

Actividad Autonomía U2

Algoritmos en PSEINT Y PYTHON

Autor: Pedro Daniel Risco Santistevan

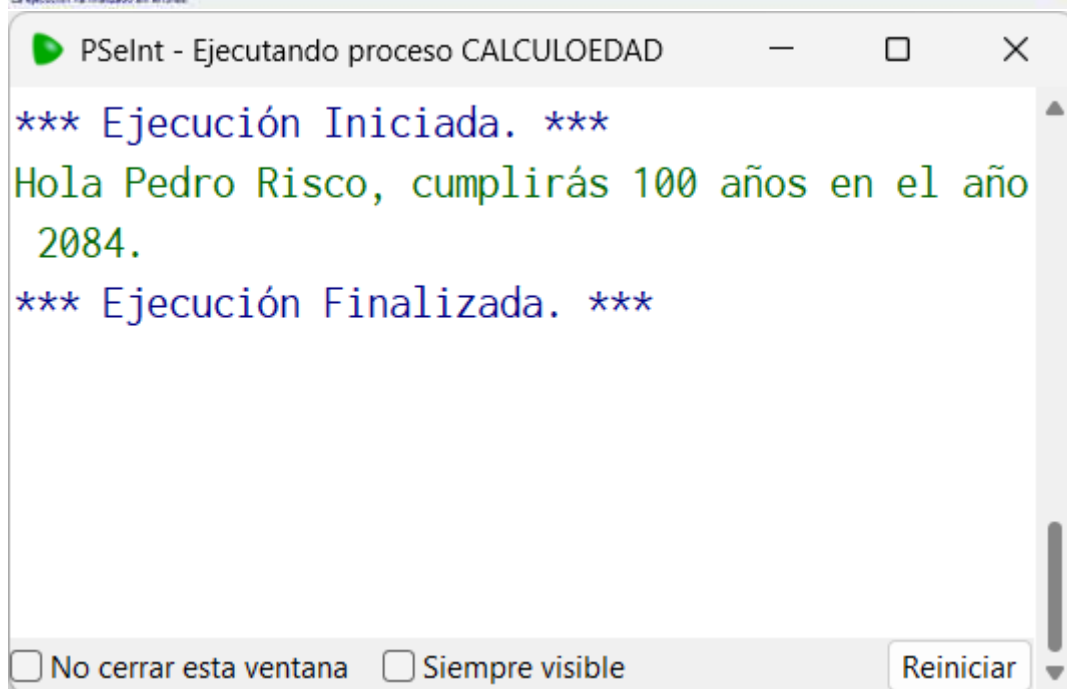
Docente: Jose Alberto Torres Ortega



```
1 Proceso CalculoEdad
2   Definir nombre Como Cadena
3   Definir edad, año_actual, año_100 Como Entero
4
5   // Datos proporcionados
6   nombre ← "Pedro Risco"
7   edad ← 48
8   año_actual ← 2024
9
10  // Calcular en qué año se cumplirá 100 años
11  año_100 ← año_actual + (100 - edad)
12
13  // Mostrar resultado
14  Escribir "Hola ", nombre, ", cumplirás 100 años en el año ", año_100, ".*"
15 FinProceso
16
17 |
```

Después clickeo en "Ejecutar" para iniciar la ejecución paso a paso y leer en este panel los detalles de cada acción que realiza el intérprete.

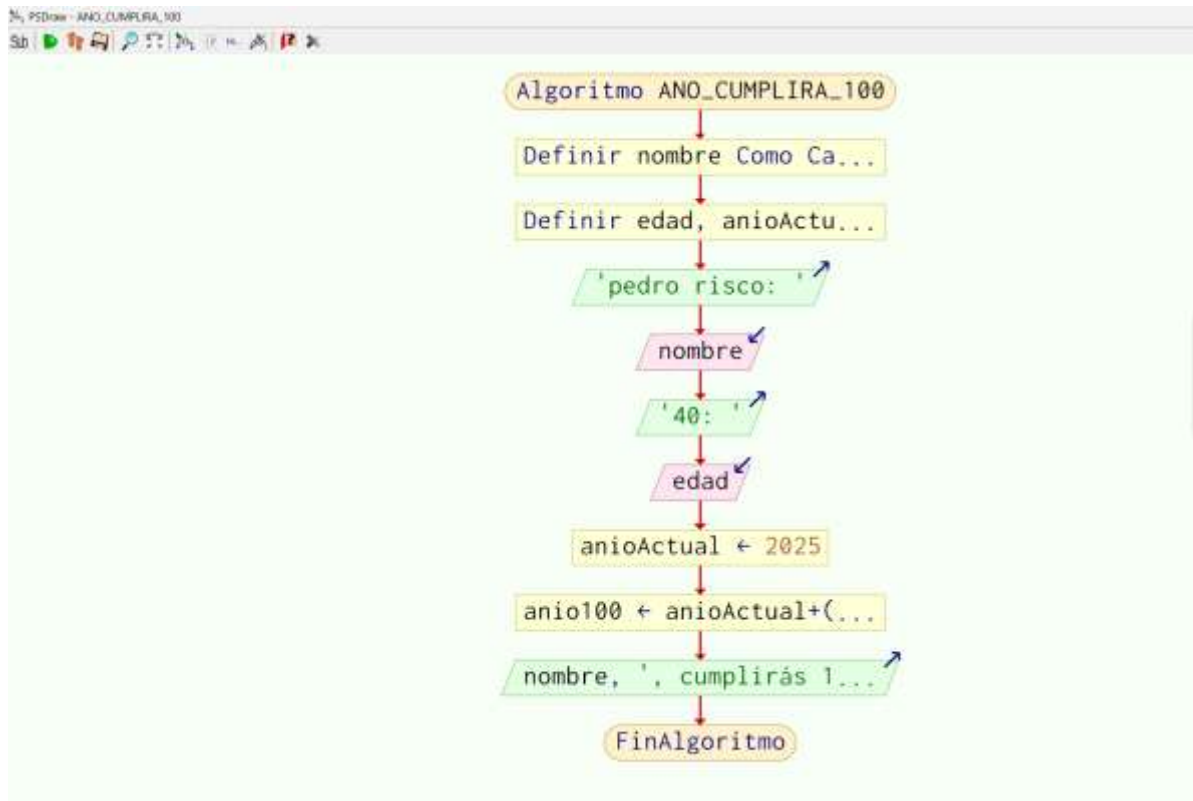
La ejecución ha finalizado sin errores.



PSeInt - Ejecutando proceso CALCULOEDAD

```
*** Ejecución Iniciada. ***
Hola Pedro Risco, cumplirás 100 años en el año
2084.
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar



Explicación del Código

1. **Se definen las variables :**
 - o nombre como **cadena** .
 - o edad, año_actual, año_100 como **enteros** .
2. **Se asignan valores iniciales :**
 - o nombre = "Pedro Risco"
 - o edad = 40
 - o año_actual = 2024
3. **Se realiza el cálculo :**
 - o $\text{año_100} = \text{año_actual} + (100 - \text{edad})$.
4. **Se muestra el resultado en pantalla.**

```

C:\Program Files\Python\Python310\
Python 3.10.11 [tags/v3.10.11:7d4cc5a, Apr  5 2023, 08:38:17] [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> # Definimos las variables
>>> nombre = "Pedro Risco"
>>> edad = 88
>>> año_actual = 2024
>>>
>>> # Cálculo del año en que cumplirá 100 años
>>> año_100 = año_actual + (100 - edad)
>>>
>>> # Mostramos el resultado
>>> print(f"¡Hola {nombre}! Cumplirás 100 años en el año {año_100}!")
¡Hola Pedro Risco, cumplirás 100 años en el año 2024.
>>>
>>>

```

Explicación del Código

1. **Se definen las variables :**
 - o nombre = "Pedro Risco"(tipo **cadena**)
 - o edad = 40y año_actual = 2024(consejo**entero**).
2. **Se calcula el año en que cumplirá 100 años.:**
 - o año_100 = año_actual + (100 - edad).
3. **Se muestra el resultado**esprint().

```

1
2 Proceso Tipos_Datos
3
4 // Declaración de variables
5 Definir edad Como Entero;
6 Definir salario Como Real;
7 Definir nombre Como Cadena;
8
9 // Asignación de valores
10 edad + 40;
11 salario + 1000.00;
12 nombre + "Pedro Risco";
13
14 // Operaciones básicas
15 Definir suma, resta, multiplicacion, division Como Real;
16 suma + edad + salario;
17 resta + salario - edad;
18 multiplicacion + edad * PI;
19 division + salario / edad;
20
21 // Mostrar resultados
22 Escribir "Nombre: ", nombre;
23 Escribir "Edad: ", edad;
24 Escribir "Salario: ", salario;
25 Escribir "Suma: ", suma;
26 Escribir "Resta: ", resta;
27 Escribir "Multiplicación con PI: ", multiplicacion;
28 Escribir "División: ", division;
29 FinProceso
30

```



PSI01 - Ejecutando proceso TIPOS_DATOS

*** Ejecución Iniciada. ***

Nombre: Pedro Risco

Edad: 40

Salario: 1500.5

Suma: 1540.5

Resta: 1460.5

Multiplicación con PI: 125.6637061436

División: 37.5125

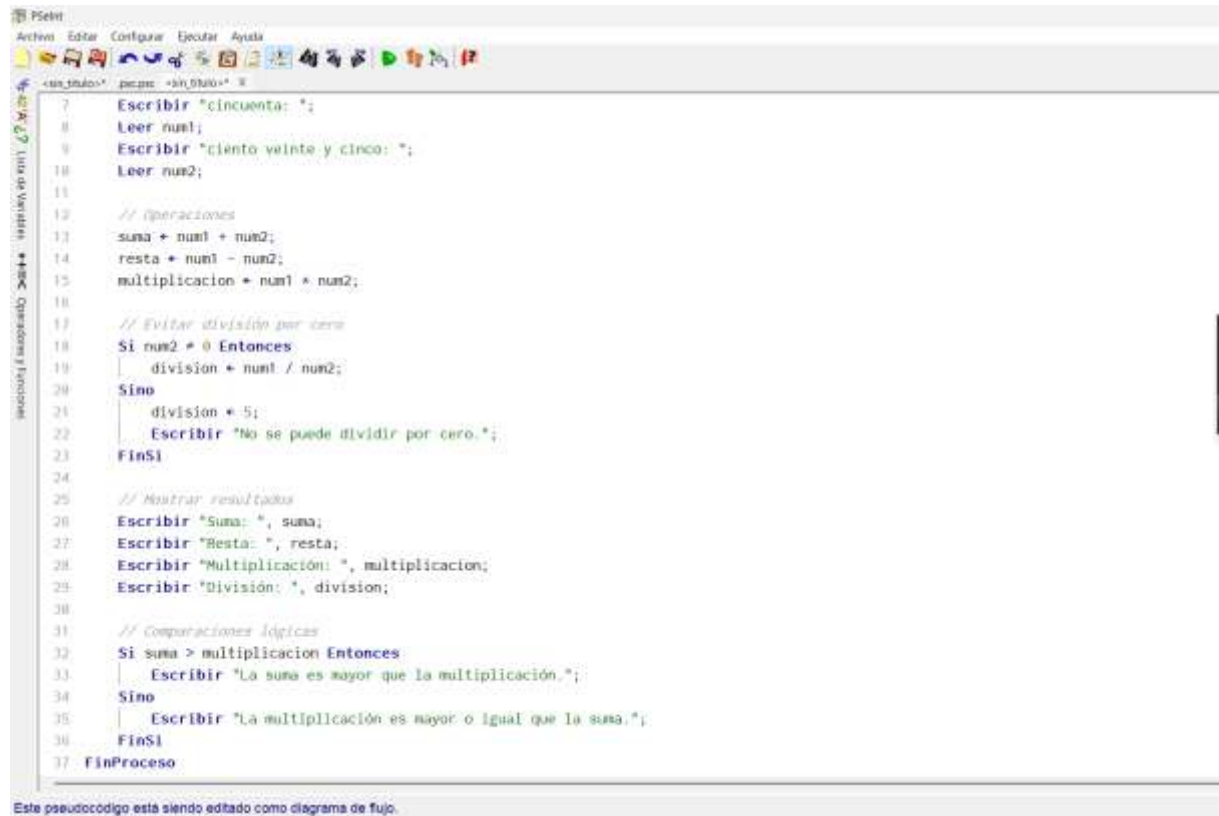
*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible

Operaciones Básicas y Expresiones Lógicas

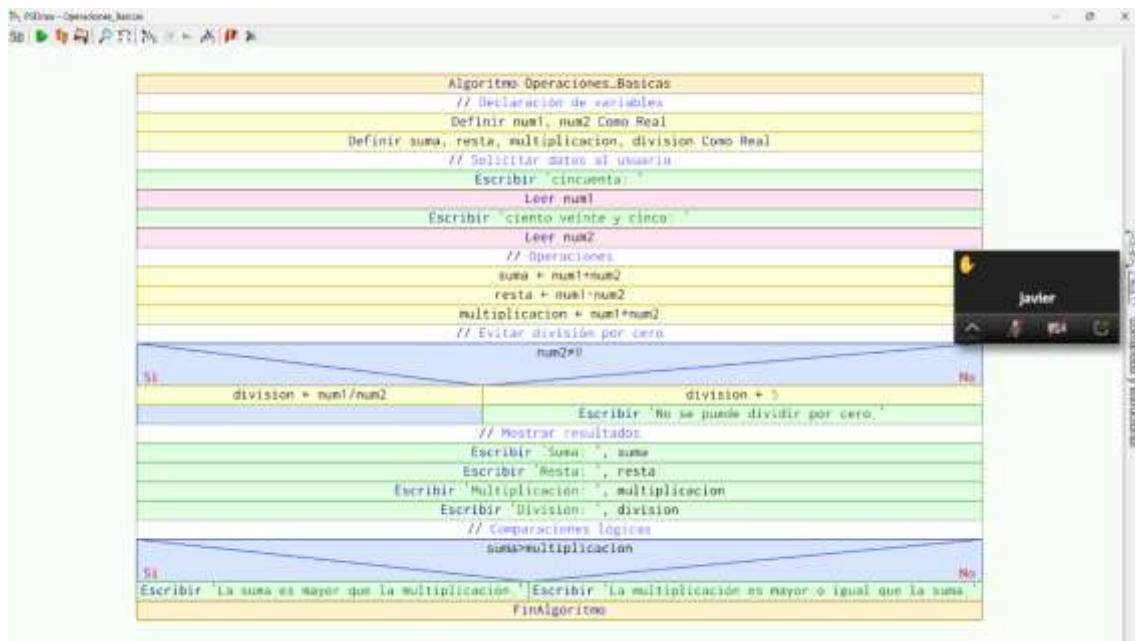
Aquí haremos un programa en PSeInt y Python que:

1. Pida dos números al usuario.
2. Realice las operaciones cuatro aritméticas básicas .
3. Utilice operadores lógicos para comparar los resultados y mostrar mensajes según la comparación



```
7  Escribir "cincuenta: ";
8  Leer num1;
9  Escribir "ciento veinte y cinco: ";
10 Leer num2;
11
12 // Operaciones
13 suma = num1 + num2;
14 resta = num1 - num2;
15 multiplicacion = num1 * num2;
16
17 // Evitar división por cero
18 Si num2 ≠ 0 Entonces
19     división = num1 / num2;
20 Sino
21     división = 5;
22     Escribir "No se puede dividir por cero.";
23 FinSi
24
25 // Mostrar resultados
26 Escribir "Suma: ", suma;
27 Escribir "Resta: ", resta;
28 Escribir "Multiplicación: ", multiplicacion;
29 Escribir "División: ", división;
30
31 // Comparaciones lógicas
32 Si suma > multiplicacion Entonces
33     Escribir "La suma es mayor que la multiplicación.";
34 Sino
35     Escribir "La multiplicación es mayor o igual que la suma.";
36 FinSi
37 FinProceso
```

Este pseudocódigo está siendo editado como diagrama de flujo.



Informe sobre Implementación de Algoritmos en PSeInt y Python

. Introducción

En este informe se describen las implementaciones de tres programas en los lenguajes **PSeInt** y **Python**. Se detallan las decisiones sobre tipos de datos, el uso de variables y constantes, y la aplicación de operaciones y estructuras de control en cada lenguaje.

Elección de Tipos de Datos

Los tipos de datos utilizados en ambos lenguajes incluyen:

- **Enteros (int):** Para representar valores numéricos sin decimales, como edades.
- **Flotantes (float/real):** Para operaciones aritméticas con decimales, como divisiones.
- **Cadenas de texto (string):** Para almacenar nombres y mensajes de salida.

En **PSeInt**, los tipos de datos se definen explícitamente (por ejemplo, Definir num1 Como Real), mientras que en **Python**, el tipado es dinámico y no requiere declaración explícita.

Uso de Variables y Constantes

Las variables fueron utilizadas para almacenar valores ingresados por el usuario y resultados intermedios en los cálculos. En **PSeInt**, se declararon antes de su uso, mientras que en **Python**, se asignaron directamente.

Ejemplo en PSeInt:

1. Definir nombre Como Cadena;
2. Definir edad Como Entero;
3. Leer nombre;
4. Leer edad;

Ejemplo en Python:

```
nombre = input("Ingrese su nombre: ")
```

```
edad = int(input("Ingrese su edad: "))
```

Las constantes, como el valor de **Pi**, fueron utilizadas para operaciones específicas, pero **PSeInt** no tiene una forma explícita de definir constantes, a diferencia de **Python**, donde se usa una convención (nombres en mayúsculas).

Ejemplo en Python:

```
PI = 3.1416
```

```
radio = 5
```

```
area = PI * radio**2
```

Implementación de Operaciones y Estructuras

Operaciones Matemáticas

Ambos lenguajes soportan operaciones aritméticas (+, -, *, /). La división por cero se manejó en ambos lenguajes con una estructura condicional.

Ejemplo en PSeInt:

Si num2 \neq 0 Entonces

```
division <- num1 / num2;
```

Sino

```
Escribir "No se puede dividir por cero."
```

FinSi

Ejemplo en Python:

```
if num2 != 0:
```

```
    division = num1 / num2
```

```
else:
```

```
print("No se puede dividir por cero.")
```

Comparaciones y Condicionales

Las comparaciones lógicas permitieron evaluar los resultados de las operaciones matemáticas y mostrar mensajes condicionales.

Ejemplo en PSeInt:

Si suma > multiplicacion Entonces

Escribir "La suma es mayor que la multiplicación."

Sino

Escribir "La multiplicación es mayor o igual que la suma."

FinSi

Ejemplo en Python:

```
if suma > multiplicacion:
```

```
    print("La suma es mayor que la multiplicación.")
```

```
else:
```

```
    print("La multiplicación es mayor o igual que la suma.")
```

La implementación de estos programas en **PSeInt** y **Python** permitió comparar la forma en que cada lenguaje maneja variables, constantes, operaciones y estructuras de control. **PSeInt**, al estar diseñado para aprendizaje, proporciona una sintaxis más estructurada y cercana al pseudocódigo. **Python**, en cambio, es un lenguaje de programación real con tipado dinámico y una sintaxis más flexible.

Ambos enfoques tienen ventajas dependiendo del contexto de uso. **PSeInt** es ideal para comprender la lógica algorítmica sin preocuparse por detalles del lenguaje, mientras que **Python** permite implementar soluciones reales con mayor eficiencia y aplicabilidad en desarrollo de software.