

En Python, una **variable** es un nombre que se usa para almacenar un valor en memoria. Funciona como un contenedor que guarda datos (números, texto, booleanos, etc.) y permite manipularlos fácilmente dentro de un programa [1](#) [2](#) [3](#).

Conceptos básicos de variables en Python

- **Definición:**
Una variable es una ubicación nombrada en la memoria que guarda un valor. Por ejemplo:
 - edad = 25
 - nombre = "Ana"
 - es_estudiante = True

Aquí edad, nombre y es_estudiante son variables con diferentes tipos de datos [1](#).

- **Tipado dinámico:**
En Python no es necesario declarar el tipo de la variable. El tipo se asigna automáticamente según el valor:
 - x = 10 # entero
 - x = "texto" # ahora es cadena

Esto se conoce como **tipado dinámico** [1](#).

- **Asignación:**
Se usa el operador = para asignar valores:
 - precio = 99.5
 - **Convenciones de nombres:**
 - Usar minúsculas y guiones bajos para separar palabras: first_name.
 - Evitar palabras reservadas como if, for, while.
 - Ser descriptivo: contador es mejor que c.
 - Python distingue mayúsculas de minúsculas: Edad y edad son diferentes [1](#).
-

Ejemplos prácticos

Variable numérica

```
contador = 0
```

```
# Variable de texto
```

```
mensaje = "Hola, mundo"
```

```
# Variable booleana
```

```
activo = True
```

```
# Reasignación
```

```
contador = contador + 1
```

Puntos importantes a tener en cuenta

- **No necesitas declarar antes de usar:** basta con asignar un valor.
- **El tipo puede cambiar:** una variable puede pasar de ser número a texto.
- **Buenas prácticas:** usar nombres claros y consistentes facilita la lectura y mantenimiento del código.
- **Errores comunes:** usar palabras reservadas como nombres de variables o mezclar tipos sin cuidado puede causar problemas

Hasta este momento hemos visto como definir variables enteras y flotantes.

Realizar su carga por asignación y por teclado.

Para iniciarlas por asignación utilizamos el operador =

```
#definición de una variable entera
```

```
cantidad=20
```

```
#definición de una variable flotante
```

```
altura=1.92
```

Como vemos el intérprete de Python diferencia una variable flotante de una

variable entera por la presencia del caracter punto.

Para realizar la carga por teclado utilizando la función input debemos llamar a la función int o float para convertir el dato devuelto por input:

```
cantidad=int(input("Ingresar la cantidad de personas:"))  
altura=float(input("Ingresar la altura de la persona en metros ej:1.70:"))
```

A estos dos tipos de datos fundamentales (int y float) se suma un tipo de dato muy utilizado que son las cadenas de caracteres.

Una cadena de caracteres está compuesta por uno o más caracteres. También podemos iniciar una cadena de caracteres por asignación o ingresarla por teclado.

Inicialización de una cadena por asignación:

```
#definición e inicio de una cadena de caracteres  
dia="lunes"
```

Igual resultado obtenemos si utilizamos la comilla simple:

```
#definición e inicio de una cadena de caracteres  
dia='lunes'
```

Para la carga por teclado de una cadena de caracteres utilizamos la función input que retorna una cadena de caracteres:

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

Problema 1:

Realizar la carga por teclado del nombre, edad y altura de dos personas. Mostrar por pantalla el nombre de la persona con mayor altura.

```
print("Datos de la primer persona")  
nombre1=input("Ingrese nombre:")
```

```
edad1=int(input("Ingrese la edad:"))
altura1=float(input("Ingrese la altura Ej 1.75:"))
print("Datos de la segunda persona")
nombre2=input("Ingrese nombre:")
edad2=int(input("Ingrese la edad:"))
altura2=float(input("Ingrese la altura Ej 1.75:"))
print("La persona mas alta es:")
if altura1>altura2:
    print(nombre1)
else:
    print(nombre2)
```

Problema 2:

Realizar la carga de dos nombres por teclado. Mostrar cual de los dos es mayor alfabéticamente o si son iguales.

```
nombre1=input("Ingrese el primer nombre:")
nombre2=input("Ingrese el segundo nombre:")
if nombre1==nombre2:
    print("Ingreso dos nombre iguales")
else:
    if nombre1>nombre2:
        print(nombre1)
        print("es mayor alfabeticamente")
    else:
        print(nombre2)
        print("es mayor alfabeticamente")
```