

PROGRAMACION II

Trabajo Practico 3: Introducción a la POO

Repositorio Github: <https://github.com/AguP10/UTN-TUPaD-P2.git>

1. Registro de Estudiantes

- a. Crear una clase Estudiante con los atributos: nombre, apellido, curso, calificación

```
7  public class Estudiantes {
8
9      public String nombre;
10     public String apellido;
11     public String curso;
12     public double calificacion;
13
14
15     void mostrarInfo() {
16         System.out.println("Alumno: " + nombre + " " + apellido);
17         System.out.println("Curso: " + curso + "\nNota: " + calificacion);
18         System.out.println("");
19     }
20     double subirCalificacion(double puntos) {
21         return calificacion += puntos;
22     }
23     double bajarCalificacion(double puntos) {
24         return calificacion -= puntos;
25     }
26 }
```

Tarea: Instanciar a un estudiante, mostrar su información, aumentar y disminuir calificaciones.

```
7 public class TP3 {
8
9     public static void main(String[] args) {
10         Estudiantes est1 = new Estudiantes();
11         est1.nombre = "Juan";
12         est1.apellido = "Perez";
13         est1.curso = "5 A";
14         est1.calificacion = 8;
15
16         est1.mostrarInfo();
17
18         est1.subirCalificacion(2);
19         est1.mostrarInfo();
20
21         est1.bajarCalificacion(4);
22         est1.mostrarInfo();
23     }
24 }
```

Output - TP3 (run) ×

run:
Alumno: Juan Perez
Curso: 5 A
Nota: 8.0

Alumno: Juan Perez
Curso: 5 A
Nota: 10.0

Alumno: Juan Perez
Curso: 5 A
Nota: 6.0

BUILD SUCCESSFUL (total time: 0 seconds)

2. Registro de Mascotas

- a. Crear una clase Mascota con los atributos: nombre, especie, edad.

```
7      public class Mascotas {
8
9          public String nombre;
10         public String especie;
11         public int edad;
12
13
14         void mostrarInfo() {
15             System.out.println("Nombre: " + nombre);
16             System.out.println("Especie: " + especie);
17             System.out.println("Edad: " + edad);
18             System.out.println("");
19         }
20
21         int cumplirAños(int años) {
22             return edad += años;
23         }
24     }
```

Tarea: Crear una mascota, mostrar su información, simular el paso del tiempo y verificar los cambios.

```
7 public class TP3 {
8
9     public static void main(String[] args) {
10         Mascotas M1 = new Mascotas();
11         M1.nombre = "Bob";
12         M1.especie = "Gato";
13         M1.edad = 3;
14
15
16         M1.mostrarInfo();
17         M1.cumplirAños(2);
18
19         M1.mostrarInfo();
20
21
22     }
23 }
```

Output - TP3 (run) ×

```
run:
Nombre: Bob
Especie: Gato
Edad: 3

Nombre: Bob
Especie: Gato
Edad: 5
```

3. Encapsulamiento con la Clase Libro

- a. Crear una clase Libro con atributos privados: titulo, autor, añoPublicacion.

```
7   public class Libros {
8       private String titulo;
9       private String autor;
10      private int anoPublicacion;
11
12      public String getTitulo() {
13          return titulo;
14      }
15      public void setTitulo(String titulo) {
16          this.titulo = titulo;
17      }
18      public String getAutor() {
19          return autor;
20      }
21      public void setAutor(String autor) {
22          this.autor = autor;
23      }
24      public int getAnoPublicacion() {
25          return anoPublicacion;
26      }
27      public void setAnoPublicacion(int anoPublicacion) {
28          if (anoPublicacion > 0) {
29              this.anoPublicacion = anoPublicacion;
30          } else {
31              System.out.println("Año invalido");
32          }
33      }
34  }
35 }
```

Tarea: Crear un libro, intentar modificar el año con un valor inválido y luego con uno válido, mostrar la información final.

```
7 public class TP3 {
8
9     public static void main(String[] args) {
10         Libros l1 = new Libros();
11         l1.setTitulo("100 años de soledad");
12         l1.setAutor("Garcia Marquez");
13         l1.setAnoPublicacion(-89);
14
15         l1.setAnoPublicacion(1967);
16
17         System.out.println("Titulo: " + l1.getTitulo());
18         System.out.println("Autor: " + l1.getAutor());
19         System.out.println("Año de publicacion: " + l1.getAnoPublicacion());
20
21
22
23
```

Output - TP3 (run) x

```
run:
Año invalido
Titulo: 100 años de soledad
Autor: Garcia Marquez
Año de publicacion: 1967
BUILD SUCCESSFUL (total time: 0 seconds)
```

4. Gestión de Gallinas en Granja Digital

- a. Crear una clase Gallina con los atributos: idGallina, edad, huevosPuestos.

```
7      public class Gallina {
8          public int idGallina;
9          public int edad;
10         public int huevos;
11
12         public void mostrarDatos() {
13             System.out.println("ID: " + idGallina);
14             System.out.println("Edad: " + edad);
15             System.out.println("Huevos puestos: " + huevos);
16             System.out.println(" ");
17         }
18
19         public int ponerHuevo(int huevosPuestos) {
20             return huevos += huevosPuestos;
21         }
22
23         public int envejecer(int años) {
24             return edad += años;
25         }
26     }
```

Tarea: Crear dos gallinas, simular sus acciones (envejecer y poner huevos), y mostrar su estado.

```
public class TP3 {  
8  
9     public static void main(String[] args) {  
10         Gallina gallina1 = new Gallina();  
11         Gallina gallina2 = new Gallina();  
12  
13         gallina1.idGallina = 1;  
14         gallina1.edad = 3;  
15         gallina1.huevos= 0;  
16  
17         gallina2.idGallina = 2;  
18         gallina2.edad = 2;  
19         gallina2.huevos= 0;  
20  
21         gallina1.envejecer(2);  
22         gallina2.envejecer(2);  
23  
24         gallina1.ponerHuevo(10);  
25         gallina2.ponerHuevo(15);  
26  
27         System.out.println("Gallina 1");  
28         gallina1.mostrarDatos();  
29         System.out.println("Gallina 2");  
30         gallina2.mostrarDatos();  
31     }  
32 }  
33 }
```

Output - TP3 (run) X



run:



Gallina 1



ID: 1



Edad: 5

Huevos puestos: 10

Gallina 2

ID: 2

Edad: 4

Huevos puestos: 15

BUILD SUCCESSFUL (total time: 0 seconds)

5. Simulación de Nave Espacial

Crear una clase NaveEspacial con los atributos: nombre, combustible.

```
7 public class NaveEspacial {
8
9     private String nombre;
10    private int combustible;
11
12    public String getNombre() {
13        return nombre;
14    }
15
16    public void setNombre(String nombre) {
17        this.nombre = nombre;
18    }
19
20    public int getCombustible() {
21        return combustible;
22    }
23
24    public void recargarCombustible(int cantidad) {
25        if (this.combustible < 500) {
26            this.combustible += cantidad;
27        } else {
28            System.out.println("Se supera la capacidad maxima");
29        }
30    }
31
32    private boolean despegar() {
33        return this.combustible > 100;
34    }
35
36
37    public int avanzar(int distancia) {
38        if (despegar()) {
39            return distancia;
40        } else {
41            System.out.println("La nave no tiene combustible");
42            return 0;
43        }
44    }
45
46    public void mostrarEstado() {
47        System.out.println("Nave: " + nombre);
48        System.out.println("Combustible: " + combustible);
49        System.out.println("Despegue = " + despegar());
50    }
```

Tarea: Crear una nave con 50 unidades de combustible, intentar avanzar sin recargar, luego recargar y avanzar correctamente. Mostrar el estado al final.

```
7 public class TP3 {
8
9     public static void main(String[] args) {
10         NaveEspacial nave = new NaveEspacial();
11         nave.setNombre("StarShip");
12         nave.recargarCombustible(50);
13         nave.mostrarEstado();
14         nave.avanzar(80);
15         System.out.println("");
16         nave.recargarCombustible(200);
17         nave.avanzar(80);
18         nave.mostrarEstado();
19     }
20 }
```

Output - TP3 (run) x

```
run:
Nave: StarShip
Combustible: 50
Despegue = false
La nave no tiene combustible

Nave: StarShip
Combustible: 250
Despegue = true
BUILD SUCCESSFUL (total time: 0 seconds)
```