

Tutoria Programacion I

Fecha: 05/10/2022

Tutora: Karla Verónica Quirós Delgado

Tema: Herencia

La herencia esta basada en realizar clases de las cuales se creen objetos que hereden características de otros, es una relación fuerte en la programación orientada a objetos, nos permite reutilizar código. Ya que vamos a tener una clase “padre o madre” que va a heredar sus características y funcionalidades a sus “hijos” que serán otras clases que comparten características iguales. De esta forma no tendremos que crear varias clases con las mismas cualidades o funciones de forma separada y es ahí donde reutilizamos código gracias a esta relación de herencia.

En el libro de programación I del profesor Caamaño se menciona un ejemplo de “el factor común entre clases” relacionando la idea como en las matemáticas reducimos expresiones sacando un factor común eso mismo haremos, pero con clases en programación haciendo que la clase padre sea el factor común de las clases hijas que lo deriven.

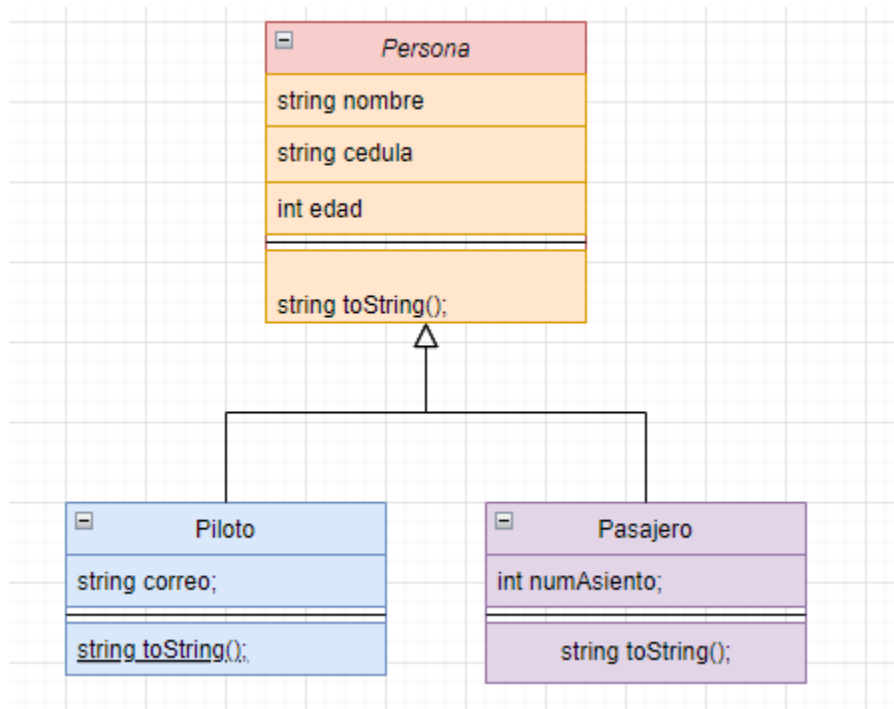
La clase padre se reconoce como “super clase” y sus hijos como “subclases”.

Cuando tenemos una clase que deriva de otra debemos declarar su super clase seguidamente del nombre de la clase derivada, ejemplo tenemos una clase empleado que hereda de la clase Persona, entonces al crear la clase empleado tendremos:

```
1. Class Persona {
2.   // protected no puede ser accedido por el usuario, pero si por sus clases derivadas(hijas)
3.   Protected:
4.   String nombre;
5.   Int edad;
6.   String cedula;
7.   Public:
8.   Persona(string,int,string);// recibe nombre, edad, cedula
9.   ~Persona();
10.  String toString();
11. }
```

```
1. Class Empleado: public Persona { //definimos de cual super clase hereda
2.   Private:
3.   Double salario; // empleado comparte hereda los atributos de persona, pero además tiene un
   salario
4.   Public:
5.   // empleado necesita recibir los atributos de la super clase y además su atributo salario
6.
7.   Empleado (string nombre, int edad, string cedula, double salario):
   Persona(nombre,edad,cedula),salario(salario){ //le entregamos sus atributos a la super clase
8.   }
9.   ~Empleado();
10.  };
11. }
```

Uml:



Nota:

