

TAREA 2

Inciso 1

Modificar la clase simple del ejemplo añadiendo un atributo de instancia entero llamado ID.

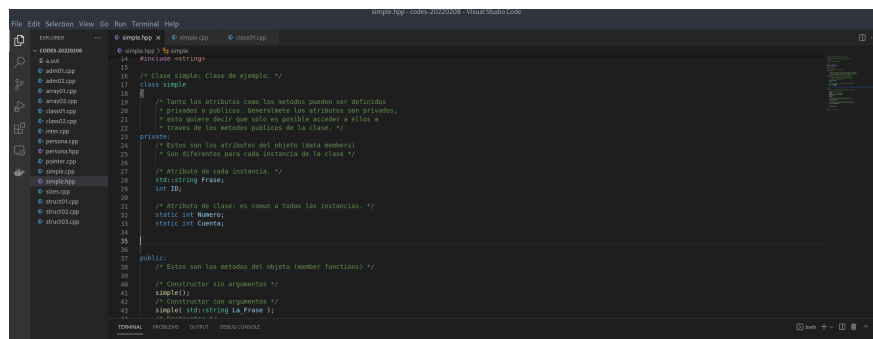


Figura 1: Se agrega el atributo ID

También se agrega un atributo para que no haya problema al borrar una instancia.

Inciso 2

Hacer que este nuevo atributo de instancia ID sea igual al valor del atributo de clase Numero al momento de crear una nueva instancia de la clase.

Se iguala el valor de ID al de la variable que servirá como una variable silenciosa.

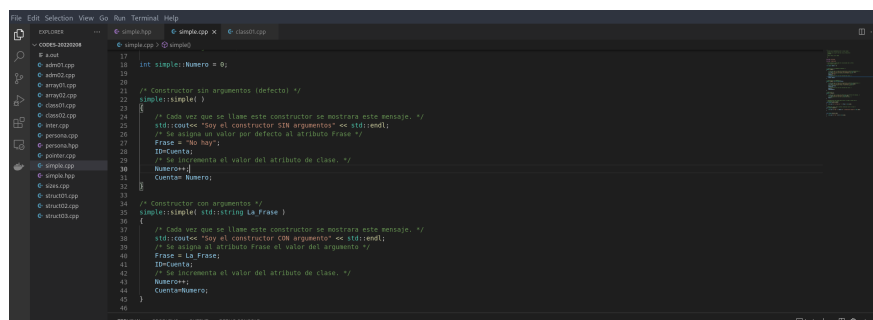


Figura 2: Se iguala el ID al atributo Numero

En el archivo con extensión .hpp se agrega un atributo con `int ID;`

Inciso 3

Agregar un método que muestre el ID de cada instancia de la clase.

Se agrega el nuevo método en el archivo .hpp.

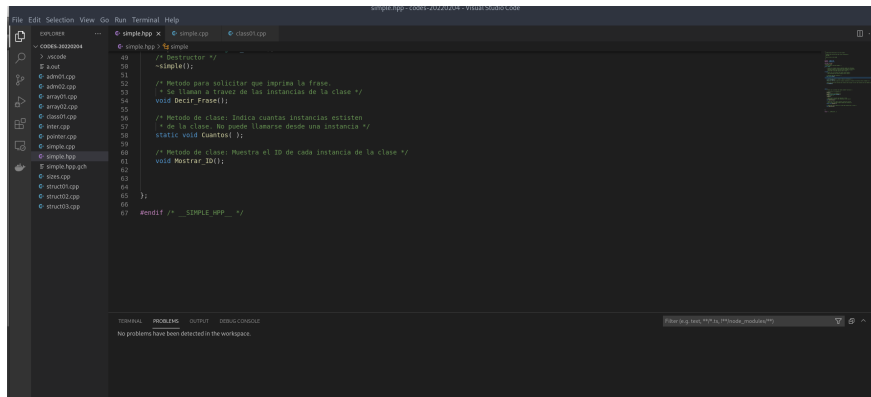


Figura 3: Se crea un nuevo método.

Ahora en el archivo .cpp se especifica que se imprima el ID.

```
1 void simple::Mostrar_ID()
2 {
3     std::cout << "ID:" << ID << std::endl;
4 }
```

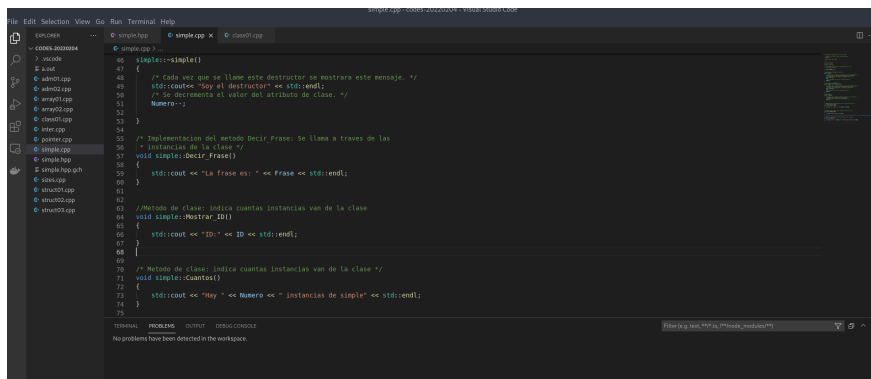


Figura 4: Se indica que se muestre el ID en el nuevo método.

Inciso 4

Modificar el código en class01.cpp para utilizar el nuevo método agregado.

En el documento class01.cpp se llaman el nuevo método para las instancias:

```

48 // Se verifica cuantas instancias van. */
49 simple: {
50   // Se llaman los metodos de los objetos. Estos deben
51   // ser llamados desde una instancia o interactuan
52   // con los atributos de cada instancia. */
53   primero: Mostrar_ID(),
54   segundo: Mostrar_ID(),
55   tercero: Mostrar_ID(),
56 }
57 // Se libera el espacio de memoria creado por new.
58 // El destructor es llamado. */
59 delete tercero;
60
61 // Se verifica cuantas instancias van. */
62 simple: {
63   // Se crea una nueva instancia.
64   simple: {
65     primero: Mostrar_ID(),
66     segundo: Mostrar_ID(),
67     tercero: Mostrar_ID(),
68   }
69 }
70 // Al finalizar la funcion con el comando return
71 // las instancias que son creadas deben ser eliminadas.
72 // los destructores deben ser llamados para cada
73 // instancia. */
74 return 0;
75 }

```

Figura 5: Para las primeras instancias se llaman al método nuevo

```

Soy el constructor CON argumento
Hay 2 instancias de simple
Hay 2 instancias de simple
Soy el constructor CON argumento
Hay 3 instancias de simple
ID:0
ID:1
ID:2
Soy el destructor
Hay 2 instancias de simple
Soy el constructor SIN argumentos
ID:0
ID:1
ID:3
Soy el destructor
Soy el destructor
Soy el destructor
[rubilin@fedora codes-20220208]$ 

```

Figura 6: Salida en consola