

## Ejercicio 1:

Cree una clase **CuentaBancaria** con tres atributos: **numeroCuenta**, **nombreTitular** y **saldo** (un número mayor a cero, no se permite sobregirar la cuenta). Debe asignar valores a los tres atributos en el método **constructor**. Además, agregue los siguientes 5 métodos:

- **getSaldo**: devuelve el **saldo** disponible en la cuenta bancaria.
- **getNumeroCuenta**: devuelve el **numeroCuenta** asociado a la cuenta bancaria.
- **getNombreTitular**: devuelve el **nombreTitular** asociado a la cuenta bancaria.
- **depositar**: recibe el parámetro **importeDeposito**, que deberá sumarse al saldo de la cuenta. El importe deberá ser positivo; de ser negativo, deberá ignorarse.
- **retirar**: recibe el parámetro **importeRetiro**, que deberá restarse del saldo de la cuenta. El importe deberá ser positivo y menor o igual al saldo disponible; de lo contrario, se ignorará.

C:\UFG\Ciclo II 2021\ESTRUCTURA DE DATOS - GRUPO V01\Unidad 3\Semana 1\Tarea\persona.cpp - [Executing] - Dev-C++ 5.11

```

7 class Persona {
8     private:
9         string nombres;
10        string apellidos;
11        int anioNacimiento;
12    public:
13        Persona(string nombres, string apellidos, int anioNacimiento);
14        string nombreCompleto();
15        int edad();
16    };
17
18 Persona::Persona(string nombres, string apellidos, int anioNacimiento) {
19     this->nombres = nombres;
20     this->apellidos = apellidos;
21     this->anioNacimiento = anioNacimiento;
22 }
23
24 string Persona::nombreCompleto(){
25     string nombreCompleto = this->nombres + " " + this->apellidos;
26     return nombreCompleto;
27 }
28
29 int Persona::edad(){
30
31     time_t now = time(0);
32     tm* ltm = localtime(&now);

```

Output:

```

Por favor ingrese sus nombres: Alejandro Leonel
Por favor ingrese sus apellidos: Cruz Carranza
Por favor ingrese su año de nacimiento: 1996
----- Datos de la persona -----
Nombre: Alejandro Leonel Cruz Carranza
Edad: 25
-----
Process exited after 29.4 seconds with return value 0
Presione una tecla para continuar . . .

```

## Ejercicio 2:

Cree una clase **Persona** con tres atributos: **nombres**, **apellidos** y **anioNacimiento**. Debe asignar valores a los tres atributos en el método **constructor**. Además, agregue los siguientes 2 métodos:

- **nombreCompleto**: debe devolver los nombres concatenados con un espacio en blanco y con los apellidos de la persona.
- **edad**: devolverá el resultado de restar el año de nacimiento del año actual (2021). Opcional: investigue cómo se puede obtener el año a partir de la fecha actual en C++.

C:\UFG\Ciclo II 2021\ESTRUCTURA DE DATOS - GRUPO V01\Unidad 3\Semana 1\Tarea\cuentabancaria.cpp - [Executing] - Dev-C++ 5.11

```

1 #include "cuentabancaria.h"
2
3 using namespace std;
4
5 int main() {
6     // Crear una cuenta bancaria
7     CuentaBancaria miCuenta("A-480920", "Jonathan Joestar", 2900.00);
8     cout << "Titular: " << miCuenta.getNombreTitular() << endl;
9     cout << "# Cuenta: " << miCuenta.getNumeroCuenta() << endl;
10    cout << "Saldo: $" << miCuenta.getSaldo() << endl;
11    cout << "-----" << endl;
12
13    // Creando otra cuenta
14    CuentaBancaria cuentaAmigo("C-165426", "Arale Guitierrez", 784661.00);
15    cout << "Titular: " << cuentaAmigo.getNombreTitular() << endl;
16    cout << "# Cuenta: " << cuentaAmigo.getNumeroCuenta() << endl;
17    cout << "Saldo: $" << cuentaAmigo.getSaldo() << endl;
18    cout << "-----" << endl;
19
20    // Creando otra cuenta
21    CuentaBancaria cuentaNoValida("A-354698", "John Cena", -500.00);
22    cout << "Titular: " << cuentaNoValida.getNombreTitular() << endl;
23    cout << "# Cuenta: " << cuentaNoValida.getNumeroCuenta() << endl;
24    cout << "Saldo: $" << cuentaNoValida.getSaldo() << endl;
25    cout << "-----" << endl;
26
27    // Hacer un retiro de la cuenta miCuenta
28    cout << "Quiero retirar $84 de la cuenta " << miCuenta.getNumeroCuenta() << endl;
29    cout << "Saldo inicial: $" << miCuenta.getSaldo() << endl;

```

Output:

```

Titular: Jonathan Joestar
# Cuenta: A-480920
Saldo: $2900
-----
Titular: Arale Guitierrez
# Cuenta: C-165426
Saldo: $ 784661
-----
Titular: John Cena
# Cuenta: A-354698
Saldo: $0
-----
Quiero retirar $84 de la cuenta A-480920
Saldo inicial: $ 2900
Saldo luego de la operacion: $2816
-----
Quiero retirar $3500 de la cuenta A-480920
Saldo inicial: $2816
! - Importe no valido
Saldo luego de la operacion: $2816
-----
Quiero retirar -$725 de la cuenta A-480920
Saldo inicial: $2816
! - Importe no valido
Saldo luego de la operacion: $2816
-----
Quiero depositar $125000 a la cuenta A-480920
Saldo inicial: $2816
Saldo luego de la operacion: $2816
-----

```

**Ejercicio 3:**

Cree una clase **Personaje** con dos atributos: **nombre** (cadena) y **puntosVida** (un número positivo). Ambos atributos deben llenarse con un **constructor**. Además, agregue los siguientes 5 métodos:

- **recibirGolpe**: recibe el parámetro **puntosPerdidos**, que representa la cantidad de puntos a sustraer de los **puntosVida** como resultado de que el personaje reciba un golpe sin defenderse.
- **estaVivo**: devolverá **true** si los puntos de vida son mayores a cero y **false**, en caso contrario.
- **getPuntosVida**: devuelve el valor del atributo privado **puntosVida**
- **getNombre**: devuelve el valor del atributo privado **nombre**
- **comer**: es lo opuesto a **recibirGolpe**, suma el parámetro **puntosGanados** a los puntos de vida del personaje, como resultado de alimentarse, si el personaje está vivo; de lo contrario, no tiene ningún efecto (no se puede “revivir” a un personaje fallecido ☹).

The screenshot shows a C++ IDE with two windows. The left window displays the source code for `Ejercicio3_ClasesObjetos_Semana1.cpp`, and the right window shows the output of the compiled program.

**Source Code (Ejercicio3\_ClasesObjetos\_Semana1.cpp):**

```

7  class Personaje{
8
9      private:
10         int puntosVida;
11         string nombre;
12
13     public:
14         Personaje(int puntosV, string nom);
15
16         void recibirGolpe(int puntosPerdidos);
17         string estaVivo();
18         int getPuntosVida();
19         string getNombre();
20         void comer(int puntosGanados);
21
22     };
23
24     // Implementacion Constructor
25     Personaje::Personaje(int puntosV, string nom){
26         this->puntosVida = puntosV;
27         this->nombre = nom;
28     }

```

**Output (CAUFG\Ciclo II 2021\ESTRUCTURA DE DATOS - GRUPO V01\Unidad 3\Semana 1\Tarea\Ejercicio3\_ClasesObjetos\_Semana1.exe):**

```

++++ Bienvenid@ al juego de Personaje
Ingrese la Vida que el personaje tendra
225
-----
Ingrese los golpes recibidos
25
-----
Ingrese los puntos en Comida
75
Nombre del personaje Momo: Alejandro Leonel Cruz Carranza
Puntos de Vida Inicial personaje Momo: 225
El personaje Momo recibio 25 golpes, bajando su vida a: 200 puntos
El personaje Momo gano 75 comidas, subiendo su vida a: 275 puntos
El personaje tiene un status TRUE
-----
Process exited after 21.86 seconds with return value 0
Presione una tecla para continuar . . .

```

**Compiler Output:**

```

- Errors: 0
- Warnings: 0
- Output Filename: C:\UFG\Ciclo II 2021\ESTRUCTURA DE DATOS - GRUPO V01\Unidad 3\Semana 1\Tarea\Ejercicio3_ClasesObjetos_Semana1.exe
- Output Size: 1.83873558044434 MiB
- Compilation Time: 0.74s

```