**Solución a Actividad Guiada 2 -Manejo de conjuntos de datos.**

* Carga de datos de csv(estaciones\_bici.csv) en un DataFrame

**Respuesta: Utilizando** el lenguaje de programación Python con la librería Pandas, se consiguió cargar en una variable de tipo de DataFrame todos los registros de este archivo csv.

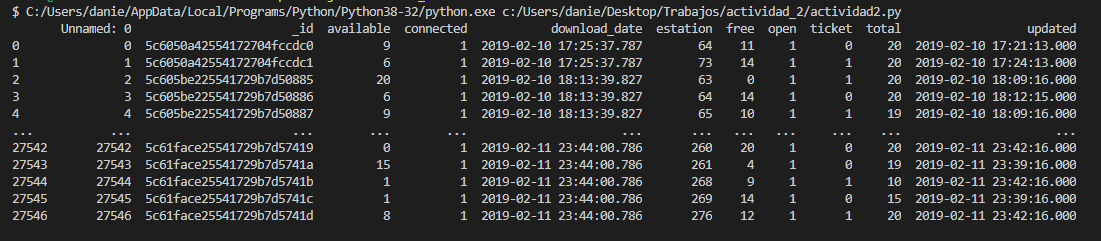
# import numpy as np

import pandas as pd

estaciones = pd.read\_csv('estaciones\_bici.csv', sep=";")

# print(estaciones)

El resultado fue una impresión por consola:



* Obtención de número de estaciones con total de 30:

**Respuesta:**

Para filtrar el DataFrame por la cantidad de estaciones que tienen de total 30, se utilizo el siguiente algoritmo:

totales = estaciones[estaciones["total"] == 30]

estations = totales.estation.drop\_duplicates()

print(estations.count())

El resultado del código anterior fue:



* Número de estación con la media más alta de bicis disponibles

**Respuesta:**

estations = totales.estation.drop\_duplicates()

# print(estations.count())

# print(estaciones)

medias = estaciones[["available", "estation"]]

aux = 0

estation\_max\_mediun = 0

for i in medias.estation.drop\_duplicates():

    media = medias[medias["estation"].isin([i])].median()

    # print("Estacion: ", i, " Media: ", media.available)

    if media.available>aux:

        aux = media.available

        estation\_max\_mediun = media.estation

print("Estacion: ", estation\_max\_mediun, " Media: ", aux)

# media = estaciones.estation.median()

# print(medias)

El resultado del algoritmo anterior fue:



* Realizar el histograma de la estación de bicis disponibles:

Para sacar el histograma de la estación de bicis disponibles se utiliza el siguiente código:

data\_estation\_max\_mediun.hist(column="available")

plt.show()

El resultado es:

