



PRINCIPIOS DE PROGRAMACIÓN CON PHYTON

Nivel 1



CERTIFICACIÓN ZTUDIA EN PRINCIPIOS DE PROGRAMACIÓN CON PYTHON NIVEL 1.

Al finalizar el curso, el Ztudiante será capaz de aplicar un conocimiento amplio y estructurado sobre los principios esenciales de la programación con Python para representar, operar, controlar y modularizar soluciones algorítmicas, lo que le permitirá construir programas funcionales, legibles y depurables que resuelvan problemas computacionales en contextos académicos y profesionales.

1. Variables y Asignación de Datos Primitivos

- 1.1. Declarar una variable de tipo Entero
- 1.2. Declarar una variable de tipo Punto Flotante
- 1.3. Declarar una variable de tipo Cadena
- 1.4. Declarar una variable de tipo Booleano
- 1.5. Asignar el valor Nulo a una variable
- 1.6. Modificar el valor original de una variable numérica mediante reasignación
- 1.7. Convertir un dato de tipo texto a tipo Entero
- 1.8. Convertir un dato de tipo texto a tipo Punto Flotante
- 1.9. Convertir un dato numérico a tipo Cadena

ACTIVIDAD 1

2. Operaciones Aritméticas Esenciales

- 2.1. Calcular la suma de dos variables numéricas
- 2.2. Calcular la resta de dos variables numéricas
- 2.3. Calcular la multiplicación de dos variables numéricas
- 2.4. Calcular la división con resultado de punto flotante
- 2.5. Calcular la división entera omitiendo el residuo
- 2.6. Elevar un número a una potencia específica
- 2.7. Obtener el módulo de una división para hallar el residuo
- 2.8. Crear expresiones aritméticas combinadas respetando la jerarquía de operadores

ACTIVIDAD 2

3. Expresiones Lógicas y Relacionales

- 3.1. Comparar la igualdad de dos valores
- 3.2. Comparar la desigualdad de dos valores
- 3.3. Evaluar condiciones de magnitud mayor o mayor-igual
- 3.4. Evaluar condiciones de magnitud menor o menor-igual
- 3.5. Evaluar múltiples condiciones obligatorias mediante el operador lógico Y
- 3.6. Evaluar múltiples condiciones alternativas mediante el operador lógico O
- 3.7. Invertir el resultado de una condición mediante el operador lógico NO



ACTIVIDAD 3

4. Estructuras Selectivas Simples y Dobles

- 4.1. Utilizar la estructura selectiva simple
- 4.2. Utilizar la estructura selectiva doble
- 4.3. Validar condiciones de texto dentro de una estructura de control
- 4.4. Validar condiciones numéricas dentro de una estructura de control

ACTIVIDAD 4

5. Estructuras Selectivas Múltiples y Anidadas

- 5.1. Utilizar la estructura selectiva múltiple
- 5.2. Implementar estructuras selectivas anidadas
- 5.3. Asignar valores a una variable basados en una estructura selectiva múltiple
- 5.4. Utilizar la asignación condicional directa mediante el operador ternario

ACTIVIDAD 5

6. Estructuras Iterativas con Rango

- 6.1. Crear un bucle utilizando un rango numérico simple
- 6.2. Crear un bucle especificando un valor de inicio y fin
- 6.3. Crear un bucle con un incremento o paso personalizado
- 6.4. Crear un bucle en orden descendente utilizando rangos inversos

ACTIVIDAD 6

7. Estructuras Iterativas por Condición

- 7.1. Crear un bucle controlado por una condición booleana
- 7.2. Utilizar una variable contador para finalizar un bucle
- 7.3. Emular la estructura Hacer-Mientras mediante un bucle infinito y una condición de ruptura
- 7.4. Interrumpir la ejecución de un bucle de forma inmediata
- 7.5. Omitir el resto de una iteración y saltar a la siguiente
- 7.6. Implementar bucles iterativos anidados

ACTIVIDAD 7

8. Manipulación Esencial de Colecciones

- 8.1. Declarar e inicializar una lista vacía
- 8.2. Declarar una lista con elementos predefinidos
- 8.3. Acceder a un elemento específico de una lista mediante su índice positivo
- 8.4. Acceder a un elemento de una lista mediante un índice de posición negativa
- 8.5. Modificar el valor de un elemento dentro de una lista por su índice
- 8.6. Agregar un nuevo elemento al final de una lista
- 8.7. Obtener la cantidad total de elementos de una lista



ACTIVIDAD 8

9. Operaciones Avanzadas con Colecciones y Matrices

- 9.1. Recorrer los elementos de una lista utilizando un bucle
- 9.2. Calcular la suma acumulada de los elementos numéricos de una lista
- 9.3. Declarar e inicializar una matriz
- 9.4. Acceder a un valor específico dentro de una matriz mediante coordenadas de doble índice
- 9.5. Recorrer los elementos de una matriz utilizando bucles anidados

ACTIVIDAD 9

10. Estructuras Clave-Valor

- 10.1. Declarar e inicializar un diccionario con pares clave-valor
- 10.2. Acceder al valor de un diccionario mediante su clave
- 10.3. Modificar el valor de una clave existente en un diccionario
- 10.4. Agregar un nuevo par clave-valor a un diccionario
- 10.5. Recorrer las claves y valores de un diccionario con un bucle

ACTIVIDAD 10

11. Diseño y Uso de Funciones

- 11.1. Crear una función personalizada
- 11.2. Definir una función que reciba un parámetro obligatorio
- 11.3. Definir una función que reciba múltiples parámetros
- 11.4. Devolver un resultado explícito desde una función
- 11.5. Invocar una función creada pasándole argumentos fijos
- 11.6. Asignar el resultado de la invocación de una función a una variable

ACTIVIDAD 11

12. Algoritmia Estructural de Base

- 12.1. Implementar la lógica de una búsqueda lineal dentro de una lista
- 12.2. Implementar la lógica del ordenamiento de burbuja
- 12.3. Crear una función que se invoque a sí misma de forma recursiva

ACTIVIDAD 12

13. Prevención de Errores en Tiempo de Ejecución

- 13.1. Interceptar errores potenciales utilizando bloques de control
- 13.2. Capturar un error específico de división por cero
- 13.3. Capturar un error específico de tipo de dato incompatible
- 13.4. Capturar un error de índice fuera de rango en una lista
- 13.5. Capturar un error de clave no encontrada en un diccionario

ACTIVIDAD 13