

Curso Hackea Tu Futuro – Python Granada (Módulo 4)

Nombre: Andrea

Apellidos: Sánchez Rojas

Fecha: 06/10/2023

Responde a las siguientes preguntas. Justifica la respuesta.

1. Tipos de archivos: ¿Qué es un archivo JSON? ¿Y un archivo CSV?

JSON es un formato de texto de JavaScript y tiene como objetivo almacenar e intercambiar datos. Formato estándar en la industria de la tecnología.

Un archivo CSV es un archivo de texto que permite que los datos se guarden en estructura de tabla.

2. ¿Qué es un ETL?

Los procesos de ETL es extraer, transformar y cargar datos a un sistema a partir del cual yo pueda hacer mis análisis.

3. ¿Para qué sirve el comando pip? ¿Y el comando import?

El comando pip sirve para poder instalar estancias o bibliotecas en Python para trabajar sobre él.

El comando import sirve para cargar archivos o bibliotecas en Python y así poder trabajar con ellos.

No se puede realizar un import sin haber instalado antes un pip.

4. Enumera las características de Python explicando cada una de ellas (menciona 4 al menos).

Lenguaje multiplataforma. Python puede ejecutarse en diferentes dispositivos y sistemas operativos.

Open source (de código abierto). Cualquier persona puede utilizarlo.

De alto nivel. Utiliza palabras clave e instrucciones parecidas al lenguaje humano, resultando más fácil de comprender y usar.

Orientado a objetos. El diseño de software se organiza alrededor de datos u objetos, en vez de usar funciones y lógica.

5. ¿Cuáles son los tipos primitivos en Python y qué valores pueden contener cada uno de ellos? Pista: Son 3 tipos.

Los tipos más sencillos son los tipos primitivos y existen 3: los textos, números y los booleanos.

- **Los números.** Nos centraremos en los enteros y los flotantes. Los números son los números tanto positivos como negativos pero sin decimales. Los flotantes son positivos, negativos pero con decimales. Los decimales en python se ponen con un punto . no con una coma.
- **Los textos.** Cualquier tecla que tengas en el teclado, no sólo las letras. Cómo se diferencia si escribo *patata*. Para que python sepa que es texto, tiene que ir siempre entre comillas. 'Patata', "Patata" comillas triples dobles, a ojos de python es lo mismo. Caracteres especiales también o incluso los números, si pongo un número entre comillas ya no es un número sino un texto.
- **Los booleanos.** Sólo hay dos y nos ayudan a expresar estados: True y False (es sensible a las mayúsculas y minúsculas).

6. Menciona alguna estructura de datos más compleja que los tipos primitivos que conozcas.

Existen varios, como las listas, diccionarios, operadores, formato de cadena (string), condicionales, bucles, funciones, módulos y paquetes.

7. Escribe la sintaxis para crear variables.

¿Cómo se crea una variable?

```
nombre_variable = tipo_de_texto
```

El nombre de la variable no puede comenzar por un número, no puede contener espacios y la tercera norma es que el nombre de la variable no puede ser una palabra reservada (una palabra que ya tiene un uso asignado). Todas las palabras reservadas en python están en inglés.

```
numero = 2  
texto = 'Luis'
```

Un dataset puede ser una variable, es un tipo de dato más complejo pero la idea es la misma.

8. Escribe la sintaxis para crear funciones.

Son como las recetas. No podemos comenzar la receta hasta que no hemos comprado todo los ingredientes que necesitamos ya que si nos quedamos cortos, no va a salir y si nos pasamos, saldrá otra cosa. Seguiremos una serie de pasos en el mismo orden y obtendremos un resultado. Una serie de cálculos que se dan siempre en el mismo orden. Cuando veamos paréntesis por ahí, eso significa que es una función. Una variable no lleva paréntesis.

```
def sum(a, b):  
    resultado = a + b  
    return resultado
```

9. Escribe la sintaxis para llamar a variables.

Para acceder al valor de una variable, debemos escribir un nombre de la variable y asignarle un valor con el símbolo =

nombre_variable = tipo_de_texto

Después, volvemos a escribir el nombre de la variable y si la hemos guardado correctamente, aparecerá su valor. Por ejemplo:

```
nombre_estudiante = 'Andrea'  
nombre_estudiante
```

10. Escribe la sintaxis para llamar a funciones.

Para invocar una función en Python, hay que escribir el nombre de la función seguido de paréntesis

```
nombre_funcion()
```

11. Explica con tus palabras para qué sirven los condicionales y los bucles en Python.

Cuando queremos hacer algo más de una vez (iterar) necesitamos un bucle. En otras palabras, los bucles se utilizan para repetir varias veces la ejecución de una parte de un programa y existen dos tipos:

Los bucles **for**. Cuando se sabe de antemano cuántas veces se va a repetir el programa.

Los bucles **while**. Cuando no se conoce de antemano el número de repeticiones del programa.

```
#BUCLE WHILE

palabra = 'abcd'

indice = 0

while indice < len(palabra):

    print(palabra[indice])

    indice = indice + 1

a
b
c
d

#BUCLE FOR

genres = ['rock', 'r&b', 'soundtrack', 'r&b', 'soul', 'pop']

for genre in genres:

    print(genre)

rock
r&b
soundtrack
r&b
soul
pop
```

12. Explica con tus palabras para qué sirven las librerías: Pandas y Numpy.

Pandas es la librería que nos permite realizar análisis de datos y representarlos en tablas. Sería como instalar una especie de Excel.

Numpy es una librería que nos permite hacer cálculos en Python. Sería algo así como instalar una especie de calculadora que nos ayudará a realizar operaciones matemáticas.

13. ¿Cómo representamos el valor vacío en Python?

Para representar un valor vacío (ausencia de valor), se indica como None, ya que es un valor reservado utilizado para indicar que una variable o una función no tienen un valor válido.

Algo que se suele hacer cuando tenemos un valor None, por ejemplo es hacer la media de todos los valores que tenemos y posteriormente, sustituirlos en cada None.

14. ¿Qué es un IDE?

Significa Entorno de Desarrollo Integrado. Es una aplicación informática que nos proporciona distintas herramientas y funcionalidades para desarrollar más cómodamente programas.

15. ¿Qué es el CRISP-DM?

Es una metodología parecida a un CRM. Lo emplean aquellas empresas que manejan grandes volúmenes de datos para facilitar la tarea de extracción, transformación, carga y posteriormente, análisis de los datos.

