

Trabajo Práctico individual de presentación obligatoria.

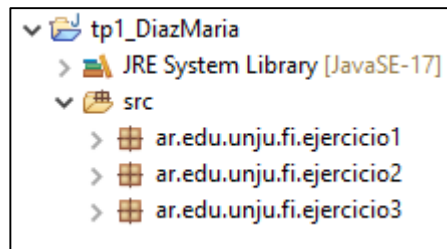
Leer las consignas atentamente.

Utilizando el IDE **Spring Tool Suite**, cree un proyecto de tipo **Java Project**. Coloque como nombre de proyecto **tp1_DiazMaria** (donde DiazMaria es el nombre del alumno).

Utilice el nombre del dominio invertido para nombrar los paquetes del proyecto:

```
ar.edu.unju.fi
```

Por cada ejercicio agregue un nuevo paquete con el nombre del ejercicio.



Uso de GitHub:

Crear un repositorio en la plataforma Github.

Desde Spring Tool Suite crear el repositorio local y subir el Proyecto al repositorio Github.

Comenzar a versionar el Proyecto Java generando commits por cada uno de los ejercicios.

En el repositorio se debe visualizar los commits con un breve mensaje referido a cada ejercicio.

El mínimo de commits generados es 10 (1 por cada ejercicio).

1. Uso de variables.

Dentro del paquete creado para el ejercicio crear la clase Main que contenga al método main().

Declare 5 variables de tipo entero para almacenar valores numéricos.

Asigne valores numéricos a cada una de las variables.

Calcule el promedio y guarde el resultado en otra variable. Muestre por consola el resultado obtenido.

Ejemplo:

8, 5, 1, 20, 9

El promedio es: 8.6

2. Uso de variables.

Dentro del paquete creado para el ejercicio crear la clase Main que contenga al método main().

Declare variables que permitan almacenar los siguientes datos en el lenguaje Java:

- El nombre de un país.
- La edad de una persona.
- La altura de un edificio.
- El precio de un producto de supermercado.
- Número de teléfono.
- El cálculo de coseno de 0.5

Asigne valores a cada variable y muéstrelas por consola.

3. Uso de variables – Estructura de control condicional.

Dentro del paquete creado para el ejercicio crear la clase Main que contenga al método main().

Declare las variables necesarias para realizar lo siguiente:

Solicitar al usuario que ingrese por consola un número entero que debe ser almacenado en una variable (Investigue como utilizar la clase **Scanner** para poder ingresar datos por consola).

Si el número ingresado es impar mostrar el doble del número y si es par mostrar el triple del número.

4. Estructura de control iterativa – estructura de control condicional.

Dentro del paquete creado para el ejercicio crear la clase Main que contenga el método main().

Declare las variables necesarias para realizar el cálculo del factorial de un número que se introduce por consola y está dentro del rango numérico [0,10]. Utilice una estructura de control iterativa **While** para resolver el ejercicio. Muestre el resultado por consola.

El factorial de un número entero (n!) se calcula de la siguiente forma:

$$\text{numero!} = \text{numero} * (\text{numero}-1) * (\text{numero}-2) \dots 1$$

$$0! = 1 \text{ (por definición)}$$

$$1! = 1$$

$$2! = 2 * 1 = 2$$

$$3! = 3 * 2 * 1 = 6$$

$$4! = 4 * 3 * 2 * 1 = 24$$

5. Estructura de control iterativa – estructura de control condicional

Dentro del paquete creado para el ejercicio crear la clase Main que contenga el método main().

Declare las variables necesarias para solicitar al usuario que ingrese por consola un número entero que esté comprendido entre [1,9], mostrar por consola la tabla de multiplicar correspondiente al número ingresado. (usar la estructura de control iterativa **for**)

Ejemplo: número ingresado 8

```
8 x 0 = 0
8 x 1 = 8
8 x 2 = 16
8 x 3 = 24
8 x 4 = 32
8 x 5 = 40
8 x 6 = 48
8 x 7 = 56
8 x 8 = 64
8 x 9 = 72
8 x 10 = 80
```

6. Uso de constructores de clase – métodos - creación de objetos.

Dentro del paquete creado para este ejercicio cree los siguientes paquetes: main y model

Dentro del paquete model crear una clase java denominada **Persona** con los siguientes atributos:

- dni
- nombre
- fecha de nacimiento (usar LocalDate)
- provincia

El modificador de acceso es **privado** para cada uno de los atributos.

Declare los **métodos accesores** para los atributos de la clase.

Dentro de la clase declare métodos que realicen las siguientes acciones:

- Calcular edad de la persona (el método retorna la edad de la persona).
- Verificar si es mayor de edad (el método devuelve true si la edad es mayor a 18 años)
- Mostrar datos, en este método se muestran todos los datos de la persona, la edad y un mensaje si es mayor de edad "La persona es mayor de edad", si no es mayor de edad se muestra un mensaje "La persona no es mayor de edad".

Agregue los siguientes constructores:

- Constructor por defecto (constructor vacío)
- Constructor que inicialice cada uno de los atributos (constructor parametrizado).
- Constructor que inicialice solo los atributos dni, nombre y fecha de nacimiento. Para el valor del atributo provincia inicializarlo con valor "Jujuy".

Dentro del método main() de la clase Main (paquete main) realice:

- La construcción de un objeto perteneciente a la clase Persona utilizando el constructor por defecto. Una vez creado el objeto mostrar el mismo por la consola (se deben mostrar todos los datos, incluyendo la edad y si es mayor de edad).

- La construcción de un objeto perteneciente a la clase Persona utilizando el constructor parametrizado. Una vez creado el objeto mostrar el mismo por la consola (se deben mostrar todos los datos, incluyendo la edad y si es mayor de edad).
- La construcción de un objeto perteneciente a la clase Persona utilizando el constructor que lleva como parámetros el dni, nombre y fecha de nacimiento. Una vez creado el objeto mostrar el mismo por la consola (se deben mostrar todos los datos, incluyendo la edad y si es mayor de edad).

Para la creación de los 3 objetos, los datos se ingresan por consola.

7. Uso de constructor de clase – métodos - constantes - creación de objetos.

Dentro del paquete creado para este ejercicio cree los siguientes paquetes: main y model.

Dentro del paquete model crear la clase **Empleado**.

Dentro de la clase declare los atributos nombre, legajo y salario. El modificador de acceso para estos atributos es privado.

Declare 2 constantes dentro de la clase, la primera constante define el salario mínimo y tiene un valor de 210.000,00\$. La segunda constante define un aumento por méritos con un valor de 20.000,00\$ (leer pdf Programación Java)

Declare un constructor parametrizado para inicializar los atributos. Dentro del constructor agregar la lógica para realizar la asignación del salario.

Si el parámetro salario es mayor o igual al salario mínimo entonces asignar al atributo salario el valor del parámetro salario.

Si el parámetro salario es menor que el salario mínimo entonces asignar al atributo salario el valor de la constante salario mínimo.

Construya un método que permita mostrar los datos del empleado, ejemplo:

Nombre del empleado: Juan Perez

Legajo: 1245

Salario \$: 950.000,75

Construya un método que permita dar un aumento al salario del empleado. Se incrementa el salario sumándole la constante aumento por méritos.

Dentro del método main() de la clase Main (paquete main) solicite al usuario que ingrese los datos para un nuevo empleado, utilice el constructor parametrizado para crear el objeto.

Luego de ingresar los datos, con el objeto creado invoque al método mostrar datos.

Invoque al método dar aumento y por último vuelva a invocar al método mostrar datos.

8. Uso de constructor de clase – métodos - creación de objetos.

Dentro del paquete creado para este ejercicio cree los siguientes paquetes: model y main.

Dentro del paquete model crear la clase denominada **CalculadoraEspecial**.

En el paquete main crear la clase denominada **Main** que contenga el método main().

El atributo de la clase es n (valor numérico entero).

El modificado de acceso del atributo es privado.

Agregue en la declaración de la clase los métodos accesoros (getters y setters).

Un constructor por defecto.

Un método llamado **calcularSumatoria()** que devuelva el resultado del siguiente cálculo:

$$\sum_{k=1}^n \left[\frac{k * (k + 1)}{2} \right]^2$$

$$\sum_{k=1}^n \left[\frac{k * (k + 1)}{2} \right]^2 = \left[\frac{1 * (1 + 1)}{2} \right]^2 + \left[\frac{2 * (2 + 1)}{2} \right]^2$$

Utilice la estructura de control iterativa “for”

Un método llamado **calcularProductoria()** que devuelva el resultado del siguiente cálculo:

$$\prod_{k=1}^n k * (k + 4)$$

$$\prod_{k=1}^n k * (k + 4) = 1 * (1 + 4) * 2 * (2 + 4)$$

Utilice la estructura de control iterativa “for”

En el método main() de la clase **Main** solicite al usuario que ingrese por consola el valor de n para construir un objeto de la clase **CalculadoraEspecial**.

Invoque los métodos del objeto y muestre el resultado por consola.

9. Uso de constructor de clase – métodos - creación de objetos.

Dentro del paquete creado para este ejercicio cree los siguientes paquetes: model y main, dentro de model crear la clase denominada **Producto**. En el paquete main crear la clase denominada **Main** que contenga el método main().

Los atributos de clase Producto son:

- Nombre
- Código
- Precio
- Descuento (valor entero comprendido entre [0,50])

El modificado de acceso de cada atributo es privado.

Agregue en la declaración de la clase los métodos accesoros (getters y setters).

Utilizar el constructor por defecto.

Un método llamado **calcularDescuento()** que devuelva el resultado de calcular el descuento del producto si es que tuviera descuento, lo que retorna representa el precio con descuento.

En el método `main()` de la clase **Main** solicite al usuario que ingrese por consola los valores para cada atributo de la clase para construir 3 objetos de la clase **Producto**. (usar estructura de control iterativa).

Muestre los datos ingresados para cada producto y el resultado de `calcularDescuento()`.

10. Uso de constructor de clase – métodos - creación de objetos.

Dentro del paquete creado para este ejercicio cree los siguientes paquetes: `model` y `main`.

Dentro de `model` crear la clase denominada **Pizza**.

En el paquete `main` crear la clase denominada **Main** que contenga el método `main()`.

Los atributos de la clase `Pizza` son los siguientes:

- `diametro`
- `precio`
- `area`
- `ingredientes especiales` (valor de tipo boolean, leer pdf Programación Java)
- `adicional ingredientes especiales 20` (constante = 500\$)
- `adicional ingredientes especiales 30` (constante = 750\$)
- `adicional ingredientes especiales 40` (constante = 1000\$)

El modificador de acceso de cada atributo es privado.

Agregue en la declaración de la clase los métodos accesoros (`getters` y `setters`).

El ejercicio consiste en solicitar al usuario que ingrese por consola los datos de 3 objetos de la clase `Pizza` (usar una estructura de control iterativa). Se ingresa el diámetro y si tiene o no ingredientes especiales (el usuario ingresa `true` o `false`).

Utilizar el constructor por defecto para la construcción de los objetos.

Los valores permitidos para el diámetro de las pizzas son: 20 (pizza pequeña), 30 (pizza mediana), 40 (pizza grande).

Crear el método `calcularPrecio()` que permita establecer el valor del precio de la pizza, el cálculo se realiza de la siguiente forma:

Si la pizza tiene un diámetro de 20 cm y no lleva ingredientes especiales, el precio es igual a 4500\$

Si la pizza tiene un diámetro de 20 cm y lleva ingredientes especiales, el precio es igual a 4500\$ más el valor de la constante adicional ingredientes especiales 20.

Si la pizza tiene un diámetro de 30 cm y no lleva ingredientes especiales, el precio es igual a 4800\$

Si la pizza tiene un diámetro de 30 cm y lleva ingredientes especiales, el precio es igual a 4800\$ más el valor de la constante adicional ingredientes especiales 30.

Si la pizza tiene un diámetro de 40 cm y no lleva ingredientes especiales, el precio es igual a 5500\$

Si la pizza tiene un diámetro de 40 cm y lleva ingredientes especiales, el precio es igual a 5500\$ más el valor de la constante adicional ingredientes especiales 40.

Crear el método `calcularArea()`, que actualiza el valor del atributo `área`.

Luego de ingresar los datos del objeto `pizza`, se deben mostrar por consola los siguientes valores:

Ejemplo:

**** Pizza 1 ****

Diámetro = 10

Ingredientes especiales = false

Precio Pizza = 4500\$

Área de la pizza = 78,539818163397