

# Tarea 11 - Dataframes con el módulo **pandas**

## Curso de Python

### Ejercicio 1

Crea un dataframe con 15 filas y 2 columnas. La primera columna se llamará **x**, la segunda **y**. Cada entrada será un número real a tu elección. Guarda el dataframe en una variable llamada **points**.

### Ejercicio 2

Del dataset **points**, muestra las filas cuyo valor en la columna **x** sea positivo.

### Ejercicio 3

Del dataset **points**, muestra las filas cuyo valor en la columna **y** sea negativo. Usa el método **.query()**.

### Ejercicio 4

Del dataset **points**, muestra las observaciones cuyos puntos (**x**, **y**) pertenezcan al primer cuadrante. Usa el método **.query()**.

### Ejercicio 5

Del dataset **points**, muestra las observaciones de la forma: “El punto (**{x}**, **{y}**) **{no/sí}** pertenece al primer cuadrante”.

### Ejercicio 6

Crea un dataframe con 10 filas y 4 columnas. La primera columna se llamará **word**; la segunda, **length**; la tercera, **vowels**; y la última, **consonants**. La columna **words** contendrá las siguientes 10 palabras: “euro”, “diez”, “algas”, “broma”, “cicutá”, “fatiga”, “nachos”, “jadeos”, “hazañas”, “boutique”. Las columnas **length**, **vowels** y **consonants** contendrán, respectivamente, la longitud, el total de vocales y el total de consonantes. Guarda el dataframe en la variable **words**.

### Ejercicio 7

Muestra las 10 observaciones de **words** con el formato “La palabra **{word}** tiene **{length}** letras, de las cuales **{vowels}** son vocales y **{consonants}** consonantes”.

## Ejercicio 8

Muestra aquellas observaciones de `words` que tienen el mismo número de vocales y consonantes. Usa el método `.query()`.

## Ejercicio 9

Investiga el método `.sort_values()` para ordenar las observaciones según la longitud de las palabras en orden descendente.

## Ejercicio 10

Convierte la columna `vowels` a lista y, con `sorted()` ordénala de mayor a menor. Investiga para ello el método `.tolist()`.