



Fundamentos de Programación en C++

Clase Muestra: Introducción a las Variables

Jhovanny Andres Mejia Guisao
Departamento de Física y Matemáticas

¿Qué es una variable?

CONCEPTO BÁSICO

"Una variable es como una CAJA donde guardamos información"

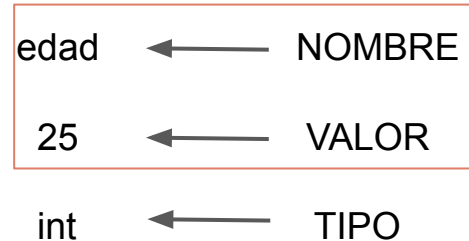
- ◆ Tiene un NOMBRE (etiqueta)
- ◆ Almacena un VALOR (contenido)
- ◆ Tiene un TIPO (qué puede guardar)

Ejemplo en la vida real:

- 📦 Caja etiquetada "Galletas" → Contiene galletas
- 📦 Caja etiquetada "Libros" → Contiene libros

🎯 ANALOGÍA PRÁCTICA

```
int edad = 25;
```



Tipos de datos básicos



TIPOS DE DATOS MÁS COMUNES

- `int` → Números enteros (ej: 10, -5, 1000)
- `float` → Números decimales (ej: 3.14, -2.5)
- `char` → Un solo carácter (ej: 'A', 'x', '!')
- `string` → Texto (ej: "Hola", "Ana")
- `bool` → Verdadero/Falso (ej: true, false)



EJEMPLOS EN C++

```
int edad = 20;
```

```
float altura = 1.75;
```

```
char inicial = 'M';
```

```
string nombre = "María";
```

```
bool esEstudiante = true;
```

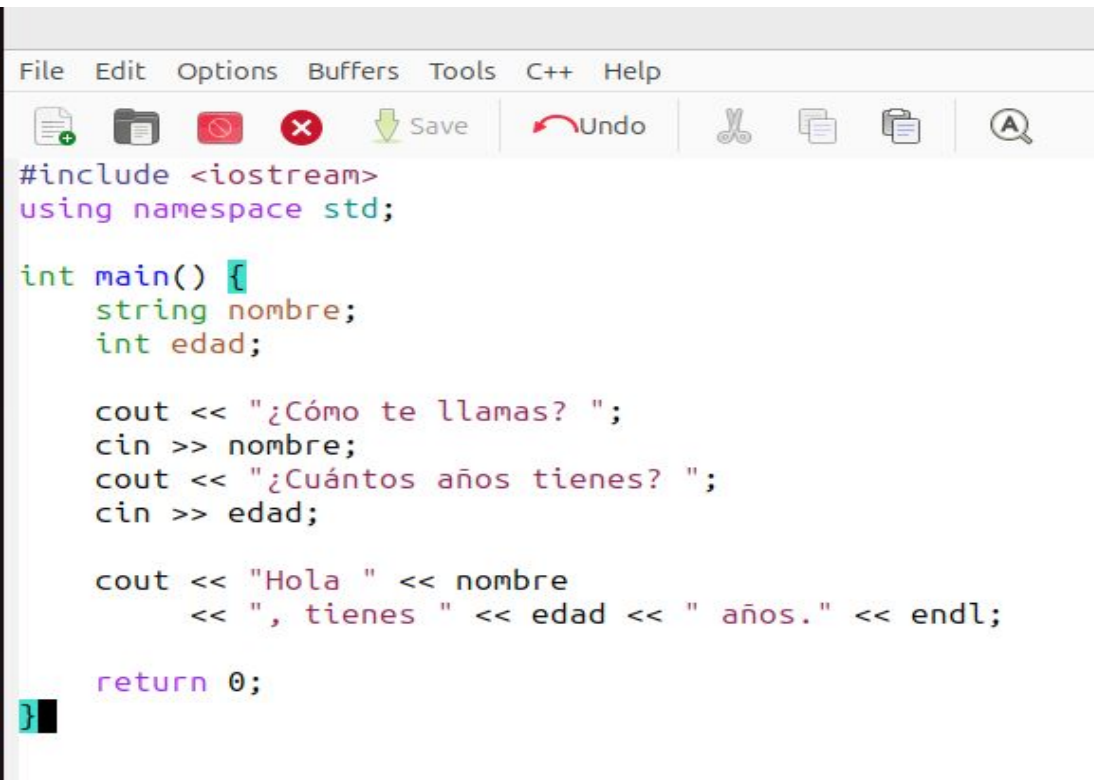
Mini-ejercicio participativo

🤔 ¿QUÉ TIPO USARÍAS?

1. Para guardar el precio de un chocolate → _____
2. Para guardar si una persona es mayor de edad → _____
3. Para guardar la primera letra de un nombre → _____
4. Para guardar el nombre completo → _____

💡 Pista: int, float, char, string, bool

Código en vivo



```
File Edit Options Buffers Tools C++ Help
[Icons: New, Open, Save, Undo, Cut, Copy, Paste, Find]
#include <iostream>
using namespace std;

int main() {
    string nombre;
    int edad;

    cout << "¿Cómo te llamas? ";
    cin >> nombre;
    cout << "¿Cuántos años tienes? ";
    cin >> edad;

    cout << "Hola " << nombre
         << ", tienes " << edad << " años." << endl;

    return 0;
}
```

<https://github.com/jmejiagu/FundamentosProgra-IBERO>

Console Output (`cout`)

Console Input (`cin`)

Pasos para compilar y ejecutar en la consola

```
g++ -o mi_ejecutable ClaseMuestra.cpp
./mi_ejecutable
```

```
g++ ClaseMuestra.cpp -o mi_ejecutable2
./mi_ejecutable
```

```
g++ ClaseMuestra.cpp
./a.out
```

Reglas para nombrar variables



REGLAS IMPORTANTES



Permitido:

- Letras y números: edad1, total2024
- Guión bajo: nombre_usuario
- Sensible a mayúsculas: Edad \neq edad



No permitido:

- Espacios: nombre usuario
- Caracteres especiales: precio@, total\$
- Empezar con número: 1nombre






Buena práctica: nombres descriptivos

Resumen

LO QUE APRENDIMOS HOY

- ✓ Las variables son contenedores de información
- ✓ Tienen nombre, tipo y valor
- ✓ Principales tipos: int, float, char, string, bool
- ✓ Seguir reglas para nombrarlas correctamente
- ✓ Podemos usarlas para programas interactivos

PRÓXIMOS PASOS

-  Practicar declarando variables
-  Experimentar con diferentes tipos
-  Consultar la bibliografía del curso

¡Gracias por su atención!
¿Preguntas?

Backup slides

Variables y Tipos de datos

Variables Locales: Se definen solo en bloque en el que se vayan a ocupar, de esta manera evitamos tener variables definidas que luego no se utilizan.

Variables Globales: No son lo más recomendable, pues su existencia atenta contra la comprensión del código y su encapsulamiento.

Variables estáticas: Se tienen que inicializar en el momento en que se declaran, de manera obligatoria.

Los tipos de datos pueden ser predefinidos o abstractos. Un tipo de dato **predefinido** es intrínsecamente comprendido por el compilador. En contraste, un tipo de datos definido por el usuario es aquel que usted o cualquier otro programador crea como una clase, que comúnmente son llamados tipos de datos **abstractos**.

Los tipos de datos más comunes en C++ son:

<i>TipodeDato</i>	<i>EspacioenMemoria</i>	<i>Rango</i>
unsigned char	8 bits	0 a 255
char	8 bits	-128 a 127
short int	16 bits	-32,768 a 32,767
unsigned int	32 bits	0 a 4,294,967,295
int	32 bits	-2,147,483,648 a 2,147,483,647
unsigned long	32 bits	0 a 4,294,967,295
enum	16 bits	-2,147,483,648 a 2,147,483,647
long	32 bits	-2,147,483,648 a 2,147,483,647
float	32 bits	3.4×10^{-38} a $3.4 \times 10^{+38}$ (6 dec)
double	64 bits	1.7×10^{-308} a $1.7 \times 10^{+308}$ (15 dec)
long double	80 bits	3.4×10^{-4932} a $1.1 \times 10^{+4932}$
void	sin valor	

Decimal	Exponencial	cientifica
1625.0	1.625e3	1.625×10^3
0.00731	7.31e-3	7.31×10^{-3}

Bases de c++

Esta línea se corresponde con el comienzo de la definición de la función principal “**main**”. La función principal es el punto por donde todos los programas inician su ejecución, independientemente de su ubicación dentro del código fuente.

Utilice los objetos de la biblioteca `iostream` para imprimir cadenas a la salida estándar. Los nombres `std :: cout` y `std :: endl` se declaran en el "archivo de encabezado".

```
//Mi primer programa en C++  
#include<iostream>  
  
int main(){  
    std::cout<< "Bienvenidos a c++! \n";  
    return 0;  
}
```

Todas las líneas que comiencen con dos signos barra (`//`) se consideran comentarios y no tienen ningún efecto sobre el comportamiento del programa

Las líneas que comienza con un símbolo de `"#"` son directivas para el preprocesador. Este archivo específico (**`iostream`**) incluye las declaraciones de la norma básica de entrada y salida de la biblioteca de C++.

Esta declaración hace que la función principal termine. Un código de retorno es 0, cuando la función principal interpreta de manera general que el programa trabajó como se esperaba, sin ningún error durante su ejecución.