

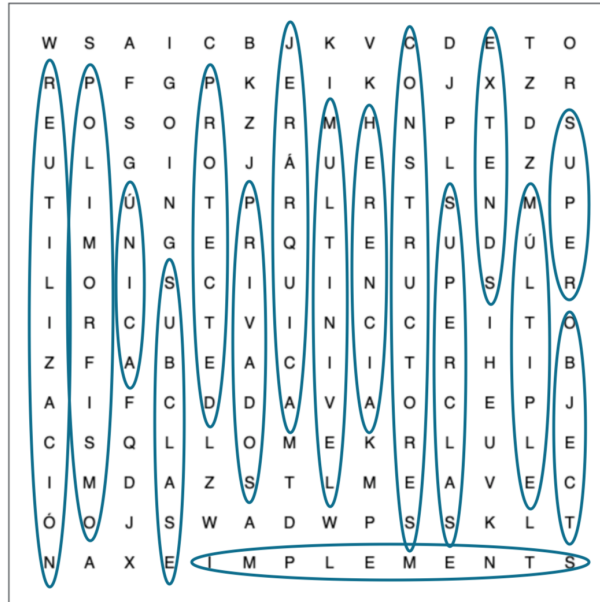
## Tarea 6: Herencia y polimorfismo

Sánchez Segura Cristian Alonso (314217547)

20 de mayo de 2020

## 1. Sopita de letras

Encuentra en la sopa de letras las palabras que completan las siguientes frases:  
(Márcalas en la sopa de letra y completa las frases)



1. Las Clases Padres también son llamadas “Clase base” o **superclase**.
2. Las Clases Hijas también son llamadas “Clase derivada” o **subclase**.
3. La herencia respalda el concepto de **reutilización**. Al heredar, estamos reutilizando los campos/atributos y métodos de la clase existente.
4. Java soporta la herencia permitiendo una clase a incorporar otra clase en su declaración. Esto se hace mediante el uso de la palabra clave **extends**.
5. Existen 3 tipos de herencia a través de clases: **única, multinivel y jerárquica**.
6. La clase **Object**. no tiene superclase, toda clase es implícitamente una subclase de la clase **Object**.
7. Una subclase sólo puede tener una superclase. Esto se debe a que Java no admite herencia **múltiple** con clases.
8. Los **constructores** y métodos/atributos **protected** no son heredados a las subclases, pero el constructor de la superclase puede invocarse desde la subclase y acceder a dichos atributos por medio de setters y getters.
9. El **polimorfismo** en Java y en la POO se refiere a que un objeto puede tomar diferentes formas de comportarse, es decir, que las subclases de una clase pueden definir su propio comportamiento.
10. Qué pasa si queremos por ejemplo hacer referencia al método constructor del padre o a algún atributo del padre si tal vez dicho atributo se llama igual que el atributo del hijo, habría una confusión. Para eso está la palabra clave **super** con la cual podemos hacer referencia a métodos y atributos de la clase padre.

## 2. Código

### 2.1. Juego de rol

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package javaapplication10;

import java.util.Random;
import java.util.Scanner;
import java.util.InputMismatchException;

/**
 *
 * Juego de rol.
 *
 * @author Cristian Alonso Sanchez Segura
 * @version 1.0
 */
public class Jugar {

    public static void main(String[] args) {

        Scanner sn = new Scanner(System.in);
        Random r = new Random();
        int opcion;
        int personaje;
        String nombre;
        boolean salir = false;
        int computadora;

        System.out.println("Bienvenido al juego de rol League Buzz");
        while (!salir) {

            System.out.println("\n1. Jugar" + "\n2. Instrucciones"
                               + "\n3. Personajes" + "\n0. Salir");

            try {

                System.out.println("Escribe una de las opciones");
                opcion = sn.nextInt();

                switch (opcion) {
                    case 1:
                        Personaje usuario = new Personaje("");
                        System.out.println("Comienza el juego.\nSelecciona tu personaje favorito.\n1. Guerrero\n2. Mago\n3. Asesino\n4. ");
                    case 2:
                        System.out.println("Instrucciones");
                    case 3:
                        System.out.println("Personajes");
                    case 0:
                        salir = true;
                }
            } catch (InputMismatchException e) {
                System.out.println("Error: No es un número válido.");
            }
        }
    }
}
```

```
try {
    personaje = sn.nextInt();
    switch (personaje) {
        case 1:
            System.out.println(" Seleccione Guerrero.\nA");
            nombre = sn.next();
            usuario = new Guerrero(nombre);
            break;
        case 2:
            System.out.println(" Seleccione Mago.\nA");
            nombre = sn.next();
            usuario = new Mago(nombre);
            break;
        case 3:
            System.out.println(" Seleccione Asesino.\nA");
            nombre = sn.next();
            usuario = new Asesino(nombre);
            break;
        case 4:
            System.out.println(" Seleccione Tirador.\nA");
            nombre = sn.next();
            usuario = new Tirador(nombre);
            break;
        default:
            System.out.println(" Solo 1 a 4");
    }
} catch (InputMismatchException e) {
    System.out.println(" Debe insertar un nmero");
    sn.next();
}
Personaje cpu = new Personaje("");
computadora = r.nextInt(3);
switch (computadora) {
    case 0:
        cpu = new Guerrero(" Camille");
        break;
    case 1:
        cpu = new Mago(" Veigar");
        break;
    case 2:
        cpu = new Asesino(" Zed");
        break;
    case 3:
        cpu = new Tirador(" Kaisa");
        break;
}
System.out.println("\n Te enfrentarás contra:\n" + cpu);
for (int i = 0; i < 200; i++) {
    cpu.setVida(usuario.atacar());
}
```

```
        usuario.recuperar();
        if (usuario.getVida() <= 0) {
            System.out.println("\nPerdiste , el ganador es: " +
                                "break;
        } else if (cpu.getVida() <= 0) {
            System.out.println("\n Ganaste!:\n" + usuario);
            break;
        } else {
            usuario.setVida(cpu.atacar());
            cpu.recuperar();
        }
    }

    break;
case 2:
    System.out.println("Instrucciones:\nEl juego se desarrolla
        + " humano-maquina , escoge tu personaje favorito , "
        + "enfrentate contra otro personaje aleatorio por
        + " de la computadora y vence. \nLos ataques son au
        + " todos los personajes inician con 100 pts de vida
        + " pierde el primero en llegar a 0 puntos de vida

    break;
case 3:
    System.out.println("Personajes:\n*Guerrero: Ideal para enfr
        + " recupera vida entre mas peleas. \n*Mago: Elimina
        + " con poderosos hechizos. \n*Asesino: Se implacab
        + " antes de que puedan verte. \n*Tirador: Mantén la
        + " a tus enemigos con rafagas a distancia.");

    break;
case 0:
    salir = true;
    break;
default:
    System.out.println("Solo n meros entre 0 y 3");
}
} catch (InputMismatchException e) {
    System.out.println("Debes insertar un n mero");
    sn.next();
}
}
}
}
```