

**CENTRO DE ESTUDIOS OCUPACIONALES - CEO
DIPLOMADO EN DESARROLLO DE SOFTWARE
Y DIPLOMADO EN INTELIGENCIA ARTIFICIAL
CONTENIDO TEMÁTICO
PROGRAMACIÓN BÁSICA CON PYTHON**

Objetivo General: Al finalizar esta clase, los estudiantes serán capaces de:

- Comprender y aplicar los fundamentos básicos de programación en Python.
- Utilizar entornos simples como Replit y Google Colab con confianza.
- Resolver problemas sencillos mediante estructuras de control y lógica de programación.
- Desarrollar un chatbot simple basado en reglas, integrando todos los conceptos aprendidos.

Duración: 16 horas divididas en 4 módulos de 4 horas cada uno.

MÓDULO 1: BIENVENIDA AL MUNDO DE PYTHON

1.1. Hola, programador/a

- ¿Por qué Python?: Breve historia, ventajas y aplicaciones (IA, desarrollo web, ciencia de datos).
- Mentalidad de crecimiento: "No se trata de saberlo todo, sino de aprender a resolverlo".

1.2. Primeros pasos en Python

- ¿Qué es la sintaxis?
- Hola mundo: ``print("¡Hola desde Python!")``
- Comentarios: ``# Esto es un comentario``
- Errores comunes y cómo leerlos.
- Mini reto: Imprimir un mensaje personalizado con tu nombre y tu objetivo en el diplomado.

1.3. Variables y tipos de datos

- ¿Qué es una variable?
- Tipos básicos: ``int``, ``float``, ``str``, ``bool``.
- Operaciones básicas: suma, resta, concatenación.
- Buenas prácticas: nombres de variables claros y legibles.

- Ejercicio práctico: Crea un perfil de usuario con nombre, edad, ciudad y si está inscrito.

1.4. Entornos de programación: Replit y Google Colab

- Crear cuenta en Replit.
- Crear un notebook en Google Colab.
- Comparar ambos entornos: ¿cuándo usar cada uno?
- Guardar, compartir y ejecutar código.
- Mini proyecto: Sube tu “perfil de usuario” a Replit y compártelo con un compañero/a.

1.5. Cierre y reflexión

- ¿Qué fue lo más sorprendente hoy?
- ¿Qué dudas te quedaron?
- Tarea: Jugar con variables: crea un pequeño “diario de programación” en Replit.

MÓDULO 2: TOMANDO DECISIONES CON PYTHON

2.1. ¿Qué es la lógica condicional?

- Analogía: “Si llueve, llevo paraguas” → `if llueve: llevar_paraguas()`
- Operadores de comparación: `==`, `!=`, `<`, `>`, `<=`, `>=`.
- Operadores lógicos: `and`, `or`, `not`.

2.2. Condicionales en acción

- Sintaxis de `if`, `elif`, `else`.
- Anidamiento de condiciones.
- Validación de datos (ej: edad válida, opción de menú).
- Ejercicio práctico: Crea un programa que diga si una persona es menor, adulta o mayor de edad.

2.3. Proyecto: Adivina el número

- Uso de `input()` y conversión de tipos (`int()`).
- Generar un número aleatorio con `random.randint()`.
- Bucle simple con `while` (introducción suave).
- Dar pistas: “muy alto”, “muy bajo”.
- Meta: El usuario tiene 5 intentos para adivinar un número entre 1 y 50.

2.4. Retroalimentación y mejora

- Compartir versiones del juego en Replit.
- ¿Cómo podrías mejorarlo? (más intentos, puntuación, etc.)
- Introducción al concepto de “iteración” para el próximo bloque.

MÓDULO 3: REPETICIÓN Y ESTRUCTURA

3.1. ¿Por qué repetir? Introducción a los bucles

- Ejemplo cotidiano: “Repetir hasta que el vaso esté lleno”.
- Diferencia entre ``for`` (sabes cuántas veces) y ``while`` (mientras se cumpla).

3.2. Listas y bucles ``for``

- Crear listas: `nombres = ["Ana", "Luis", "María"]``
- Iterar con ``for``: ``for nombre in nombres:``
- Índices, ``len()``, y métodos básicos: ``append()``, ``remove()``.
- Ejercicio: Crea una lista de tareas y muéstralas numeradas.

3.3. Bucle ``while`` y validación

- Uso de banderas y contadores.
- Evitar bucles infinitos.
- Menú interactivo: “1. Sumar, 2. Restar, 3. Salir”.
- Mini proyecto: Calculadora básica con menú que se repite hasta que el usuario elija salir.

3.4. Integración: Mejora tu “Adivina el número”

- Agregar puntuación.
- Permitir jugar varias veces.
- Guardar historial de partidas en una lista.

MÓDULO 4: CONSTRUYE TU PRIMER CHATBOT

4.1. ¿Qué es un chatbot?

- Ejemplos reales: asistentes virtuales, soporte al cliente.
- Diferencia entre reglas y aprendizaje automático (breve introducción para IA).
- Enfoque de hoy: “chatbot basado en reglas”

4.2. Diseño del chatbot

- Planificación: ¿Qué preguntas responderá?
- Flujo de conversación: bienvenida → preguntas → despedida.
- Uso de ``if-elif-else`` para detectar palabras clave.
- Ejemplo:
 - Usuario: “Hola” → Bot: “¡Hola! ¿En qué puedo ayudarte?”
 - Usuario: “¿Cómo estás?” → Bot: “¡Genial! ¿Y tú?”

4.3. Programación del chatbot

- Usar ``input()`` y ``lower()`` para normalizar texto.
- Detectar intenciones con ``in``: ``if "nombre" in mensaje:``

- Agregar respuestas aleatorias para hacerlo más natural.
- Incluir un modo “salir”.
- Reto adicional:
 - Guardar el historial de la conversación en una lista.
 - Mostrar el historial al final.

4.4. Presentaciones y cierre

- Cada estudiante (o pareja) muestra su chatbot en Replit.
- Comentarios positivos y sugerencias entre pares.
- Reflexión final:
 - ¿Qué aprendiste?
 - ¿Cómo aplicarás esto en tu diplomado?
 - ¡Felicitaciones! Has dado tu primer gran paso en programación.

Recursos Adicionales

- Replit: <https://replit.com>
- Google Colab:
<https://colab.research.google.com>
- Documentación oficial de Python (en español):
<https://docs.python.org/es/>
- Juego interactivo para practicar:
<https://www.codecombat.com>

Consejos del Docente

- Fomenta el error como parte del aprendizaje: “Si no fallas, no estás intentando lo suficiente”.
- Celebra los pequeños logros: cada línea de código que funciona es un triunfo.
- Conecta con sus metas: “Este chatbot es el primer paso hacia un asistente de IA o una app completa”.