

<b>Cargo:</b>	Docente		
<b>Nombre:</b>	M.Sc. David Fabián Cevallos Salas		
<b>Asignatura:</b>	Programación Web		
<b>Carrera:</b>	Marketing Digital y Comercio Electrónico	<b>Nivel:</b>	Cuarto nivel
<b>Estudiante:</b>			

## ACTIVIDAD PRÁCTICO EXPERIMENTAL EN EL ENTORNO ACADÉMICO

### Implementación de bifurcaciones empleando Python.

#### 1. Objetivos

- Aprender a utilizar estructuras de bifurcación en Python, como la instrucción if-else.
- Practicar la implementación de condiciones y decisiones en programas Python.
- Familiarizarse con el flujo de control en programas Python mediante bifurcaciones.

#### 2. Antecedentes/Escenario

En programación, las bifurcaciones son estructuras que permiten tomar decisiones en función de ciertas condiciones. En Python, la instrucción if-else es una de las formas más comunes de implementar bifurcaciones.

#### 3. Recursos necesarios

- Computadora con Python instalado.
- Entorno de desarrollo integrado (IDE) o editor de texto para escribir y ejecutar programas Python.

#### 4. Planteamiento del problema

Se solicita a los estudiantes implementar bifurcaciones en Python para resolver el siguiente problema: Dado un número ingresado por el usuario, el programa debe determinar si el número es positivo, negativo o cero, e imprimir el resultado.

#### 5. Pasos por realizar

##### 5.1. Inicio del Programa:

- Abrir un entorno de desarrollo integrado (IDE) o un editor de texto para escribir el programa Python. De preferencia trabaje con Google Colab.

##### 5.2. Entrada de Datos:



- Solicitar al usuario que ingrese un número utilizando la función `input()` y almacenarlo en una variable.

### 5.3. Bifurcaciones con if-else:

- Utilizar la estructura if-else para verificar la condición del número ingresado.
- Si el número es mayor que cero, imprimir "El número es positivo".
- Si el número es menor que cero, imprimir "El número es negativo".
- Si el número es igual a cero, imprimir "El número es cero".

### 5.4. Ejecución del Programa:

- Ejecutar el programa y probar diferentes números para verificar Inicio del Programa:
- Analizar el funcionamiento de las bifurcaciones.

## 6. Desarrollo

### 6.1. Implementar el siguiente código fuente.

```
python

# Solicitar al usuario que ingrese un número
numero = float(input("Ingrese un número: "))

# Verificar la condición del número e imprimir el resultado
if numero > 0:
    print("El número es positivo")
elif numero < 0:
    print("El número es negativo")
else:
    print("El número es cero")
```

### 6.2. Ejecutarlo en el entorno de trabajo Google Colab o entorno Python de su preferencia.

