

Programación Orientada a Objetos

Dra. Maricela Claudia Bravo
Contreras

mari_clau_18@hotmail.com

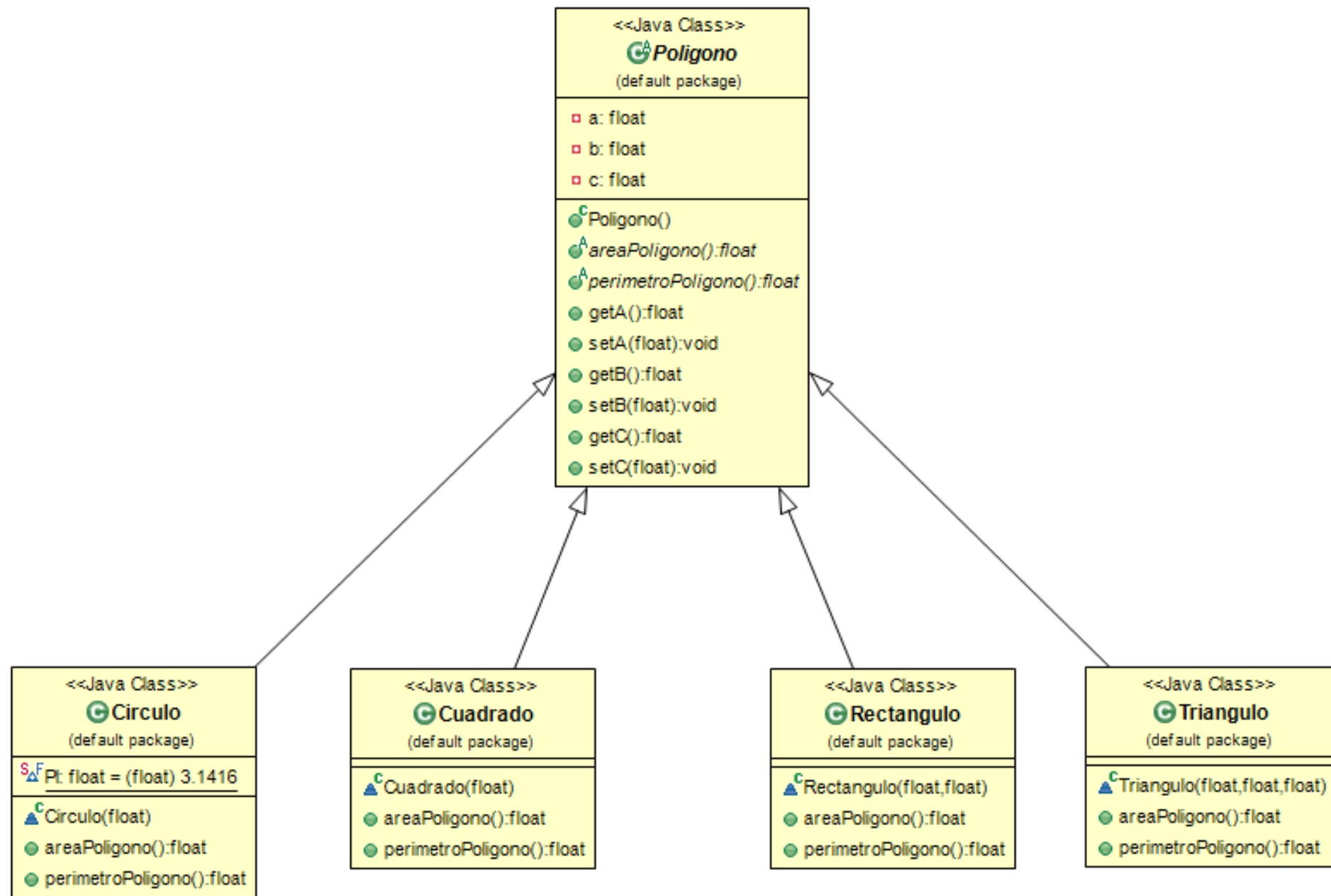
¿Qué es el Polimorfismo?

- ▶ Capacidad que permite a dos clases diferentes responder de forma distinta a un mismo mensaje
- ▶ Esto significa que dos clases que tengan un método con el mismo nombre y que respondan al mismo tipo de mensaje (es decir, que reciban los mismo parámetros), ejecutarán acciones distintas.
- ▶ El polimorfismo se puede implementar de la siguiente forma:
 - ▶ Mediante herencia y sobrescritura de métodos
 - ▶ Mediante clases abstractas y sobrescritura de métodos abstractos
 - ▶ Mediante interfaces y la implementación de métodos abstractos

Clases Abstractas

- ▶ Una clase abstracta es una clase de la que no se pueden crear objetos, pero puede ser utilizada como clase padre para otras clases.
- ▶ Declaración:

```
abstract class NombreClase {  
    .....  
}
```



Ejemplo clase Abstracta Polígono

```
public abstract class Poligono
{
    private float a, b, c;

    public abstract float areaPoligono();
    public abstract float perimetroPoligono();
}
```

```
public class Circulo extends Poligono
{
    static final float PI=(float) 3.1416;
    Circulo(float a)
    {
        this.setA(a);
    }
    @Override
    public float areaPoligono()
    {
        return PI*(this.getA()*this.getA());
    }
    @Override
    public float perimetroPoligono()
    {
        return 2*PI*this.getA();
    }
}
```

```
public class Rectangulo extends Poligono
{
    Rectangulo(float a, float b)
    {
        this.setA(a);
        this.setB(b);
    }
    @Override
    public float areaPoligono()
    {
        return this.getA()*this.getB();
    }
    @Override
    public float perimetroPoligono()
    {
        return (2*this.getA())+(2*this.getB());
    }
}
```

Dudas O comentarios???