

TEMARIO

Unidad 1 - Introducción

- ¿Qué es Python?
- Ventajas y desventajas
- Ecosistema Python y Comunidad – Librerías extendidas
- Descarga – Opensource
- Instalación, configuración y hardware necesario
- Errores sintácticos y lógicos, localización en pantalla y correcciones
- Importancia del versionado
- GIT Colaborativo – Pair Programming
 - Introducción a **GIT**
 - Creando un repositorio, **clonar**, **branches**
 - Borrar, guardar (**STASH**), recuperar (**POP**)

Unidad 2 – Software

Características de Python

- Software libre
- Alto nivel
- Multiparadigma
- Portable
- Programación Secuencial y Orientada a Objetos
- Multiplataforma
- Interpretado
- Tipado dinámico
- Estructura (TAB)

Entorno de Desarrollo Intérprete – IDEs

- Elección según el propósito del trabajo: **PyCharm**, **PyDev**, **Atom**, **Spyder**, **PyScripter**, **Eclipse**, **IPython**.
- Entornos especiales: **Anaconda** (Data Science Platform), **Jupyter** (Notebooks).
- Consola, pantalla gráfica y entorno
- Salida de datos por *pantalla*
 - Sentencias: **print ()**
- Ingreso de datos por *teclado*
- Sentencias: **input ()**

Unidad 3 – Estructura y primeros Trabajos con datos

VARIABLES, CONSTANTES

- Flujo de datos, estructura, linealidad, condicionales, bucles

- Estructuras condicionales simples, compuestas y anidadas
- Sentencias: if, elif, else, :
- Estructuras repetitivas
- Sentencias: for, range, while, else :
- Estructuras modificaciones
- Sentencias: break, continue, pass
- Operadores:
- Comparación: ==, <, <=, >, >=, !=
- Lógicos: and, not, or
- Aritméticos: +, -, *, **, /, //, %, (ver librería math)
- Asignación: =, +=, -=, *=, **, /=, //=, %=
- Especiales: is, is not, in, not in

Espacios, nombres, ámbitos, objetos

- Variables y constantes- Tipos
- Procesamiento de cadenas

Listas [variables]

- Índices
- Recorrer listas
- Sentencias: append(), clear(), copy(), count(), extend(), index(), insert(), pop(), remove(), reverse(), sort(), etc.

Tuplas (Constantes)

- Índices
- Recorrer Tuplas
- Sentencias: index(), count(), etc.

Diccionarios {clave:valor asociado}

- Funcionamiento de diccionarios
- Sentencias: clear(), copy(), fromkeys(), get(), items(), keys(), pop(), popitem(), reverse(), setdefault(), update(), values(), etc.
- Sets y otros

Unidad 4 – Funciones

- Iterar: ejecución repetida de un conjunto de sentencias
Sentencias: def (): return
- Parámetros de entrada de datos
- Retorno de datos a la salida
- Return de listas
- Parámetros con valor por defecto (=val;*,**)

Unidad 5 – Módulos y Librerías

- Uso de métodos y funciones de un archivo externo

Sentencias: Import, from

- Generar un modulo
- Uso de librerías
- Generar archivos, leerlos, escribirlos (TXT-plano/ Binario)
JSON (Javascript)
Pickle (Python)
- Instalación de librerías, ecosistema,
Métodos: pip, conda,
Download e instalación MSI, Linuc, etc

Unidad 6 – Clases

Sistema para empaquetar atributos de datos y funcionalidad métodos para instanciar

Sentencias: class ():, self

- Objetos clases
- Objetos instancias
- Objetos métodos
- Herencias, herencias múltiples

Unidad 7 – Entorno visual - WEB Django

Pantallas graficas - Libreria: tkinter, numpy y matplotlib

- Pantallas, Frames, Labels, bottons,etc
- Ubicación de elementos, colores, formatos, tamaños, etc.
- Ingreso de daros desde pantalla (get)
- Salida de datos por pantalla
- Acciones de botones para llamar a funciones
- Graficas de funciones matematicas y otros datos.(series, tortas, 3d, etc.)
- Python y “Django” e la web framework

Ejemplos de uso intensivo de Django (Instagram, Pinterest, Mozilla, National Geographic, etc.)

Unidad 8 – Bases de Datos locales y en la nube

Tipo: SQL (Mysql) y NoSQL (Mongo)

Librería: mysql.connector

Librería: pymongo

- Python y “Big Data”
- Conexión
- cursor(), .execute(), .close
- Crear Bases, tablas, columnas
- Tipos de datos, caracteres, numéricos, fecha-hora
- Buscar, insertar, actualizar, borrar, seleccionar, elementos desde y hacia una base de datos
- Where, from. %like%
- Firebase, Google Cloud IoT - u otro hub para OIT AWS (Amazon) Azure (MSoft)

Unidad 9 – Fechas, Horas

- Modulo time, datetime
- Manejo de fechas y horas

- Uso en aplicaciones web, base de datos, multiplataforma, etc.

Unidad 10 – Internet Of Things – IOT

- Programación de eventos - Timed Event
Librería: Scheduler
- Módulo sched / Scheduler
- Declaración de programadores
- Llamado a funciones como eventos
- Programar eventos y poner en marcha el programador
- Programación de eventos considerando prioridades
- Cancelación de eventos
- Python y Internet Of Things – IOT
- Python y MicroControladores (un matrimonio perfecto)
Librería: Zerynth
- Ejemplos de uso intensivo de Raspberry Pi y NodeMCU (ESP8266)
IOT Con BBDD, Python y Android
- Python y Amazon - AWS IoT
- Protocolo MQTT

Cronograma de Clases

Clase 1	Unidad 1
Clase 2	Unidad 2 – 3
Clase 3	Unidad 4
Clase 4	Unidad 5
Clase 5	Unidad 6
Clase 6	Unidad 7
Clase 7	Unidad 8 – 9
Clase 8	Coding Dojo
Clase 9	Unidad 10
Clase 10	Examen