cual sería “Baldin Bikes” ubicada en el estado de Nueva York, obtiene el primer lugar con más de 1000 ventas, por lo que también es la que más demanda tiene y la que mayor producto necesita para vender.

Gráfico, Gráfico de proyección solar

Descripción generada automáticamente

Gráfico

Descripción generada automáticamente

Para el segundo KPI se visualizó cual sería el modelo más vendido, sacando en base al cuadro de mando, que el modelo más vendido sería el número 6 “Surly Ice Cream Truck Frameset – 2016” con un total de 167 productos vendidos

Gráfico, Gráfico de proyección solar

Descripción generada automáticamente

Imagen que contiene instrumento, estacionaria, lápiz, colorido

Descripción generada automáticamente

# Conclusión

Para terminar este informe, se pudo observar que se logró el objetivo propuesto en la introducción, en donde se utilizó un modelo relacional de la base de datos con información de ventas de bicicletas y de estas se pudo obtener un análisis sobre la gestión de estas ventas a través de los KPIs corporativos.

En base al uso del Data WareHouse como método para almacenar información en la base de datos, este nos permite organizar los datos y obtener información sobre ellos, por lo que fue más fácil profundizar en los datos que se estaban utilizando y así obtener la información requerida de una forma más sencilla.

# Referencias

1. Sepúlveda, M. (Abril 2022). Data Mining & Bigdata - Clase 4 Visualización. Classroom.