

Primeros programas en C++

- “Hola mundo” en C++ :

// programa hola.cpp

```
#include <iostream>

using namespace std; ///// define el espacio de nombres std y evita usar esta directiva

int main()
{
    cout << "hola mundo" << endl;
    return 0;
}
```

compilación:

>> g++ hola.cpp -o hola

(g++ ejecuta el compilador de C++)

Primeros programas en C++

- Otro ejemplo: cin, cout: entrada / salida estandar

```
// programa numeros
#include <iostream>

using namespace std;

int main()
{
    cout << "Introducir dos numeros" << endl;
    int n1, n2;

    cin >> n1 >> n2;

    cout << "la suma de " << n1 << " + " << n2 << " es: " << n1 + n2 << endl;

    return 0;
}

// g++ numeros.cpp -o numeros
```

Además, cerr, clog son objetos para generar mensajes de error e información relativa al programa.

Clases en C++

usoPersona.cpp

```
#include <iostream>
#include "Persona.h"

int main()
{
    Persona P1, P2;

    P1.setDatos("Pepe", 123456);
    P1.mostrarDatos();
    P2.mostrarDatos();

    return 0;
}
```

Compilación y ejecución en línea de comandos (una posibilidad):

>> g++ -c Persona.cpp usoPersona.cpp → se generan ambos .o (código objeto)

>> g++ -o uso Persona.o usoPersona.o → se genera el ejecutable uso (código ejecutable)

>> ./uso

Clases en C++

Clase Persona: Persona.h

```
class Persona
{
    private:
        char nombre[30];
        long dni;

    public:
        Persona();
        void setDatos(char nombre[], long doc);
        void mostrarDatos();
};
```

ATRIBUTOS

Persona.cpp

```
#include <iostream>
#include <string.h>
#include "Persona.h"

using namespace std;

Persona::Persona()
{
    nombre[0] = '\0';
    dni = 0;
}

void Persona::setDatos(char nom[], long doc)
{
    strcpy(nombre, nom);
    this->dni = doc;
}

void Persona::mostrarDatos()
{
    cout << "Nombre: " << this->nombre << " dni: " << this->dni << endl;
}
```

FUNCIONALIDAD
COMPORTAMIENTO

Constructores

- Un constructor es una función miembro de propósito específico que se ejecuta automáticamente cuando se crea un objeto de una clase.
- Sirve para inicializar los miembros dato de una clase.
- Tiene el mismo nombre que la clase.
- No retorna ningún valor → no se puede especificar un valor de retorno (ni void).
- Puede tomar cualquier número de parámetros (0 o más).
- Permite que un objeto se pueda inicializar por si mismo cuando se crea, sin necesidad de llamar a otra función.
- Se utiliza normalmente para inicializar los atributos.

Clases en C++: clases Fecha y Persona

Fecha.h

```
class Fecha
{
private:
    int dia;
    int mes;
    int anio;

public:
    Fecha();
    Fecha(int d, int m, int a);
    void estFecha(int d, int m, int a);
    void mostrarFecha();
};
```

Fecha.cpp

```
#include <iostream>
#include "Fecha.h"

Fecha::Fecha()
{
    dia = 1;
    mes = 1;
    anio = 2999;
}

Fecha::Fecha(int d, int m, int a)
{
    dia = d;
    mes = m;
    anio = a;
}

void Fecha::estFecha(int d, int m, int a)
{
    dia = d;
    mes = m;
    anio = a;
}

void Fecha::mostrarFecha()
{
    std::cout << dia << '/' << mes << '/' << anio << std::endl;
}
```

Clases en C++: clases Fecha y Persona

Persona.h

```
#include "Fecha.h"

class Persona
{
    private:
        char nombre[30];
        long dni;
        Fecha fechaNac;

    public:
        Persona();
        void setDatos(char nombre[], long doc);
        void mostrarDatos();
};
```

Persona.cpp

```
#include <iostream>
#include <string.h>
#include "Persona.h"

using namespace std;

Persona::Persona()
{
    nombre[0] = '\0';
    dni = 0;
}

void Persona::setDatos(char nom[], long doc)
{
    strcpy(nombre, nom);
    this->dni = doc;
}

void Persona::mostrarDatos()
{
    cout << "Nombre: " << this->nombre << " dni: " << this->dni << " Fecha de nacimiento: " << endl;
    (this->fechaNac).mostrarFecha();
}
```

Clases en C++: clases Fecha y Persona

usoPersona.cpp

```
#include <iostream>
#include "Persona.h"

int main()
{
    Persona P1, P2;

    P1.setDatos("Pepe", 123456);
    P1.mostrarDatos();
    P2.mostrarDatos();

    return 0;
}
```

Compilación y ejecución en línea de comandos:

>> g++ -c Fecha.cpp Persona.cpp usoPersona.cpp → se generan los .o

>> g++ -o uso usoPersona.o Persona.o Fecha.o → se genera el ejecutable uso

>> ./uso