Capítulo 3:

# Vídeos creados

En este capítulo se da una explicación detallada sobre el material audiovisual creado para la elaboración de este TFG.

En cada apartado se expondrán los objetivos de cada uno de los vídeos, junto con un índice de los contenidos, así como una explicación resumida del mismo. También, se añaden los problemas encontrados en la realización de cada vídeo y una conclusión que incluye una vista global del vídeo y un aprendizaje personal de cara a la mejora para futuros vídeos.

## Vídeo 1: Introducción a la programación con Python

### Descripción general

En este vídeo (Figura 3.1) se habla de Python como lenguaje de programación, se hace un análisis sobre sus principales características como su estructura, la funcionabilidad como lenguaje de programación, funcionabilidad referente al código y el impacto que tiene en la actualidad. Siendo el primero de una serie de vídeos para el aprendizaje de la programación en Python. En este vídeo se ha tratado las bases del lenguaje, se ha explicado los aspectos fundamentales de este lenguaje que lo hacen hoy en día el lenguaje más utilizado.

*(Figura 3.1)*

### Objetivos

El objetivo de este vídeo es que los espectadores que no tienen ningún conocimiento de programación se sientan cómodos y familiarizados con Python. Además, para usuarios que sí tengan alguna base de programación, se proporciona una visión global de las características fundamentales de Python para comenzar con una buena base para el aprendizaje de este lenguaje. También, se muestra por qué Python está teniendo un gran impacto en la actualidad gracias a sus características mencionadas en el vídeo.

Debido a que es un vídeo introductorio, se inspira a los espectadores a aprender Python destacando las razones que hacen de Python un lenguaje valioso, y las posibilidades que ofrece en la actualidad junto con el potencial de crecimiento.

### Descripción del vídeo

Curso de programación en Python.

Vídeo 1: Se expone una introducción al lenguaje y el por qué está entre los lenguajes más empleados hoy en día (es el más empleado, al menos en 2023 según la web TIOBE). Con algo más de detalle, este vídeo explica:

* Características de Python: creador, de código abierto, legibilidad, es interpretado y de alto nivel.
* Estructura del código: indentación y su error
* Funcionabilidad del código: tipado dinámico, biblioteca estándar y bibliotecas de terceros y diferencia entre biblioteca, paquete y módulo.
* Ejemplo: paquete “datetime”, módulos “date” y “math”.
* Funcionabilidad del código: versatilidad, lenguaje multiparadigma y multiplataforma.
* Relevancia actual: crecimiento constante, es el lenguaje más utilizado según TIOBE, tiene una comunidad activa y facilita ser efectivos.

## Vídeo 2: Herramientas necesarias para programar en Python

### Descripción general

En este vídeo (Figura 3.2) se ha explicado de forma detallada cómo descargar e instalar Python. También, se explica el manejo del intérprete interactivo de Python, permitiendo al usuario poder adentrarse en la programación mediante la práctica con comandos y su resultado. Además, se añade una alternativa para la realización de programas, Visual Studio Code, que facilita la escritura de código. Se explica cómo utilizar de manera detallada Visual Studio Code para poder comenzar a escribir código en un archivo. De forma introductoria se realiza un programa básico, para explicar cómo ejecutar un programa y su salida por pantalla.

*(Figura 3.2)*

### Objetivos

El objetivo de este vídeo es que el usuario pueda descargar e instalar de manera fácil y rápida los recursos necesarios para la programación con Python. Además, también se introduce al espectador con un primer programa para que entienda los conceptos básicos sobre el desarrollo de un programa y su traducción al lenguaje común.

### Descripción del vídeo

Curso de programación en Python**.**

Vídeo 2: Se explican los programas que necesitamos para empezar a programar en Python:

* Descargar e instalar el software de Python desde https://www.python.org/downloads/
* Visión básica del Intérprete Interactivo de Python: operaciones aritméticas, ventajas einconvenientes.
* Visual Studio Code (VSCode): un editor de código muy versátil.
* Descargar e instalar VSCode desde https://code.visualstudio.com/
* Instalar la extensión de Python para VSCode.
* Opciones útiles: crear carpeta o archivo .py (de Python), refrescar, ejecutar...
* Función print() para escribir un texto en la terminal (pantalla de salida).

## Vídeo 3: Variables y tipos de datos en Python

### Descripción general

En este vídeo (Figura 3.3), se explica el concepto de variable y la forma en la que una variable es usada en programación. Se indican las reglas necesarias para declarar correctamente una variable y se muestran las palabras reservadas que no pueden ser utilizadas como nombre. De manera adicional, se aportan consejos sobre las variables que están estandarizados entre la comunidad de programadores, aunque no son obligatorios.

Los tipos de datos simples es otro tema del que se habla en el vídeo, explicando su significado y los tipos que hay dentro de python. De manera individual se explican cada uno de los tipos de datos simples aportando información como el valor que representan, nomenclatura y otros aspectos de relevancia.

Finalmente se da una explicación para que el usuario comprenda qué es la conversión entre tipos de datos y las formas en la que puede realizarla.

*(Figura 3.3)*

### Objetivos

El objetivo principal de este video es en la comprensión del usuario sobre qué es una variable y, mediante consejos, tenga una buena práctica desde un inicio en la declaración de variables.

Otro objetivo del vídeo es que el espectador sepa identificar los diferentes tipos de datos que hay, además de conocer sus principales características y la sintaxis de cada tipo de dato, pudiendo realizar conversiones entre los diferentes tipos de datos.

### Descripción del vídeo

Curso de programación en Python

Vídeo 3: Conceptos básicos de las variables, palabras reservadas y los tipos de datos.

* Qué es una variable, reglas y consejos para su uso (estándares camelCase y snake\_case).
* Operador de asignación
* Palabras reservadas en Python.
* Tipo bool: uso y valores que pueden adquirir: True y False.
* Tipo int: definición, uso de la memoria y nomenclatura.
* Tipo float: definición, notación científica y función round().
* Tipo complex: qué es y manejo de estos datos.
* Tipo str: formas de crear una cadena de texto.
* Conversión de tipos: forma explícita e implícita para cambiar el tipo de dato de una expresión.

## Vídeo 4: Tipos de operadores en Python

### Descripción general

En el vídeo (Figura 3.4), se da una explicación sobre qué son los operadores y cuáles son los tipos de operadores disponibles en Python. De cada operador diferente se habla de los tipos de datos con los que se pueden operar, el resultado que se obtiene como resultado de una operación y se aporta información de los diversos operadores dentro de cada tipo de operador. También, se aporta una descripción de cada operador junto con su símbolo y un ejemplo de su uso para entender de manera práctica su funcionamiento.

*(Figura 3.4)*

### Objetivos

Como objetivo de este vídeo se establece que el espectador entienda qué es un operando y cómo se utiliza. También, el espectador sabrá cuales son los tipos de operadores que hay, además de saber qué función tiene cada uno de ellos, cómo se utilizan y el resultado que se obtiene en una operación.

### Descripción del vídeo

## Vídeo 5: Manipulación de cadenas de texto

### Descripción general

En el vídeo (Figura 3.5), se da una explicación profundizando en las cadenas de texto. En concreto se explican las secuencias de escape, que son una herramienta fundamental para imprimir por pantalla. También se explican las expresiones literales, que es una herramienta complementaria a las secuencias de escape. Como no podía faltar, se explican las funciones input y print, utilizadas para interactuar con el usuario.

La explicación de la posición de los caracteres es un aspecto importante del vídeo debido a que también es utilizado en la extracción de segmentos de las cadenas de texto, aspecto explicado también en el vídeo.

También se explican las cadenas de texto f-string, utilizadas para formatear variables permitiendo incluirlas en cadenas de texto.

Finalmente, se muestran algunas funciones útiles para la gestión de cadenas como la función para obtener la longitud de una cadena o la función para poner en minúscula la cadena de texto.

*(Figura 3.5)*

### Objetivos

El objetivo de este vídeo es que el espectador amplie su conocimiento en el manejo de las cadenas de texto porque es algo esencial en cualquier programa. El telespectador puede aprender a pedir información por pantalla y conocer las opciones disponibles para imprimir datos por pantalla para dar información al usuario. También aprenderá el manejo de una cadena con las posiciones de sus caracteres. Otro objetivo es que el espectador entienda que existe una amplia gama de funciones disponibles para tratar cadenas en sus programas.

### Descripción del vídeo

## Vídeo 6: Ejercicio práctico: Cálculo de la intensidad y la potencia

### Descripción general

En el vídeo (Figura 3.6), se expone un ejercicio práctico para aplicar la teoría vista en los vídeos anteriores. Se introduce al usuario el uso de comentarios, junto con la declaración y asignación de variables. Además, se realizan algunas operaciones a las variables y comprobaciones a los tipos de datos utilizando funciones de Python.

Por otro lado, se utilizan varias funciones para mostrar por pantalla información del programa, formateando variables para facilitar la lectura del usuario. Además, se añade un apartado para el manejo de cadenas de texto.

*(Figura 3.6)*

### Objetivos

El objetivo principal de este vídeo es que el espectador pueda relacionar la teoría vista anteriormente con un programa funcional. Siendo por tanto, una forma de afianzar los conocimientos o terminar con el aprendizaje de manera práctica y motivar al espectador a realizar sus propios programas o modificaciones a un programa hecho para ampliar sus conocimientos.

### Descripción del vídeo

## Vídeo 7: Control de flujo: Sentencias condicionales

### Descripción general

En este vídeo (Figura 3.7), se explican de manera individual todas las sentencias condicionales. Explicando su funcionamiento y estructura, empezando por la condicional if, seguida de la sentencia if-else, a continuación, la estructura if-elif-else y las sentencias anidadas. Por último, se explican la sentencia match-case. Para asegurar el entendimiento de la información expuesta, se añaden ejemplos para cada una de las sentencias y se explican de manera detallada.

*(Figura 3.7)*

### Objetivos

El objetivo del vídeo es que el espectador sepa identificar las opciones que hay para controlar el flujo del programa mediante sentencias condicionales, conozca el funcionamiento de cada una de ellas, cómo funcionan y la función que cumplen dentro de un programa.

### Descripción del vídeo

## Tiempo total

En este apartado se añade el tiempo de docencia disponible tras la elaboración de todos los vídeos realizados. Para ello, se muestra a continuación una tabla (Tabla 3.1) que contiene tanto el título de cada vídeo como en tiempo invertido en cada uno. Para tener una visión global del trabajo realizado se añade una suma de todos los vídeos.

|  |  |  |
| --- | --- | --- |
| Orden | Título y URL del vídeo | Duración (min) |
| 1 | Introducción a la programación con Python | 6:11 |
| 2 |  |  |
| 3 |  |  |
| Duración total | |  |
| Duración promedio | |  |

Tabla 3.1: Desglose de la duración total del contenido.