



---

# Sesión 1

Aprende C desde cero

The background features a dark gray gradient with two large, overlapping circles. The left circle is solid black, and the right circle is a lighter shade of gray. The text is positioned on the left side, within the black circle.

# **ESTRUCTURA DEL PROGRAMA**

# Visión global

```
/* Bibliotecas que utiliza el programa */  
#include <stdio.h>  
  
/* Punto de inicio del programa */  
int main(void) {  
  
    /* Cuerpo del programa */  
    printf("Hola mundo\n");  
  
    /* Acaba la ejecución del programa */  
    return 0;  
}
```

# Visión global

```
/* Bibliotecas que utiliza el programa */  
#include <stdio.h>
```

```
/* Punto de inicio del programa */  
int main(void) {
```

```
    /* Cuerpo del programa */  
    printf("Hola mundo\n");
```

```
    /* Acaba la ejecución del programa */  
    return 0;  
}
```

# Comentarios

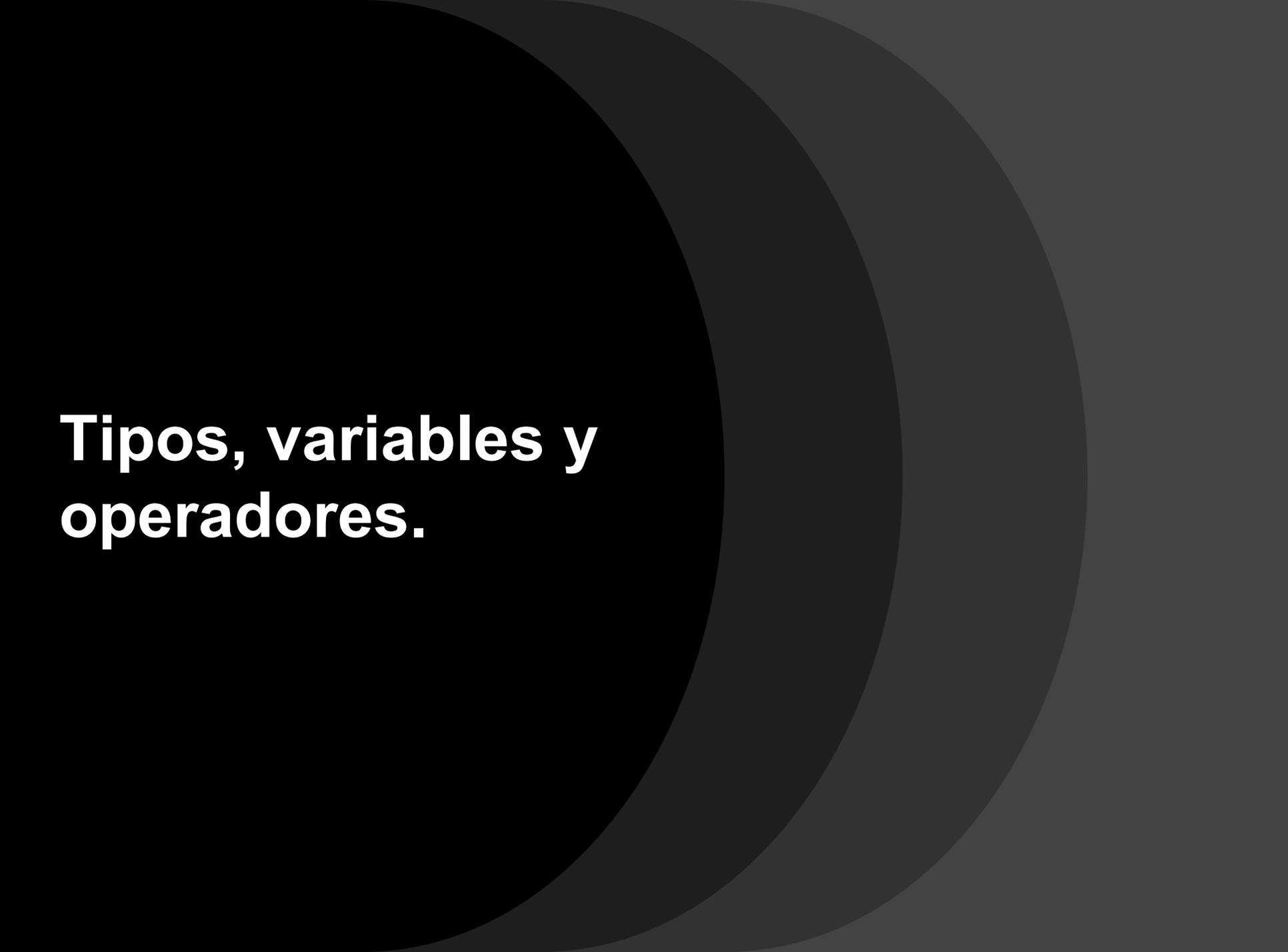
*“Líneas de código que el compilador ignora y sirven para ayudar al programador a hacer más legible su código.”*

Tipo	Palabra Clave
Línea	//
Bloque	<code>/* texto */</code>

```
/* Este fragmento de código es un ejemplo de  
como comentar bloques de código.*/
```

```
// La línea de abajo SÍ que se ejecutará:  
printf("Hola mundo\n");
```

```
/* Todo lo que ponga aquí dentro va a ser  
ignorado... */
```



**Tipos, variables y  
operadores.**

# Tipos de datos básicos

Tipo	Palabra Clave	Representan
Entero	<code>int</code>	Los números naturales
Decimal	<code>float</code>	Los números reales
Carácter	<code>char</code>	Las letras y cadenas

*Aunque hay más tipos de datos, nosotros nos centraremos en los básicos, dejando fuera del alcance del curso el resto.*

# Variables

Las variables se utilizan para almacenar un tipo de dato y su sintaxis es:

```
tipo_de_dato nombre_de_variable [= valor];
```

Dónde:

- **tipo\_de\_dato** indica el dato a almacenar.
- **nombre\_de\_variable** indica con qué seudónimo o alias se va a conocer este dato a lo largo del código.
- **valor** indica el contenido inicial de la variable. No es obligatorio.

Ejemplos:

```
int year = 2017;  
int piso;  
float pi = 3.1415, radio2 = 2.0;
```

Restricciones para nombrar variables:

- No puede ser una palabra clave.
- No puede empezar por un número.
- No puede contener acentos, tildes, ñ...

# Marcas de formato

Sirven para indicar en qué formato se debe representar un valor por pantalla

```
printf("Pi vale %f\n", pi);
```

Donde:

- **printf** es la función que se encarga de mostrar el mensaje por pantalla.
- **%f** es la marca de formato para números reales.
- **pi** es una variable del tipo float.

Representar	Marca de formato
Entero	<b>%d</b>
Decimal	<b>%f</b>
Carácter	<b>%c</b>
Cadena	<b>%s</b>
Salto de línea	<b>\n</b>
Tabulador	<b>\t</b>

# Ejemplo: marcas de formato

```
/* Bibliotecas que utiliza el programa */
#include <stdio.h>

/* Punto de inicio del programa */
int main(void) {

    /* Declaraciones e inicializaciones */
    char *name = "Sergio";
    int day = 1;
    int month = 1;
    int year = 2017;

    /* Cuerpo del programa */
    printf("Soy %s\nHoy es %d del %d de %d y
digo:\t;Hola!\n", name, day, month, year);

    /* Acaba la ejecución del programa */
    return 0;
}
```

# Operadores

Tipo	Palabra Clave
Suma	+
Resta	-
Producto	*
División	/
Resto de división	%

```
// Declaración de variables
    int lado, area;
// Inicialización de variables
    lado = 100;
/* Asignación del resultado de una
operación a una variable */
    area = lado * lado;
// Mostrar el resultado
    printf("El area de un cuadrado de
lado %d es: %d\n", lado, res);
```



**Ejercicio**

# Ejercicio: Calculadora

Desarrolla un programa en C que dados dos números calcule su: suma, resta, producto y división.

- Los dos números deberán almacenarse en variables.
- Los resultados de las operaciones deberán guardarse en variables.
- Tras la ejecución del programa se mostrará para cada operación su resultado.

Ejemplo:

Dados los números 5 y 2, la salida del programa debería ser:

```
Bienvenido/a a "Calcooladora v1.0".  
  
Primera cifra: 5.00  
Segunda cifra: 2.00  
  
...calculando resultados...  
  
Suma: 7  
Resta: 3  
Producto: 10  
División: 2.5  
  
¡Adiós!
```

La salida debería ser lo más parecido a la que se muestra, teniendo en cuenta los tabuladores y los saltos de línea.

## Recuerda que puedes:

Interactuar con otros alumnos en el foro del curso:

<http://mooc.uji.es/mod/forum/view.php?id=1654>

Acceder a todos los materiales en el repositorio:

<https://siserte@bitbucket.org/siserte/repositoriocursoc.git>

Contactar con los profesores:

- [adcastel@uji.es](mailto:adcastel@uji.es)
- [siserte@uji.es](mailto:siserte@uji.es)