

# Jabes Estevez 2025-0905

## Video 1:

```
main.cpp
1 // Jabes Estevez 2025-0905
2 #include <iostream>
3 using namespace std;
4
5 class mensajes {
6     public:
7     void MensajePuntos(){
8         cout << "....." << endl;
9         cout << "mensaje de puntos" << endl;
10        cout << "....." << endl;
11    }
12
13    void MensajeLineas(){
14        cout << "-----" << endl;
15        cout << "mensaje de lineas" << endl;
16        cout << "-----" << endl;
17    }
18    void MensajeAsteriscos(){
19        cout << "*****" << endl;
20        cout << "mensaje de MensajeAsteriscos" << endl;
21        cout << "*****" << endl;
22    }
23 };
24
25 int main(){
26
27     mensajes miMensaje, otromensaje;
28     otromensaje.MensajeAsteriscos();
29     miMensaje.MensajeAsteriscos();
30     miMensaje.MensajePuntos();
31     miMensaje.MensajeLineas();
32 }

```

\*\*\*\*\*
mensaje de MensajeAsteriscos
\*\*\*\*\*
\*\*\*\*\*
mensaje de MensajeAsteriscos
\*\*\*\*\*
.....
mensaje de puntos
-----
mensaje de lineas
-----
...Program finished with exit code 0
Press ENTER to exit console.

## Video 2

The screenshot shows a C++ development environment with the following details:

- Toolbar:** Includes icons for file operations (New, Open, Save), run (Run, Stop, Share), and settings.
- Code Editor:** The file `main.cpp` is open. The code defines a `personas` class with private members `nombre` (string) and `peso` (float). It includes methods to set and get both values. The `main` function creates an instance of `personas`, sets the name to "Joselu", sets the weight to 43, and then prints both values to the console.
- Output Window:** Displays the program's output:

```
Joselu
43
...Program finished with exit
Press ENTER to exit console.
```

```
1 // Jabel Estevez 2025-0905
2 #include <iostream>
3 using namespace std;
4
5 class personas{
6     private:
7         string nombre;
8         float peso;
9
10    public:
11        void setNombre(string _nombre){
12            nombre = _nombre;
13        }
14
15        string getnombre(){
16            return nombre;
17        }
18        void setpeso(float elpeso){
19            peso = elpeso;
20        }
21        float getpeso(){
22            return peso;
23        }
24    };
25 int main(){
26     personas unapersona;
27     unapersona.setNombre("Joselu");
28     cout << unapersona.getnombre();
29     unapersona.setpeso(43);
30     cout << endl;
31     cout << unapersona.getpeso();
32 }
```

## Video 3

main.cpp

```
1 // Jabel Estevez 2025-0905
2 #include <iostream>
3 using namespace std;
4
5 class animal{
6     private:
7     string especie;
8
9 public:
10    animal() { especie = "sin especie"; }
11    animal(string _especie) { especie = _especie; } // constructor
12    string getespecie(){return especie;}
13};
14 int main(){
15     animal tarzan("perro"), boli("gato"), billgates;
16
17     cout << tarzan.getespecie() << endl;
18     cout << boli.getespecie() << endl;
19     cout << billgates.getespecie();
20
21 }
```



perro

gatosin especie

...Program finished with exit code 0

Press ENTER to exit console.

## Video 4

```
1 // Jabel Estevez 2025-0905
2 #include <iostream>
3 using namespace std;
4
5 class animal{
6     private:
7         string especie;
8
9     public:
10         animal() { especie = "sin especie"; }
11         animal(string _especie) { especie = _especie; } // constructor
12         string getespecie(){return especie;}
13         ~animal(){
14             cout << "destructor de: " << especie << endl;
15         }
16     };
17 int main(){
18 {
19     animal tarzan("perro"), boli("gato"), billgates;
20
21     cout << tarzan.getespecie() << endl;
22     cout << boli.getespecie() << endl;
23     cout << billgates.getespecie() << endl;
24     cout << "bloque de codigo" << endl;
25 }
26     cout << "finc de la fucnion";
27 }
```

```
perro
gato
sin especie
bloque de codigo
destructor de: sin especie
destructor de: gato
destructor de: perro
finc de la fucnion

...Program finished with exit code 0
Press ENTER to exit console.
```

Video 5:

```
1 //Jabes Estevez
2 #include <iostream>
3 using namespace std;
4 class punto{
5     public:
6         int x, y;
7         punto(int _x = 0 ,int _y= 0){x = _x; y = _y;}
8     };
9
10 void mostrarpunto(punto p){
11     cout << "x:" <<p.x << endl;
12     cout << "y:" <<p.y << endl;
13 }
14
15 int main(){
16     mostrarpunto(punto(23,34));
17 }
```



x:23

y:34

## Video 6

```
mampapp  
1 //Jabes Estevez  
2 using namespace std;  
3 #include <iostream>  
4  
5 class coche{  
6 private:  
7     string marca;  
8     string color;  
9 public:  
10    coche(){ marca = "sin marca"; color = "sin color";}  
11    coche(string _marca, string _color){ marca = _marca; color =  
12        _color;}  
13    //consultores  
14  
15    string getmarca() const {return marca;}  
16    string getcolor() const {return color;}  
17 };  
18  
19  
20 int main()  
21 {  
22    coche micoche("ferrari", "rojo");  
23    cout << micoche.getmarca() << endl;  
24    cout << micoche.getcolor();  
25 }
```

ferrari  
rojo  
== Code Execution Successful ==

## Video 7

main.cpp



Run

Output

```
1 //Jabes Estevez
2 using namespace std;
3 #include <iostream>
4
5 class coche{
6 private:
7     string marca;
8     string color;
9 public:
10    coche(){ marca = "sin marca"; color = "sin color";}
11    coche(string _marca, string _color){ marca = _marca; color =
12        _color;}
13
14    //consultores
15    string getmarca() const {return marca;}
16    string getcolor() const {return color;}
17
18    //mofificadores (setters)
19    void setmarca(string _marca){ marca = _marca; }
20    void setcolor(string _color){ color = _color;}
21 };
22 int main()
23 {
24     coche micoche;
25
26     micoche.setcolor("rojo");
27     micoche.setmarca("BMW");
28     cout << micoche.getcolor() << endl;
29     cout << micoche.getmarca() << endl;
```

rojo

BMW

== Code Execution Suc

## Video 8

main.cpp



Share

Run

Output

```
1 //Jabes Estevez
2 using namespace std;
3 #include <iostream>
4 //declaracion de clase
5 class rectangulo{
6     private:
7         // atributos
8         int base, altura;
9     public:
10        // constructor
11        rectangulo(){ base = 10; altura =5; }
12
13        //consultores
14        int getaltura();
15        int getbase() {return base; }
16        int getarea() {return altura * base;}
17
18    };
19
20    // implementacion de metodos de clase
21    inline int rectangulo::getaltura() {return altura;}
22    int main()
23    {
24        rectangulo r;
25        cout << r.getbase();
26    }
```

10

==== Code Execut