



Programación con Java

TAREA 08

Índice.....	1
Introducción.....	2
Programación Orientada a Objetos en Java.....	3
Ejercicio 01:.....	3
Imágenes Ejercicio01:.....	3
Ejercicio 02:.....	4
Imágenes Ejercicio02:.....	4
Ejercicio 03:.....	7
Imágenes Ejercicio 03:.....	7
Ejercicio 04:.....	9
Imágenes Ejercicio04:.....	9
Webgrafía.....	10

Con estos ejercicios aprenderemos a definir clases con sus atributos, las variables que tendrá la clase; constructores, las variables que podremos añadirle al objeto al llamarlo; y métodos, las acciones que harán determinadas llamadas.

Esto nos será útil para una mejor gestión de datos, como definiremos nosotros los datos que entrarán nos dará más control; aparte de poder llamarlo en el mismo archivo o en un archivo a parte.

Programación Orientada a Objetos en Java

Ejercicio 01:

La primera clase que crearemos será una llamada "Persona", en esta deberemos crear seis atributos para definir a esta persona, le daremos: nombre, edad, DNI; sexo, estará definido por H o M; peso y altura. Una vez tengamos los atributos tendremos que darle una opción predeterminada, y estas serán sus valores por defecto, String = "", &c. ; exceptuando al DNI y al sexo, para ellos el sexo será "H" y el DNI estará vacío.

Por último implementaremos tres constructores donde definiremos los valores que tendrán los atributos, según las variables que insertemos en la llamada del objeto podremos dar unos u otros; por defecto al estar vacío todos estarán en sus valores por defecto; podremos definir el nombre, edad, sexo y dejar a los demás en valores por defecto; y por último, en el último constructor definiremos todas las variables.

Imágenes Ejercicio01:

main llamando al objeto y mostrándolo en terminal

```
public static void main(String[] args) {
    Persona persona1 = new Persona();
    Persona persona2 = new Persona(nombre:"Marissa", edad:21, sexo:"M");
    Persona persona3 = new Persona(nombre:"Alfredo", edad:32, sexo:"H", dni:"39256249J", peso:75.2, altura:1.78);

    System.out.println(persona1.toString() + "\n" + persona2.toString() + "\n" + persona3.toString());
}
```

OBJETO

Creación de los atributos

```
public class Persona {
    // ATRIBUTOS
    private String nombre;
    private int edad;
    private String sexo;
    public final String SEXO_HOMBRE = "H";
    private String dni;
    private double peso;
    private double altura;
}
```

Creación de los constructores

```
// CONTRUCTORES
public Persona() {
    this.nombre = "";
    this.edad = 0;
    this.sexo = SEXO_HOMBRE;
    this.peso = 0.0;
    this.altura = 0.0;
}

public Persona(String nombre, int edad, String sexo) {
    this.nombre = nombre;
    this.edad = edad;
    this.sexo = sexo;
    this.peso = 0.0;
    this.altura = 0.0;
}

public Persona(String nombre, int edad, String sexo,
    String dni, double peso, double altura) {
    this.nombre = nombre;
    this.edad = edad;
    this.sexo = sexo;
    this.dni = dni;
    this.peso = peso;
    this.altura = altura;
}
```

Creación del método para hacerlo String

```
// METODOS
@Override
public String toString() {
    return " - Persona:\n\t{" +
        "nombre='" + nombre + '\'' +
        ", edad='" + edad + '\'' +
        ", sexo='" + sexo + '\'' +
        ", dni='" + dni + '\'' +
        ", peso='" + peso + "kg" + '\'' +
        ", altura='" + altura + "m" + '\'' +
        '}';
}
```

Respuesta

```
- Persona:
  {nombre='', edad='0', sexo='H', dni=null', peso=0.0kg', altura=0.0m'}
- Persona:
  {nombre='Marissa', edad='21', sexo='M', dni=null', peso=0.0kg', altura=0.0m'}
- Persona:
  {nombre='Alfredo', edad='32', sexo='H', dni=39256249J', peso=75.2kg', altura=1.78m'}
```

Programación Orientada a Objetos en Java

Ejercicio 02:

La siguiente clase que haremos será para la creación de una contraseña teniendo en cuenta un tamaño, para ello lo haré con JOptionPane.

Deberemos empezar creando dos atributos: longitud y contraseña, por defecto será de ocho caracteres. Después de crear estos haremos los constructores: el primero será la opción por defecto al llamar al objeto sin variables, se generará una contraseña de ocho caracteres de longitud; y al segundo le definiremos con un Integer como parámetro que será el tamaño de la contraseña.

Para la generación de la contraseña utilizaré la anotación ASCII (menos para los símbolos que no están seguidos), aplicando un filtro para tener una contraseña más común: primero generará un número aleatorio entre el 0 y el 10000, cuando el número generado sea divisible entre 13 (número primo con pocos divisores) generará un símbolo, si es divisible entre el primo 7 (que tiene más divisores) será un número, y después de estos filtros si es par será una letra mayúscula y sino minúscula.

Imágenes Ejercicio02:

main

Aquí preguntaremos por la longitud dentro de un try, que está en un do-while para repetirse mientras no insertes un número

```
public static void main(String[] args) {
    boolean mal = false;
    boolean pred = false;
    String contra = "";
    do {
        String input = JOptionPane.showInputDialog("Ingresa la longitud de la contraseña que se generará");
        if (input.isEmpty()) {
            Password password = new Password();
            contra = password.getPass();
            pred = true;
            mal = false;
        } else {
            try {
                int longitud = Integer.parseInt(input);
                Password password = new Password(longitud);
                contra = password.getPass();
                pred = false;
                mal = false;
            } catch (Exception e) {
                JOptionPane.showMessageDialog(null, "INGRESE UN VALOR ENTERO", "Error 404",
                    JOptionPane.ERROR_MESSAGE);
                mal = true;
            }
        }
    } while (mal);
    if (pred) {
        JOptionPane.showMessageDialog(null, "Tu contraseña será:\n" + contra + "\n\nOPCIÓN PREDETERMINADA.",
            "Password", JOptionPane.INFORMATION_MESSAGE);
    } else {
        JOptionPane.showMessageDialog(null,
            "Tu contraseña será:\n" + contra + "\n\nGENERADA ALEATORIAMENTE.",
            "Password", JOptionPane.INFORMATION_MESSAGE);
    }
}
```

Programación Orientada a Objetos en Java

CREACIÓN DEL OBJETO

```
public class Password {  
    // ATRIBUTOS  
    int longitud;  
    final int LONG_PRED = 8;  
    String pass;  
    final String[] simbolos = { "!", "@",  
                                "#", "$", "%", "&", "-",  
                                "+", "=", "/", "<", ">" };  
}
```

Creación de los atributos

```
// CONSTRUCTORES  
public Password() {  
    this.longitud = LONG_PRED;  
  
    Random numAleatorio = new Random();  
    StringBuilder pass = new StringBuilder();  
    filtroGeneracion(simbolos, pass, longitud, numAleatorio);  
    System.out.println(pass.toString()); // Mostrar en consola  
    this.pass = pass.toString();  
}  
  
public Password(int longitud) {  
    this.longitud = longitud;  
  
    Random numAleatorio = new Random();  
    StringBuilder pass = new StringBuilder();  
    filtroGeneracion(simbolos, pass, longitud, numAleatorio);  
    System.out.println(pass.toString()); // Mostrar en consola  
    this.pass = pass.toString();  
}
```

Creación de los constructores

```
// METODOS  
public void setLongitud(int longitud) {  
    this.longitud = longitud;  
}  
  
public String getPass() {  
    return pass;  
}
```

Creación de los método para dar la longitud y ver la pass.

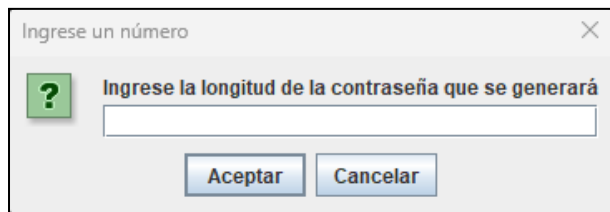
Programación Orientada a Objetos en Java

CREACIÓN DEL MÉTODO

```
public static void filtroGeneracion(String[] simbolos, StringBuilder pass, int longitud, Random numAleatorio) {  
    for (int i = 1; i <= longitud; i++) {  
        int indiceLetra = numAleatorio.nextInt(10000);  
        if (indiceLetra % 13 == 0) {  
            indiceLetra = numAleatorio.nextInt(simbolos.length);  
            pass.append(simbolos[indiceLetra]);  
        } else if (indiceLetra % 7 == 0) {  
            indiceLetra = numAleatorio.nextInt(10) + 48;  
            char character = (char) indiceLetra;  
            pass.append(character);  
        } else if (indiceLetra % 2 == 0) {  
            indiceLetra = numAleatorio.nextInt(26) + 65;  
            char character = (char) indiceLetra;  
            pass.append(character);  
        } else {  
            indiceLetra = numAleatorio.nextInt(26) + 97;  
            char character = (char) indiceLetra;  
            pass.append(character);  
        }  
    }  
}
```

Este método tendrá la función del filtro comentado antes

Ingresar el tamaño del cual queremos la contraseña

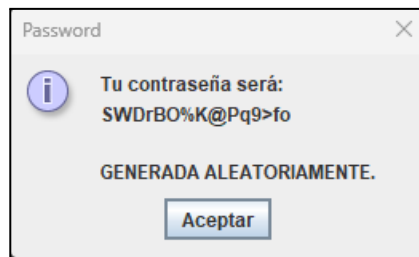


Ingrese un número

Ingrese la longitud de la contraseña que se generará

Aceptar Cancelar

Contraseña generada

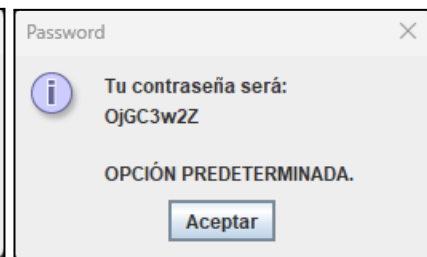


Password

Tu contraseña será:
SWDrBO%K@Pq9>fo

GENERADA ALEATORIAMENTE.

Aceptar



Password

Tu contraseña será:
OjGC3w2Z

OPCIÓN PREDETERMINADA.

Aceptar

Programación Orientada a Objetos en Java

OBJETO

Creación de los atributos

```
public class Electrodomestico {
    // ATRIBUTOS
    double precio;
    final double PRECIO_PRED = 100;
    double peso;
    final double PESO_PRED = 5;
    String color;
    final String blanco = "blanco";
    final String negro = "negro";
    final String rojo = "rojo";
    final String azul = "azul";
    final String gris = "gris";
    String[] opcionesColores = {"blanco", "negro",
                                "rojo", "azul", "gris"};

    String consumo;
    final String CONSUMO_A = "A";
    final String CONSUMO_B = "B";
    final String CONSUMO_C = "C";
    final String CONSUMO_D = "D";
    final String CONSUMO_E = "E";
    final String CONSUMO_F = "F";
    String[] opcionesConsumos = {"A", "B", "C",
                                "D", "E", "F"};
}
```

Creación de los constructores

```
// CONSTRUCTORES
public Electrodomestico() {
    this.precio = PRECIO_PRED;
    this.peso = PESO_PRED;
    this.color = blanco;
    this.consumo = CONSUMO_F;
}

public Electrodomestico(double precio, double peso) {
    this.precio = precio;
    this.peso = peso;
    this.color = blanco;
    this.consumo = CONSUMO_F;
}

public Electrodomestico(double precio, double peso, String color, String consumo) {
    this.precio = precio;
    this.peso = peso;
    boolean mal = false;
    for (int i = 0; i < opcionesColores.length; i++) {
        if (opcionesColores[i].equals(color.toLowerCase())) {
            this.color = color;
            mal = false;
            break;
        } else {
            mal = true;
        }
    }
    if (mal) {
        JOptionPane.showMessageDialog(parentComponent:null,
            "Los colores disponibles son:\n- Blanco\n- Negro\n- Rojo"+
            "\n- Azul\n- Gris", title:"Opciones colores",
            JOptionPane.INFORMATION_MESSAGE);
        this.color = blanco;
    }

    for (int i = 0; i < opcionesConsumos.length; i++) {
        if (opcionesConsumos[i].equals(consumo.toUpperCase())) {
            this.consumo = consumo;
            mal = false;
            break;
        } else {
            mal = true;
        }
    }
    if (mal) {
        JOptionPane.showMessageDialog(parentComponent:null,
            "Los tipos de consumo de energia disponibles son:"+
            "\n- A\n- B\n- C\n- D\n- E\n- F",
            title:"Opciones colores", JOptionPane.INFORMATION_MESSAGE);
        this.consumo = CONSUMO_F;
    }
}
```

Creación del método para hacerlo String

```
// METODOS
@Override
public String toString() {
    return "    Electrodomestico\n\t{" +
        "precio=" + precio + "€" + '\'' +
        ", peso=" + peso + "kg" + '\'' +
        ", color=" + color + '\'' +
        ", consumo=" + consumo + '\'' +
        '}';
}
```

Respuesta

```
Electrodomestico
    {precio='100.0?', peso=5.0kg', color=blanco', consumo=F'}
Electrodomestico
    {precio='12.5?', peso=10.0kg', color=blanco', consumo=F'}
Electrodomestico
    {precio='21.6?', peso=34.0kg', color=rojo', consumo=B'}
```

Programación Orientada a Objetos en Java

Ejercicio 04:

Por último, haremos una clase para recopilar la información de series. Los atributos que tendrá serán: el título, número de temporadas, si ha sido entregada, su género y el autor; tendremos que darle valores por defecto a los atributos: a temporadas, se le asignará el entero tres; y a entregado, false.

Imágenes Ejercicio04:

main

```
public static void main(String[] args) {
    Serie serie1 = new Serie();
    Serie serie2 = new Serie(titulo:"Black Mirror", autor:"Charlie Brooker");
    Serie serie3 = new Serie(titulo:"Hunter x Hunter", temporadas:6, genero:"anime", autor:"Yoshihiro Togashi");

    System.out.println(serie1.toString() + "\n" + serie2.toString() + "\n" + serie3.toString());
}
```

OBJETO

Creación de los atributos

```
public class Serie {
    // ATRIBUTOS
    String titulo;
    final String TITULO_PRED = "";
    int temporadas;
    final int TEMP_PRED = 3;
    boolean entregado;
    final boolean ENTREGA_PRED = false;
    String genero;
    final String GENERO_PRED = "";
    String autor;
    final String AUTOR_PRED = "";
}
```

Creación de los constructores

```
// CONSTRUCTORES
public Serie() {
    this.titulo = TITULO_PRED;
    this.temporadas = TEMP_PRED;
    this.entregado = ENTREGA_PRED;
    this.genero = GENERO_PRED;
    this.autor = AUTOR_PRED;
}

public Serie(String titulo, String autor) {
    this.titulo = titulo;
    this.temporadas = TEMP_PRED;
    this.entregado = ENTREGA_PRED;
    this.genero = GENERO_PRED;
    this.autor = autor;
}

public Serie(String titulo, int temporadas,
String genero, String autor) {
    this.titulo = titulo;
    this.temporadas = temporadas;
    this.entregado = ENTREGA_PRED;
    this.genero = genero;
    this.autor = autor;
}
```

Creación del método para hacerlo String

```
// METODOS
@Override
public String toString() {
    return " - Serie\n\t{" +
        "titulo='" + titulo + '\'' +
        ", temporadas=" + temporadas + '\'' +
        ", entregado=" + entregado + '\'' +
        ", genero=" + genero + '\'' +
        ", autor=" + autor + '\'' +
        '}';
}
```

Respuesta

```
- Serie
  {titulo='', temporadas=3, entregado=false, genero='', autor=''}
- Serie
  {titulo='Black Mirror', temporadas=3, entregado=false, genero='', autor='Charlie Brooker'}
- Serie
  {titulo='Hunter x Hunter', temporadas=6, entregado=false, genero='anime', autor='Yoshihiro Togashi'}
```

- <https://elcodigoascii.com.ar/>
- <https://github.com/MarcEsteve/java-bootcamp-techtalent-tgn2024/tree/main/C2-Java-Basics/src/UD8POO>
- El copiloto de confianza como ayuda siempre, importante:
<https://chat.openai.com/>