UNIVERSIDAD DEL BÍO-BÍO

Paradigmas de la programación

Resumen de clase

UNIVERSIDAD DEL BÍO-BÍO

Variables Primitivas

Tipo	Memoria	Valor por defecto	Valor mínimo	Valor máximo
byte	1 byte – 8 bits	0	-128	127
short	2 byte – 16 bits	0	-32.768	32.767
char	2 byte – 16 bits	‘u0000’	-	-
int	4 byte – 32 bits	0	-2.147.483.648	2.147.483.647
float	4 byte – 32 bits	0.0	±3.40282347E+38F aprox.	
long	8 byte – 64 bits	0	-9.223.372.036.854.775.808	9.223.372.036.854.775.807
double	8 byte – 64 bits	0.0	±1.79769313486231570E+308 aprox.	
boolean	1 byte – 8 bits	false	-	

UNIVERSIDAD DEL BÍO-BÍO

Observación 1

Cuando se realiza una declaración de un float y se asigna un valor decimal, es necesario agregar al final del número una letra ‘f’, la cual indicará que la asignación corresponde a un flotante y no a un **double**.

```
float a = 5.7;
long b = 123456789101112;
```

UNIVERSIDAD DEL BÍO-BÍO

Observación 2

Cuando se realiza una declaración de un long y se asigna un valor superior a lo que puede almacenar un **int**, es necesario agregar al final de su del número una letra 'L'.

```
float a = 5.7;
long b = 123456789101112;
```

UNIVERSIDAD DEL BÍO-BÍO

En una clase se puede encontrar

- Atributos
- Métodos
- Constructores

```
public class Circulo
{
    float radio;

    public Circulo(float radio)
    {
        this.radio = radio;
    }

    float obtenerArea()
    {
        return 3.141516f*radio*radio;
    }
}
```

Nombre de la clase

Atributo

Constructor

Método

Creación de Objetos(Instancias)

El operador **new** es el encargado de crear instancias asociadas de una Clase.

```
Circulo circulo = new Circulo(5);
```

UNIVERSIDAD DEL BÍO-BÍO

Constructor

Cuando se instancia un objeto como en la siguiente imagen:

```
Circulo circulo = new Circulo(5);
```

a continuación se llama al constructor declarado dentro de la clase(En el caso que no se haya declarado se llama al constructor por defecto)

UNIVERSIDAD DEL BÍO-BÍO

Constructor

Un constructor, se compone del nombre de la clase, seguido de los parámetros necesarios para instanciar el objeto.

Obs 1: La cantidad de parámetros pueden ser tantos como sean necesarios.

Obs 2: Un constructor no dispone de datos de retorno como un método.

```
public Circulo(float radio)
{
    this.radio = radio;
}
```

UNIVERSIDAD DEL BÍO-BÍO

Operador **this**

Se ocupa para hacer referencia a un método o un atributo dentro de la clase. También es utilizado para eliminar ambigüedades en el caso que una variable local se llame igual que un atributo.

```
float radio;

public Circulo(float radio)
{
    this.radio = radio;
}
```

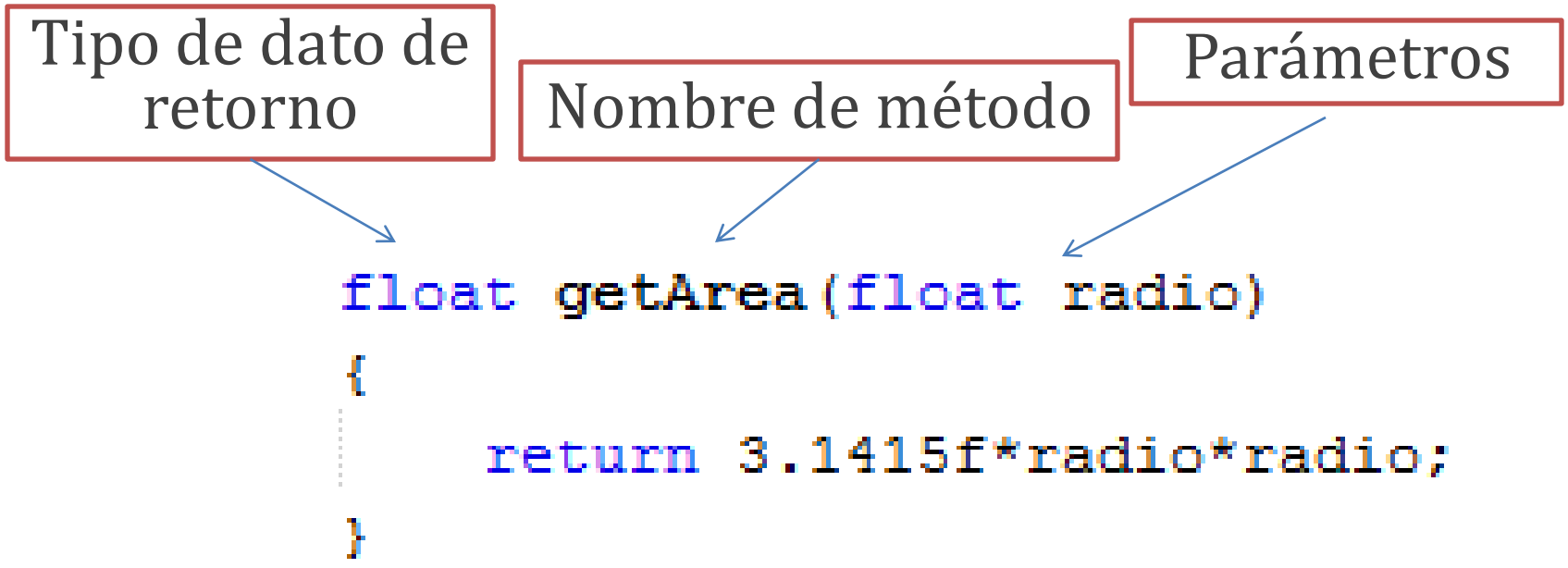
Método

Conjunto de instrucciones definidas dentro de una clase, que realizan una determinada tarea y a las que podemos invocar mediante un nombre.

```
float getArea(float radio)
{
    return 3.1415f*radio*radio;
}
```

UNIVERSIDAD DEL BÍO-BÍO

Método (Estructura)



Método (Uso)

```
public class Circulo
{
    float getArea(float radio)
    {
        return 3.1415f*radio*radio;
    }
}
```

Para utilizar un método de una clase se necesita instanciar un objeto y posteriormente invocar dicho método.

Obs.: El método getArea pide un dato de tipo float para poder ser ejecutado.

```
Circulo instancia = new Circulo();
instancia.getArea(5);
```

UNIVERSIDAD DEL BÍO-BÍO

Método (Uso)

```
public class Circulo {

    public float getArea(float radio) {
        return PI() * dobleValor(radio);
    }

    private float PI() {
        return 3.1415f;
    }

    private float dobleValor(float valor) {
        return valor * valor;
    }

}
```

Los métodos propios de una clase también pueden interactuar entre ellos invocándose directamente por su nombre.

Obs.: Al igual que los atributos en que se puede usar el operador **this** para llamarlos, también se podrá utilizar en los propios métodos(Dentro de la clase).

Ejemplo:

this.PI();

this.dobleValor(15f);

UNIVERSIDAD DEL BÍO-BÍO

Modificadores de acceso

Los modificadores de acceso, como su nombre indica, determinan desde qué clases se puede acceder a un determinado elemento. Java tiene cuatro tipos:

1. **public**
2. **private**
3. **protected**
4. **Por defecto** (no tiene ninguna palabra clave asociada, pero se suele conocer como default)

UNIVERSIDAD DEL BÍO-BÍO

Modificadores de acceso

Modificador	Misma clase	Mismo Package	Subclase	Distinto Package
private	Sí	No	No	No
default	Sí	Sí	No	No
protected	Sí	Sí	Sí/No	No
public	Sí	Sí	Sí	Sí

UNIVERSIDAD DEL BÍO-BÍO

Encapsulamiento

Oculta el estado de los datos miembros de un objeto, de forma que solamente sea posible modificarlos mediante los métodos definidos para dicho objeto. De esta forma, los detalles de implementación permanecen "ocultos" a las personas que usan las clases, evitando así modificaciones o accesos indebidos a los datos que almacenan las instancias.

Encapsulamiento

En el lenguaje de programación Java, se recomienda como buena práctica restringir el acceso de los atributos a privado, pudiendo ser modificados solamente a través de sus métodos asociados.

Obs: Por convención los métodos se escriben en lower camel case, comienza el nombre en minúscula y la siguiente palabra en mayúscula.

Obs: Para actualizar un valor de un atributo u obtener dicho valor, los métodos asociados se indican con el prefijo set/get seguido del nombre del atributo.

```
public class Perro
{
    private String raza;
    private float peso;

    public Perro(String raza, float peso)
    {
        this.raza = raza;
        this.peso = peso;
    }

    public String getRaza() {
        return raza;
    }

    public void setRaza(String raza) {
        this.raza = raza;
    }

    public float getPeso() {
        return peso;
    }

    public void setPeso(float peso) {
        this.peso = peso;
    }
}
```

UNIVERSIDAD DEL BÍO-BÍO

Paquetes de Java y API(Package)

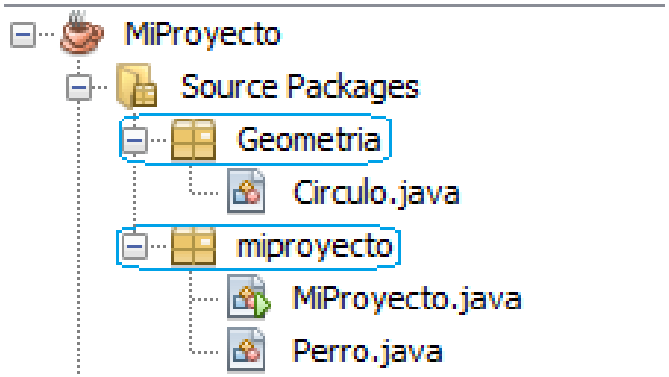
Un paquete en Java se utiliza para agrupar clases relacionadas. Se debe pensar como una carpeta en un directorio de archivos. Usamos paquetes para evitar conflictos de nombre y para escribir un mejor código el cual sea mantenible.
Los paquetes se dividen en dos categorías:

- Paquetes incorporados (paquetes de la API de Java)
- Paquetes definidos por el usuario (crea tus propios paquetes)

UNIVERSIDAD DEL BÍO-BÍO

Paquetes de Java(Package)

Para usar la clase Circulo en MiProyecto.java, se deberá importar(Usando la palabra clave import) indicando el nombre del package que proviene , esta declaración se hace entre medio del package y la declaración de clase(class NombreClase).




```
package miproyecto;

import Geometria.Circulo;

public class MiProyecto {
```