TC1028 Pensamiento Computacional para Ingeniería

Componentes de un programa y tipos de datos

Tecnológico de Monterrey





Componentes de un programa

- 1. Identificador
- 2. Variable
- 3. Constante
- 4. Instrucciones





Identificador



Identificador

Son nombres dados a variables, constantes y funciones de un programa. Se forman con la combinación de letras, números y otros símbolos. El primer carácter debe ser una letra.

Ejemplos: PI, vocales, x, i, etc.





Variable



Variable

Es un identificador que puede cambiar durante la ejecución de un programa.

Ejemplos:
$$x = 1$$

 $y = 2$
 $i = 1$
 $x = x + y$
 $i = i + 1$





Constante



Constante

Es un identificador al que se le asigna un valor fijo; es decir, no cambia durante la ejecución del programa. Puede ser un número, un carácter o una lista de caracteres.

Ejemplos:

PUERTO = 3307 USUARIO = "root" PASSWORD = "123456"





Instrucciones

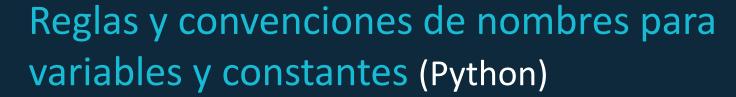


Instrucciones

Unidad ejecutable más pequeña de un programa. Las instrucciones controlan el flujo u orden de ejecución.

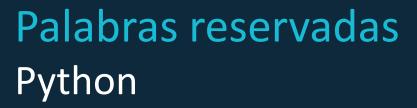
Ejemplos: and, break, continue, elif, else, for, if, import, is, not, or, print, return, while, with, etc.





- El primer carácter no puede ser un número o dígito.
- ♣ Inicie con letra o guion bajo _ (El resto puede ser letra, número o guion bajo _)
 NOTA: Los nombres que comienzan con guion bajo (_simple, _ _o doble) se reservan para variables con significado especial.
- ❖ No use **símbolos especiales** como !, @, #, \$, %, etc.
- ❖ Los nombres de las variables pueden tener la combinación de letras en minúsculas (a z) o MAYÚSCULAS (A Z) o dígitos (0 9) o un guion bajo (_). Por ejemplo:
 - ☐ snake case
 - ☐ MACRO_CASE
 - □ camelCase
 - No pueden usarse como identificadores, las palabras reservadas.





Existen ciertas palabras que tienen significado especial para el intérprete de Python. Estas no pueden utilizarse para nombrar variables o constantes.

Algunas palabras reservadas son las siguientes:

- ❖ and
- break
- continue
- ❖ def
- elif
- else
- for
- from

- 🌣 if
- import
- in
- is
- not
- or
- print
- return

- while
- with





Señala si el nombre de las siguientes variables es correcto/incorrecto:

- 1. _num_Alumno
- 2. #exterior
- 3. 5Telefono
- 4. Radio
- 5. direccion_casa
- 6. miPerro
- 7. nombre Pila

- 8. nombreProceso
- 9. correo_electronico
- 10. practica2
- 11. nombre del perro
- 12. teléfono
- 13. numero-lista
- 14. while



Tipos de datos

Son los valores que puede tomar una variable.

Los tipos de datos son los siguientes:

- Numéricos
- Alfanuméricos
- Lógicos





Tipos de datos

- ❖ Datos numéricos: Pueden representarse de dos formas:
 - ☐ Números enteros, los cuales no tienen componentes fraccionarios y pueden ser positivos o negativos
 - Números reales, pueden tener cifras decimales y pueden ser positivos o negativos.
- ❖ Datos alfanuméricos: Son datos que contienen caracteres no numéricos ya sean letras, caracteres especiales (,.=´+) o los dígitos mismos.
- ❖ Datos Lógicos: Podemos hablar de otro tipo de datos llamado "booleano", el cual sólo puede tomar uno de dos valores : verdadero o falso.





Tipos de datos Numéricos

Tipo de dato	Python	C
Entero	num = 40	int num = 40;
Real	numReal = 45.2	float numReal = 45.2;

Como puede ver en **Python**, a diferencia de muchos otros lenguajes, no se declara el **tipo de la variable** al crearla.





Tipos de datos Alfanuméricos

Tipo de dato	Python	С
Caracter	letra = 'c'	char letra = 'a';
Cadena de caracteres	c = "Hola mundo" c = "12345"	<pre>char c[10] = "Hola mundo";</pre>

Como puede ver en **Python**, a diferencia de muchos otros lenguajes, no se declara el **tipo de la variable** al crearla.





Tipos de datos

Lógicos o booleanos

Tipo de dato	Python	C
Booleano	x = True x = 5 > 3	bool x = True;
Booleano	y = False y = 3 > 5	bool y = False;

Como puede ver en **Python**, a diferencia de muchos otros lenguajes, no se declara el **tipo de la variable** al crearla.





Tipos de datos

Python ve diferente la variables si tiene mayúsculas o minúsculas:

Ejemplo:

En este ejemplo, las dos variables son distintas

```
Nombre = "juan"
nombre = "pedro"
```





Thonny

- Instala Thonny: https://thonny.org/
- Thonny es un entorno de desarrollo integrado (IDE) para el lenguaje Python diseñado para principiantes.







Tipos de datos

Python

- Entero (Integer)
- Real (Float)
- Lógico o booleano (Boolean)
- Cadena de caracteres (String)







- Inicializa 5 variables con diferentes tipos de datos (entero, real, booleano, string)
- Con la instrucción type averigua el tipo de dato detectado.

Ejemplo:





Escribe un algoritmo para verificar si un precio dado por el usuario es válido o no lo es, para ser válido debe ser un valor positivo o cero.

Casos de prueba:

precio	Resultado
10	El precio es válido
0	El precio es válido
-3	El precio no es válido



Actividad grupal Algoritmo

- 1. Pedir el precio
- 2. Si el precio es mayor o igual a cero

Escribir "El precio es válido"

SiNo

Escribir "El precio no es válido"



Crear un algoritmo que dados tres números encuentre el más pequeño.

Casos de prueba:

radio	área
10, 23, 1	1
200, -23, 9	-23
0, 0, 0	0



Algoritmo

```
1. Pedir a, b, c
2. Si (a < b y a < c)
     menor = a
  SiNo
     Si (b < a y b < c)
       menor = b
     SiNo
       menor = c
   Escribir menor
```



Definir el **algoritmo** y utiliza **Thonny** para diseñar el **programa** en **Python**:

- Pedir al usuario el radio.
- Calcular el área de un círculo a partir de su radio con la fórmula:
 area = PI * radio²
- Guarda el nombre de tu archivo en Python: areaCirculo.py

Casos de prueba:

radio	área
10	314.1592
5	78.5398
-5	78.5398



Actividad grupal Algoritmo

- 1. Definir el valor de PI como 3.141592
- 2. Pedir el radio
- 3. area = PI * radio*radio
- 4. Escribir (area)



Programa

```
areaCirculo.py
   # Programa que calcula el área de un círculo
    """ 1. Definir el valor de PI como 3.141592
    Pedir el radio
    area = PI * radio*radio
     4. Escribir (area) """
 7 PI = 3.141592
 8 radio = float(input("Introduce el valor del radio: "))
 9 area = PI * radio * radio
10 print ("El area del circulo es:", area)
Shell ×
```

>>> %Run areaCirculo.py

Introduce el valor del radio: 10 El area del circulo es: 314.1592





Definir el **algoritmo** y utiliza **Thonny** para diseñar el **programa** en **Python**:

- Un estudiante desea conocer el área de un triángulo a partir de la base y la altura.
- ❖ Usar la fórmula: area = b*h/2
- Guarda el nombre de tu archivo en Python: areaTriangulo.py

Casos de prueba:

base	altura	área
10	2	10
25	15	187.5
5.5	2.8	7.69999



Definir el **algoritmo** y utiliza **Thonny** para diseñar el **programa** en **Python**:

- ❖ Convierta el precio de un producto de pesos a dólares, si se tiene el tipo de cambio del dólar y el precio en pesos del producto. El resultado debe mostrar "El precio del producto en dólares es:" X .
- Guarda el nombre de tu archivo en Python: pesosDolares.py

Casos de prueba:

tipoCambio	precioPesos	precioDolares
19.5	100	5.128205
20	350.5	17.525
21.2	230	10.849056



Definir el **algoritmo** y utiliza **Thonny** para diseñar el **programa** en **Python**:

- Un alumno desea conocer la calificación final de su materia de Programación.
- La rúbrica de esta materia se compone de la siguiente manera:

Parcial 120%

Parcial 2 35%

Proyecto final 15%

Examen final 30%

Guarda el nombre de tu archivo en Python: calificacion.py



Gracias

