

# Arena USAC



**Universidad San Carlos de Guatemala**

**Facultad de ingeniería.**

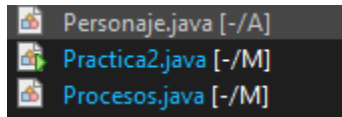
**Ingeniería en ciencias y sistemas**

**Javier Antonio Barrios Calderón**

**202308325**

# Manual de técnico

Para este programa se trabajo con hilos para realizar diferentes procesos siendo tres archivos donde se ejecuta cada proceso.



En el apartado de practica2 es donde se muestran los menús en la consola y la información necesaria para varias funciones la cual se envía a el aparatado de procesos.

```
public static void menu(){
    System.out.println("\n-----Menu de seleccion escoja una opcion porfavor-----");
    System.out.println("Crear personaje=1");
    System.out.println("Modificar personaje=2");
    System.out.println("Eliminar personaje=3");
    System.out.println("Ver lista de personajes=4");
    System.out.println("Batalla=5");
    System.out.println("ver historial de batallas=6");
    System.out.println("Buscar personaje por nombre=7");
    System.out.println("Guardar o caragar partida=8");
    System.out.println("Salir=9");
    System.out.print("Ingrese la opcion que quiere realizar: ");
}

boolean val=false;
while(!val){
    menu();
    int opc=validaciones();
    switch(opc){
        case 1:
            agregarp();
            break;
        case 2:
            buscaredit();
            break;
        case 3:
            buscarelminar();
            break;
        case 4:
            verp();
            break;
        case 5:
            Buscarbatalla();
            break;
        case 6:
            Procesos.bitacora();
            break;
        case 7:
            buscarpornombre();
            break;
        case 8:
            Guardarcargaestado();
            break;
        case 9:
            val=true;
            System.out.println("Javier Antonio Barrios Calderon 20238325");
            System.out.println("Saliendo del menu");
            break;
        default:
            System.out.println("****Opcion invalida seleccione una opcion del menu porfavor****");
    }
}
```

```

public static void agregar() {
    System.out.println("\n-----Creacion de personajes-----");
    System.out.print("Ingrese nombre de su personaje:");
    String nombre=scanner.nextLine();
    if (procesos.valnombrep(nombre)) {
        System.out.println("***ERROR: Ya hay un personaje registrado con ese nombre elija uno");
        return;
    }
    if(procesos.vallet(nombre)) {
        System.out.println("***ERROR: solamente puede tener letras el nombre***");
        return;
    }
    System.out.print("ingrese el arma/objeto del personaje: ");
    String objeto=scanner.nextLine();

    int HP = valornum("Ingrese la cantidad de HP del personaje 100-500: ",100,500);
    int ataque = valornum("Ingrese el ataque del personajes 10-100: ",10,100);
    int velocidad = valornum("Ingrese la velocidad del personaje 1-10: ",1,10);
    int agilidad = valornum("Ingrese la agilidad del personaje 1-10: ",1,10);
    int defensa = valornum("Ingrese el valor de la defesna 1-50: ",1,50);
    boolean e = procesos.agregar(nombre,objeto,HP,ataque,velocidad,agilidad,defensa);
    if (e) {
        System.out.println("Personaje agregado exitosamente");
    } else {
        System.out.println("ERROR: No se pudo agregar personaje verifique sus datos");
    }
}
}

```

```

public static void buscaedit() {
    System.out.println("\n-----Modificar Personaje-----");
    System.out.print("Ingrese el ID o el nombre del personaje que quiera buscar: ");
    String buscar=scanner.nextLine();
    int val=1;

    try {
        int id = Integer.parseInt(buscar);
        Procesos ver=new Procesos();
        ver.buscarID(val,id);
    } catch (NumberFormatException e) {
        String nombre=buscar;
        Procesos ver=new Procesos();
        ver.buscarnombre(val,nombre);
    }
}

public static void buscarpornombre() {
    System.out.println("\n-----bucar Personaje-----");
    System.out.print("Ingrese el nombre del personaje que quiera buscar: ");
    String buscar=scanner.nextLine();
    int val=3;
    if(buscar.matches("[a-zA-Z]+")) {
        String nombre=buscar;
        Procesos ver=new Procesos();
        ver.buscarnombre(val,nombre);
    }else{
        System.out.println("***ERROR:Ingrese un valor valido***");
        return;
    }
}

```

En el apartado de procesos es donde se ejecutan los procesos primordiales de la practica ya que aquí es donde se almacenan la información de los personajes creados, se realizan las búsquedas por ID o nombre de los personajes, creación de archivos de guardado y la capacidad de cargar partidas. Prácticamente en este apartado es donde se ejecuta todo la practica 2 esto se realizo de esta manera para poder mantener un mayor orden dentro de la práctica ya que muchas veces se llaman las mismas funciones para varias situaciones.

```

public boolean valnombrep(String nombrep) {
    for (int i = 0; i < Cantidadpersonajes; i++) {
        if (nombres[i].equalsIgnoreCase(nombrep)) {
            return true;
        }
    }
    return false;
}

public boolean vallet(String nombrep){
    if(!nombrep.isEmpty() && nombrep.matches("[a-zA-ZáéíóúÁÉÍÓÚÑ ]+")){
        return false;
    }
    return true;
}

public boolean agregarp(String nombre, String objeto, int HP, int ataque, int velocidad, int agilidad, int defensa){
    nombres[Cantidadpersonajes]=nombre;
    objetos[Cantidadpersonajes]=objeto;
    healthpoints[Cantidadpersonajes]=HP;
    ataques[Cantidadpersonajes]=ataque;
    velocidades[Cantidadpersonajes]=velocidad;
    agilidades[Cantidadpersonajes]=agilidad;
    defensas[Cantidadpersonajes]=defensa;
    Batallasganadas[Cantidadpersonajes]=0;
    Batallasperdidas[Cantidadpersonajes]=0;
    Cantidadpersonajes++;
    System.out.println("Personaje agregado con No. ID: "+Cantidadpersonajes);
    return true;
}

```

```

public void lista(){
    if(Cantidadpersonajes==0){
        System.out.print("***No hay personajes creados***");
        return;
    }
    System.out.println("***Listado de los personajes***");
    System.out.printf("%-5s %-20s %-20s %-10s %-10s %-10s %-10s %-10s %-20s %-20s\n","ID", "Nombre", "objeto", "HP","ataque","velocidad","agilidad","defensa","Batallasganadas","Batallasperdidas");
    for (int a = 0; a < Cantidadpersonajes; a++) {
        System.out.printf("%-5s %-20s %-20s %-10s %-10s %-10s %-10s %-10s %-20s %-20s\n", (a + 1), nombres[a],objetos[a],healthpoints[a],ataques[a],velocidades[a],agilidades[a],defensas[a],Batallasganadas[a],Batallasperdidas[a]);
    }
}

public String buscarID(int Val,int ID){
    int lista = ID-1;
    int validacion=Val;
    if(nombres[lista]==null){
        System.out.println("***Personaje no encontrado por ID revise que ingreso correctamente los datos***");
    }else{
        System.out.println("***Personaje seleccionado***");
        System.out.printf("%-5s %-20s %-20s %-10s %-10s %-10s %-10s %-10s %-20s %-20s\n","ID", "Nombre", "objeto", "HP","ataque","velocidad","agilidad","defensa","Batallasganadas","Batallasperdidas");
        System.out.printf("%-5s %-20s %-20s %-10s %-10s %-10s %-10s %-10s %-20s %-20s\n", (ID), nombres[lista],objetos[lista],healthpoints[lista],ataques[lista],velocidades[lista],agilidades[lista],defensas[lista],Batallasganadas[lista],Batallasperdidas[lista]);

        if(validacion==1){
            Practica2.modificar(ID-1);}
        if(validacion==2){
            Practica2.eliminar(ID-1, nombres[lista]);
        }
    }
    return null;
}

```

```

public boolean eliminarp(int elim, String nom){
    int posicion=elim;
    String nome=nom;
    for(int i=posicion; i<Cantidadpersonajes-1; i++){
        nombres[i]=nombres[i+1];
        objetos[i]=objetos[i+1];
        healtpoints[i]=healtpoints[i+1];
        ataques[i]=ataques[i+1];
        velocidades[i]=velocidades[i+1];
        agilidades[i]=agilidades[i+1];
        defensas[i]=defensas[i+1];
        Batallasganadas[i]=Batallasganadas[i+1];
        Batallasperdidas[i]=Batallasganadas[i+1];
    }

    nombres[Cantidadpersonajes]=null;
    objetos[Cantidadpersonajes]=null;
    healtpoints[Cantidadpersonajes]=0;
    ataques[Cantidadpersonajes]=0;
    velocidades[Cantidadpersonajes]=0;
    agilidades[Cantidadpersonajes]=0;
    defensas[Cantidadpersonajes]=0;
    Batallasganadas[Cantidadpersonajes]=0;
    Batallasperdidas[Cantidadpersonajes]=0;
    Cantidadpersonajes --;

    System.out.println("***El personaje "+nome+" se elimino con exito***");
    return true;
}

```

En personaje es donde esta todos los get y set que sirven para poder ejecutar de manera simultanea las batallas que se generan entre los personajes seleccionados también los procesos que se ejecutan dentro de la simulación de la pelea.

```
// Getters y setters
public int getId() {
    return id;
}

public void setID(int id) {
    this.id=id;
}

public String getNombre() {
    return nombre;
}

public void setNombre(String nombre) {
    this.nombre = nombre;
}

public String getArma() {
    return arma;
}

public void setArma(String arma) {
    this.arma = arma;
}

public int getHp() {
    return hp;
}

public void setHp(int hp) {
    this.hp = hp;
    this.hpActual = hp;
}

public int getHpActual() {
    return hpActual;
}
```

```
public int Ataque(){
    if (hp > 0) {
        double dañoBase = (ataque * ataque) / (double)(ataque + defensa);
        double crit = 0.20;

        double aleatorio = Math.random();

        if (aleatorio < crit) {
            double bonus = 1.5 + Math.random();
            dañoBase *= bonus;
            System.out.println(nombre+" realizo un ataque critico");
        }

        return (int) dañoBase;
    } else {
        return 0;
    }
}

public void dañorealizado(int daño){
    this.hp -=daño;
}
```