Tarea 3

1.

Clase base: Una clase base es aquella que no dependen ninguno de sus atributos u objetos de la clase de alguna otra clase, se podría decir que en términos de herencia, sería la clase padre, la clase que se mantiene fija, en el aspecto de herencia. Es también por así llamarlo la clase principal de un programa, sería la clase primaria sin incluir la clase main en donde se corre todo el programa en sí.

Clase derivada: son clases que dependen de las clases bases, ya que algunos de sus métodos son también heredados, y muchas veces, el compilador arrojara malos resultados, ya que al ser dependientes estas clases, a veces podrán generar errores lógicos.

3.1 clase base class Vector2d

{

public int x, y;

public Vector2d(int x, int y)

{

this.x = x; this.y = y;

}

public override string ToString()

{

return String.Format("{0},{1}", x, y);

}

}

3.2 clase derivada

abstract class Figura

{

public Vector2d position;

public string fill, border;

//Constructor por defecto

public Figura() : this(new Vector2d(100, 100))

{

}

//constructor de figura

public Figura(Vector2d pos)

{

position = pos;

fill = "white";

border = "black";

}

public abstract void Dibuja();

}

4.

Herencia simple

La herencia simple consiste en cuando una clase, hereda a una clase hijo, y a solo una le hereda sus atributos, es igual al concepto general de herencia, con la limitante de solo poder heredar de una clase padre a una clase hijo, y solo a una clase hijo.

Herencia multiple.

Consiste en la utilización de las propiedades de una clase a varias clases mas, lo que significa que en esta propiedad una sola clase padre puede heredarle atributos, u objetos de esta a varias clases hijo sin ninguna limitación entre ellas.

¿en c# se puede hacer herencia multiple?

Únicamente hay herencia múltiple de interfaz, pero no de clase. Esto implica que puedes heredar las declaraciones, pero no las implementaciones.

Tarea 4

1.1

a = 10; y b = 15;

1.2

a=10; y b=12

1.3

public virtual string Display()

{

return a.ToString();

}

2.2 (D) no esta definido

2.3 la clase base porque no depende de otras clases o atributos