ISIL APRENDE HACIENDO

TITULO

PROGRAMACION ORIENTADA A OBJETOS

Semana 14

POLIMORFISMO

OBJETIVOS

- Entender el concepto de polimorfismo .
- Aplicar la programación haciendo uso de polimorfismo .

AGENDA

- Definición de Polimorfismo.
- Sobrecarga de métodos.
- Sobrecarga de Constructores.
- Jerarquía de Clases.
- Sobre escritura de métodos.

POLIMORFISMO

Definición

- Es una de las características mas importantes de la programación orientada a objetos constituye un mecanismo que permite implementar métodos en forma general.

- Esta característica denominada Polimorfismo funciona con las referencias a las jerarquías de Clases.

- El polimorfismo permite diseñar e implementar sistemas que fácilmente se puedan extender.

Polimorfismo

Sobrecarga de métodos

La sobrecarga de métodos se da cuando en una clase se tiene 2 o mas métodos con el mismo nombre.

•Los métodos se diferencian por la cantidad, tipo o orden de sus parámetros. Todo esto constituye la “La firma del método ”.

Ejemplo :

```
public class   Operación{

        public double sumaTiempo(){ … }
        public double sumaTiempo(int  a){ … }
        public double sumaTiempo(double b){ … }
        public double sumaTiempo(int  a , double b){ … }
        public double sumaTiempo(double a, int b){ … }

}
```

Polimorfismo

Sobrecarga de Constructores

•La sobrecarga de métodos constructores se da cuando una clase tiene mas de un método constructor.

•Los constructores se diferencian por cantidad, tipo o orden de parámetros.

Ejemplo : Constructores distintos de la clase Operación.

```
public class   Operación{

        public Operación(){….}
        public Operación(int  a){….}
        public Operación(int a, double b){….}
        public Operación(double  a , int b){….}

      }
```

•Estos métodos permiten instanciar objeos considerando distintos tipos de datos disponibles.

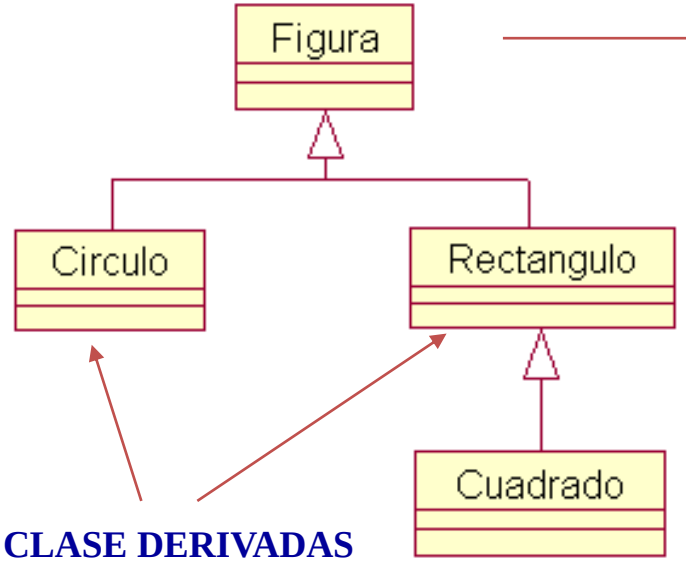
Polimorfismo

Jerarquía de Clases

- Las relaciones de herencia permiten formar Jerarquías de Clases.
- Una Jerarquía de clase es cuando una clase base tiene características comunes que pueden ser aprovechada por diversas clases derivadas.
- En una Jerarquía de clases se puede observar diferentes clases que pueden tener el papel de SuperClase y Clase al mismo tiempo.

Polimorfismo

- Ejemplo : Jerarquía de Herencia




```
Public   class Figura {
     protected double  x, y;
     protected String name;

     public Figura (double x, double y, String name) {
          this.x = x;
          this.y = y;
          this.name = name;
     }
     public final String getName() {
          return name;
     }
      //Método polimorfico
     public static String muestraArea(Figura fig) {
          double area = fig.area();
          return ("Area " + fig.getName() + ": " + area);
     }
}
```

Polimorfismo

Sobre escritura de métodos

La sobre escritura de métodos se da en una jerarquía de herencia, cuándo un método de una clase **derivada** tiene el mismo nombre, los mismos parámetros y el mismo tipo de retorno que un método de la clase **base**, se dice que el método de la clase **derivada** ***sobrescribe*** al método de la clase **base**. Cuándo se llama a un método sobrescrito dentro de una clase **derivada**.

Uso de **super** para acceder a una variable oculta o a un método sobrescrito.

Si desde la clase **derivada** se quiere acceder a una variable oculta de la clase **base** o a un método sobrescrito de la **base** se usa la palabra **super** de la siguiente forma:

super.nombreDeLaVariableOculta

super.nombreDelMetodoSobreescrito (*lista de argumentos*)