



INSTITUTO POLITÉCNICO NACIONAL
"UNIDAD PROFESIONAL INTERDISCIPLINARIA DE INGENIERÍA
Y CIENCIAS SOCIALES Y ADMINISTRATIVAS"
INGENIERIA EN INFORMATICA



CALCULADORA

MONTES CASIANO HERMES FRANCISCO
PROGRAMACIÓN ORIENTADA A OBJETOS

Yepez Granados Aylin Fernanda
2020601028

2NM42

Introducción

El objetivo de la práctica es realizar una calculadora semi-validada que tenga dos formas de funcionar, una por medio de los argumentos de java y otra con un menú impreso por consola; La consola aceptará uno o dos números según sea el caso y debe ser capaz de resolver:

- Suma
- Resta
- Multiplicación
- División
- Porcentaje
- Piso
- Techo
- Redondeo

En este reporte se hablará del desarrollo seguido para realizar la práctica y de las habilidades adquiridas.

Desarrollo

Primero se creó un proyecto en Maven para comenzar a codificar la calculadora, se cambió el JDK como siempre para poder trabajar; Fue necesario hacer 2 paquetes para separar nuestro código, el primero incluiría la lógica del negocio y el segundo la interfaz de usuario.

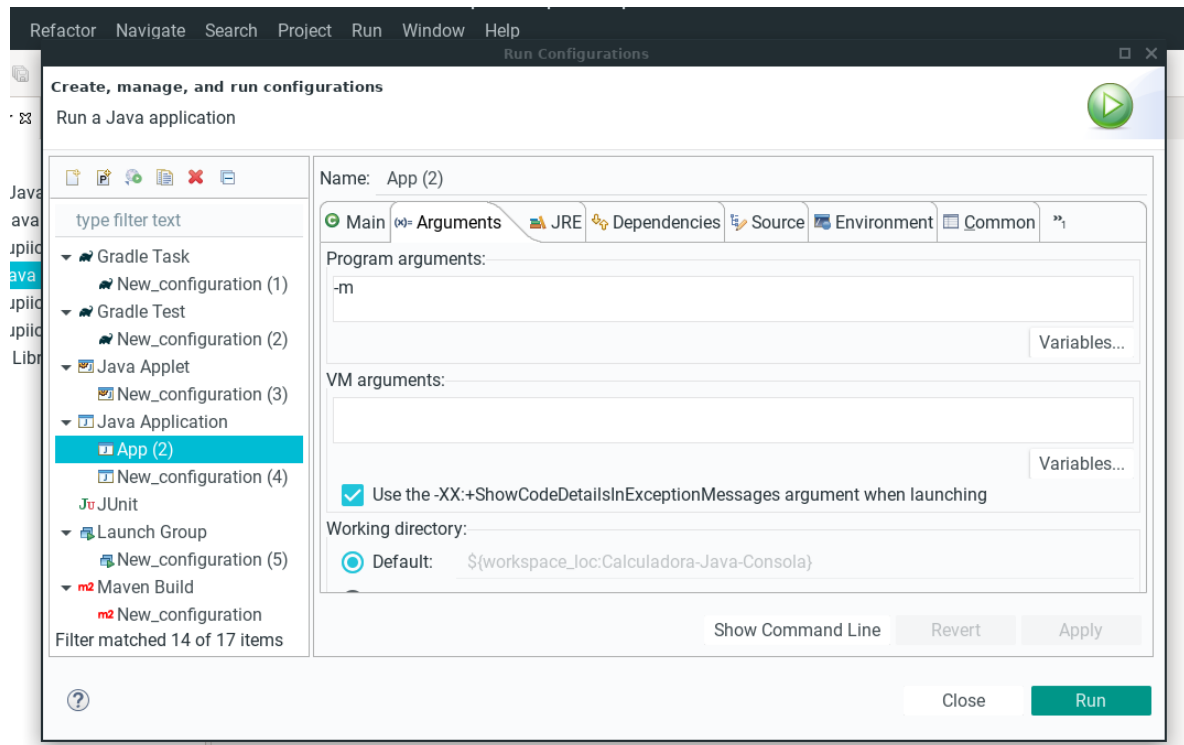
Durante el desarrollo de la práctica se crearon...

- ❖ Dentro de la lógica del negocio (bs)
 - > Calculadora: la cual incluye las funciones necesarias para la calculadora (suma, resta, multiplicación, etc.)
 - > Operación: la cual incluye los datos y métodos de una operación como lo son el operador y los números que iban a operar
 - > OperationEnum: Guarda una “colección” de nuestra operaciones, en específico guarda un identificador, nombre, comando, descripción y el número de parámetros mínimo con el que funciona dicha operación
 - > OperationMode: Como se mencionó, esta es una calculadora que tiene dos modos de funcionar, por Menú y por Argumentos/Parámetros, esta clase se encarga justo de controlar esa situación.
 - > OperationStatus: Como su nombre lo dice, contiene los estados que puede tener nuestro sistema, ya sea exitoso (success), un error (error) o un error de valor (value error)
- ❖ Dentro de ui
 - > UserInput: Controla las entradas que tiene el usuario en el sistema, es decir, en donde el usuario puede ingresar datos
 - > UserOutput: Tiene los mensajes que serán impresos en pantalla para que el usuario los pueda visualizar

Estas clases se crearon buscando tener un buen nivel de abstracción y aunque no completamente, si se trataba de adaptar el programa a modo de que en un futuro pudiese crecer.

Algo importante que quiero destacar de esta práctica es como podemos pasar parámetros/argumentos cuando corremos un programa desde eclipse, los pasos son:

1. Dar click en más opciones, la flecha que está a un lado del símbolo de play
2. Elegir "Run Configurations"
3. Seleccionar la pestaña de arguments
4. Comenzar a escribir en el primer recuadro nuestros argumentos.



Conclusión

En conclusión hacer el programa tal cual lo pide la página de la materia no me parece difícil, las cosas realmente se complican y debes tener una buena lógica, cuando empiezas a abstraer el programa y a generar una clase tras otra al mismo tiempo que las enlazar.

En esta práctica me llevo el conocimiento de los enums pues es algo que en su totalidad no sabía, la forma en que se pasan los argumentos desde un IDE a la corrida del programa y también me llevo una buena idea de lo que es, tener un problema principal y comenzar a repararlo en pedazos para poder atacarlo.