



Resumen

Python es un lenguaje de programación versátil, intuitivo y eficaz. Utilizado por desarrolladores y analistas de datos, Python es hoy en día uno de los lenguajes de programación más populares en el mundo y con mayor demanda en el mercado laboral. Al llevar este curso, el estudiante aprenderá los conceptos fundamentales del software a través de nuestra metodología práctica basada en proyectos.

El curso consta de 16 horas totales a lo largo de 8 semanas, las cuales llevarán al estudiante a aprender desde conceptos básicos de programación, hasta métodos sofisticados para resolver cualquier tipo de problemas en Python. Se cubrirán temas como **Tipos de Datos, Condicionales, Lógica, For/While Loops, Funciones, Programación Orientada a Objetos**, y muchos más. Además, durante el curso, se resolverán múltiples ejercicios y proyectos participativos que ayudarán a reforzar los conocimientos adquiridos. Por último, al finalizar el curso, el estudiante deberá aprobar un examen teórico en el que tendrá que responder cinco preguntas de opción múltiple, con el fin de evaluar el grado de aprendizaje.

¿Para quién es este curso?

Este curso está diseñado para estudiantes que no tienen experiencia en programación, o estudiantes que ya conocen otros lenguajes, pero quieren adentrarse en el mundo de Python. NO es necesario ningún tipo de experiencia ni conocimiento previo.

Objetivos

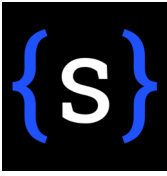
1. Desarrollar una base de programación sólida.
2. Adquirir habilidades para crear juegos interactivos y resolver problemas complejos en Python.
3. Crear una conexión



TEMARIO

- **INTRODUCCIÓN A PYTHON**
- **OBJETOS Y TIPOS DE DATOS**
 - Introducción a los tipos de datos
 - Variables en Python
 - Strings
 - Números en Python
 - Listas
 - Diccionarios
 - Booleans en Python y operaciones lógicas (and/or)
- **IF, ELIF Y ELSE STATEMENTS**
 - Tests condicionales
 - Igualdad y desigualdad
 - If statements básicos
 - Encadenando if-elif-else statements
- **FOR LOOPS**
 - Haciendo un loop a una lista
 - Evitar errores de indentación
 - Creación de listas numéricas con loops
- **WHILE LOOPS E INPUT DEL USUARIO**
 - Función input()
 - Introducción al while loop
- **FUNCIONES**
 - Definición y anatomía de una función
 - Argumentos y parámetros

| Proyectos: |
|--|
| Nuestra metodología se basa en aprender haciendo. Por ende, los temas antes descritos serán reforzados mediante ejercicios prácticos y diversos proyectos, ya sean durante la clase o como tarea en casa. Entre estos, se encuentran: |
| 1) Rock, Paper, Scissors: Al igual que el juego real, el estudiante será capaz de programar un juego de piedra, papel o tijeras en Python utilizando if conditionals, la función input() y while loops. |
| 2) Hangman: El clásico juego del ahorcado. El estudiante programará un juego del ahorcado de manera visual, es decir, podrá ver el desarrollo del juego a medida que se adivinan las letras. El objetivo de este proyecto es reforzar los conceptos de listas, for loops y lógica. |
| 3) Number Guessing: Un proyecto en donde se practican la mayoría de los conceptos del curso. Se trata un sencillo programa en donde la computadora, utilizando el random package, genera un número aleatorio, y el jugador lo tiene que adivinar. |



INSTRUCTOR



Día 1 - Bienvenida

- Introducción a VSCode y Python
- Variables y funciones
- Comentarios
- Tipos de datos
 - Definir serie de variables
- Formateo de strings
- Mad libs
- Tarea: hacer una historia usando al menos 5 inputs

2. Rock, paper, scissors (input, conditionals, while loop, random) (codesters)

- Tarea: terminar y agregar
- Condicional

3. Rock, paper, scissors

4. Number guesser

5. Number guesser

6. Game inventory

7. Game inventory

8. Ejercicios

9. Hangman

10. Hangman

11. ...Hangman

12. Turtle

13. Turtle

14. Penguin Rush (codesters)

15. Penguin Rush (codesters)

16. Penguin Rush (codesters)

17. Penguin Rush (codesters)

18. Ejercicios

19. Brick Breaker (codesters)

20. Brick Breaker (codesters)

21. Brick Breaker (codesters)

22. Brick Breaker (codesters)

23. Sort and search algorithms

24. Sort and search algorithms

25. Sort and search algorithms

26. Sort and search algorithms

27. Map based text adventure

28. Map based text adventure

29. Map based text adventure



- 30. Map based text adventure
- 31. Map based text adventure
- 32. Examen

32 horas, 16 semanas



INFORMACIÓN GENERAL

-Inicio: XYZ
-Duración: XYZ
-Horario: XYZ de 00:00pm a 00:00pm



INVERSIÓN:
\$XYZ