INSTITUTO TECNOLÓGICO DE TOLUCA

CARRERA: Ingeniería Mecatrónica

PROYECTO: Herencia

FECHA DE ENTREGA: 02 de Octubre del 2024

Tema #3: Herencia

Integrantes de Equipo:

Baltazar Hernández Saul

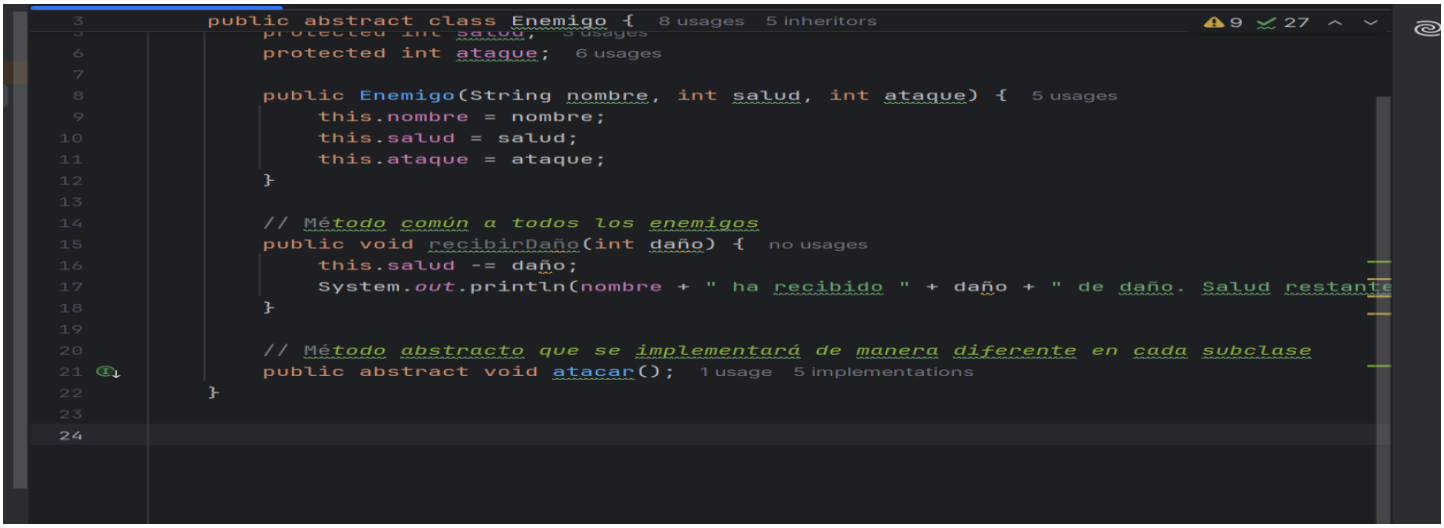
Granados Arias Romel

Josue David Claudio Hernández

Rojas Ortiz Jimena

MATERIA: Programación Avanzada

DOCENTE: Castro Magaña Jesus Aurelio

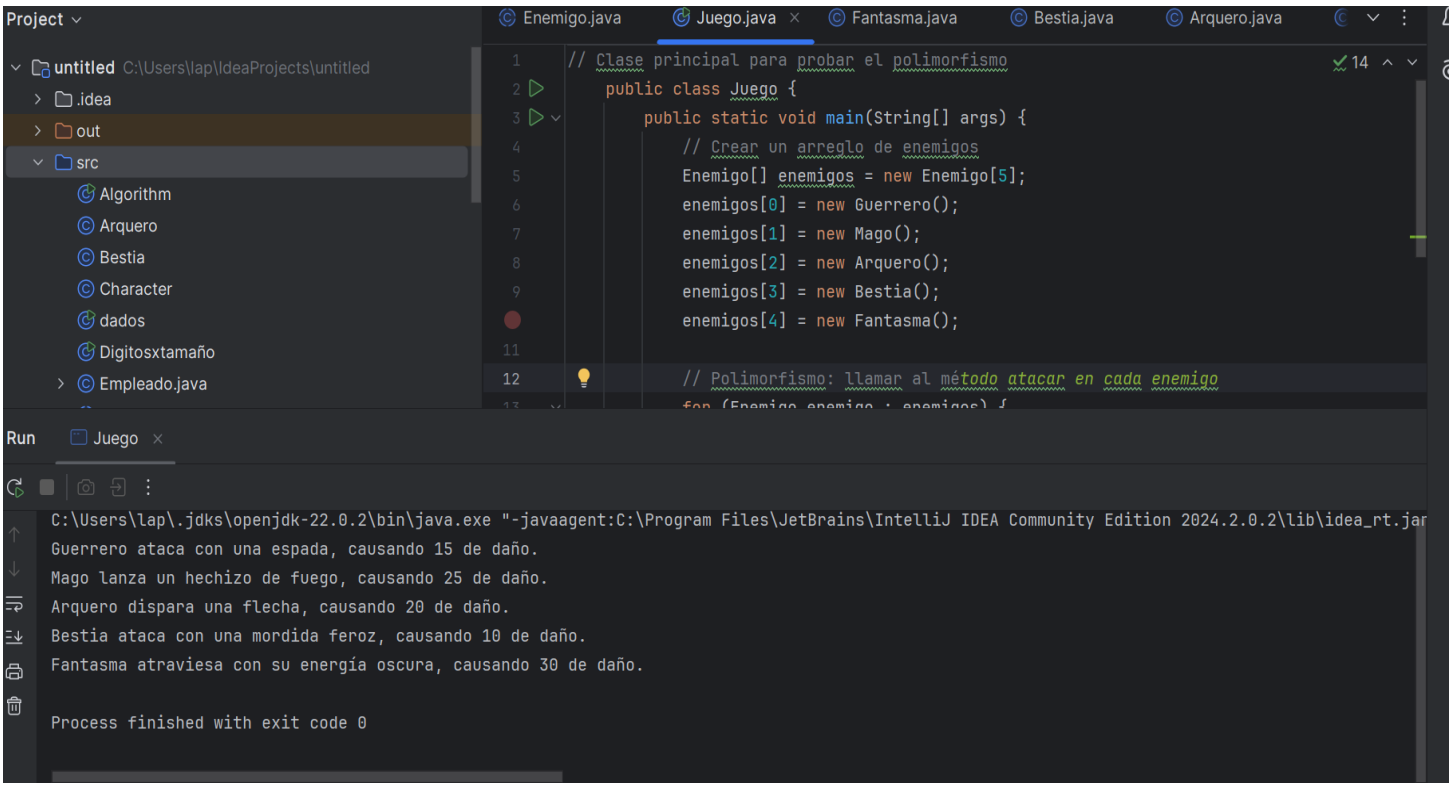


```java
public abstract class Enemigo {  8 usages  5 inheritors
    protected int ataque;  6 usages

    public Enemigo(String nombre, int salud, int ataque) {  5 usages
        this.nombre = nombre;
        this.salud = salud;
        this.ataque = ataque;
    }

    // Método común a todos los enemigos
    public void recibirDaño(int daño) {  no usages
        this.salud -= daño;
        System.out.println(nombre + " ha recibido " + daño + " de daño. Salud restante
    }

    // Método abstracto que se implementará de manera diferente en cada subclase
    public abstract void atacar();  1 usage  5 implementations
}
```

Enemigo.java | Juego.java | Fantasma.java | Bestia.java | Arquero.java

C:\Users\lap\IdeaProjects\untitled\src\Bestia.java

```java
// Clase Bestia, que extiende Enemigo
public class Bestia extends Enemigo {  1 usage
    public Bestia() { super( nombre: "Bestia", salud: 150, ataque: 10); }

    @Override  1 usage
    public void atacar() {
        System.out.println(nombre + " ataca con una mordida feroz, causando " + ataque
    }
}
```



Project

untitled C:\Users\lap\IdeaProjects\untitled

- .idea
- out
- src
  - Algorithm
  - Arquero
  - Bestia
  - Character
  - dados
  - Digitosxtamaño
  - Empleado.java

Enemigo.java | Juego.java | Fantasma.java | Bestia.java | Arquero.java

```java
// Clase principal para probar el polimorfismo
public class Juego {
    public static void main(String[] args) {
        // Crear un arreglo de enemigos
        Enemigo[] enemigos = new Enemigo[5];
        enemigos[0] = new Guerrero();
        enemigos[1] = new Mago();
        enemigos[2] = new Arquero();
        enemigos[3] = new Bestia();
        enemigos[4] = new Fantasma();

        // Polimorfismo: llamar al método atacar en cada enemigo
        for (Enemigo enemigo : enemigos) {
```

Run | Juego

```
C:\Users\lap\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.0.2\lib\idea_rt.jar
Guerrero ataca con una espada, causando 15 de daño.
Mago lanza un hechizo de fuego, causando 25 de daño.
Arquero dispara una flecha, causando 20 de daño.
Bestia ataca con una mordida feroz, causando 10 de daño.
Fantasma atraviesa con su energía oscura, causando 30 de daño.

Process finished with exit code 0
```

CODIGO EN JAVA

```
// Clase base Enemigo
public abstract class Enemigo {
    protected String nombre;
    protected int salud;
    protected int ataque;

    public Enemigo(String nombre, int salud, int ataque) {
        this.nombre = nombre;
        this.salud = salud;
        this.ataque = ataque;
    }

    // Método común a todos los enemigos
    public void recibirDaño(int daño) {
        this.salud -= daño;
        System.out.println(nombre + " ha recibido " + daño + " de daño. Salud restante: " + salud);
    }

    // Método abstracto que se implementará de manera diferente en cada subclase
    public abstract void atacar();
}

// Clase Guerrero, que extiende Enemigo
```

```
public class Guerrero extends Enemigo {
    public Guerrero() {
        super("Guerrero", 100, 15);
    }

    @Override
    public void atacar() {
        System.out.println(nombre + " ataca con una espada, causando " + ataque + " de daño.");
    }
}

// Clase Mago, que extiende Enemigo
public class Mago extends Enemigo {
    public Mago() {
        super("Mago", 70, 25);
    }

    @Override
    public void atacar() {
        System.out.println(nombre + " lanza un hechizo de fuego, causando " + ataque + " de
daño.");
    }
}

// Clase Arquero, que extiende Enemigo
public class Arquero extends Enemigo {
    public Arquero() {
```

```
        super("Arquero", 80, 20);
    }

    @Override
    public void atacar() {
        System.out.println(nombre + " dispara una flecha, causando " + ataque + " de daño.");
    }
}

// Clase Bestia, que extiende Enemigo
public class Bestia extends Enemigo {
    public Bestia() {
        super("Bestia", 150, 10);
    }

    @Override
    public void atacar() {
        System.out.println(nombre + " ataca con una mordida feroz, causando " + ataque + " de daño.");
    }
}

// Clase Fantasma, que extiende Enemigo
public class Fantasma extends Enemigo {
    public Fantasma() {
        super("Fantasma", 50, 30);
    }
```

```
    @Override
    public void atacar() {
        System.out.println(nombre + " atraviesa con su energía oscura, causando " + ataque + " de daño.");
    }
}

// Clase principal para probar el polimorfismo
public class Juego {
    public static void main(String[] args) {
        // Crear un arreglo de enemigos
        Enemigo[] enemigos = new Enemigo[5];
        enemigos[0] = new Guerrero();
        enemigos[1] = new Mago();
        enemigos[2] = new Arquero();
        enemigos[3] = new Bestia();
        enemigos[4] = new Fantasma();

        // Polimorfismo: llamar al método atacar en cada enemigo
        for (Enemigo enemigo : enemigos) {
            enemigo.atacar();
        }
    }
}
```