

Controller

Il diagramma delle classi rappresenta un sistema di gestione di un calcolatore che manipola variabili numeriche complesse e opera su uno stack.

Metodi:

- **isNumeric(string): bool**: verifica se una stringa è rappresentativa di un numero.
- **isCalculatorOpValid(string): bool**: verifica se l'operazione specificata è valida per il calcolatore.
- **isStackOpValid(string): bool**: verifica se l'operazione specificata è valida per lo stack.
- **isVarOpValid(string): bool**: verifica se l'operazione specificata è valida per le variabili.
- **getVar(int): NumeroComplesso**: restituisce il valore della variabile specificata.
- **setVar(int, NumeroComplesso)**: imposta il valore della variabile specificata.
- **addVar(int): boolean**: aggiunge una variabile.
- **subtractVar(int): boolean**: sottrae una variabile.
- **varToStack(int): boolean**: sposta il valore di una variabile sullo stack.
- **stackToVar(int): boolean**: sposta il valore dallo stack a una variabile.
- **addStack(): boolean**: esegue l'operazione di somma sullo stack.
- **subtractStack(): boolean**: esegue l'operazione di sottrazione sullo stack.
- **multiplyStack(): boolean**: esegue l'operazione di moltiplicazione sullo stack.
- **divideStack(): boolean**: esegue l'operazione di divisione sullo stack.
- **sqrtStack(): boolean**: esegue l'operazione di radice quadrata sullo stack.
- **errorHandler(Error)**: gestisce gli errori nel sistema.
- **inputHandler(String)**: gestisce gli input del sistema.

Controller			
-isNumeric(string)	: bool		
-isCalculatorOpValid(string)	: bool		
-isStackOpValid(string)	: bool		
-isVarOpValid(string)	: bool		
- getVar(int)	: NumeroComplesso		
- setVar(int, NumeroComplesso)			
+ addVar(int)	: boolean		
+ subtractVar(int)	: boolean		
+ varToStack(int)	: boolean		
+ stackToVar(int)	: boolean		
+ addStack()	: boolean		
+ subtractStack()	: boolean		
+ MultiplyStack()	: boolean		
+ DivideStack()	: boolean		
+ sqrtStack()	: boolean		
+ invertSign()	: boolean		
+ errorHandler(Error)	: String		
+ inputHandler(String)			

CalcStack

Il diagramma delle classi rappresenta una classe denominata **Stack** che fornisce un'implementazione di uno stack generico.

Metodi:

- **add(T)**: Aggiunge un elemento di tipo generico **T** allo stack.
- **clear()**: Cancella tutti gli elementi presenti nello stack.
- **drop(): T**: Rimuove e restituisce l'elemento in cima allo stack.
- **dup(): boolean**: Duplica l'elemento in cima allo stack e restituisce true se l'operazione ha successo.
- **swap(): boolean**: Scambia i due elementi più in cima allo stack e restituisce true se l'operazione ha successo.
- **over(): boolean**: Copia il secondo elemento in cima allo stack sopra il primo elemento e restituisce true se l'operazione ha successo.

CalcStack<T>			
- stack	: Stack<T>		
+ add(T)			
+ clear()			
+ drop():	T		
+ dup():	boolean		
+ swap():	boolean		
+ over():	boolean		
+ isEmpty():	boolean		

- **isEmpty(): boolean**: Verifica se lo stack è vuoto e restituisce true se lo è.

Error

La classe error gestisce le eccezioni e gli errori di inserimento da parte dell'utente.

Metodi:

- **NoValueInStackToOperate**: Rappresenta un errore che si verifica quando si tenta di eseguire un'operazione sullo stack, ma non ci sono valori sufficienti nello stack.
- **NoValueInVarToOperate**: Rappresenta un errore che si verifica quando si tenta di eseguire un'operazione su variabili, ma non ci sono valori sufficienti nelle variabili.
- **CantDivideByZero**: Rappresenta un errore che si verifica quando si tenta di eseguire una divisione per zero.

<<Enumerator>> Error
InputError NoValueInStackToOperate NoValueInVarToOperate CantDivideByZero

NumeroComplesso

La classe NumeroComplesso gestisce le funzioni e le interazioni relative numeri complessi.

Metodi:

- **realPart**: Rappresenta la parte reale del numero complesso.
- **complexPart**: Rappresenta la parte immaginaria del numero complesso.
- **NumeroComplesso(double)**: Crea un numero complesso con la parte reale specificata e la parte immaginaria impostata a zero.
- **NumeroComplesso(double, double)**: Crea un numero complesso con le parti reale e immaginaria specificate.
- **NumeroComplesso(NumeroComplesso)**: Crea un numero complesso copiando un altro numero complesso.
- **getRealPart(): double**: Restituisce la parte reale del numero complesso.
- **getComplexPart(): double**: Restituisce la parte immaginaria del numero complesso.
- **getModule(): double**: Calcola e restituisce il modulo (magnitudine) del numero complesso.
- **getAngle(): double**: Calcola e restituisce l'angolo (argomento) del numero complesso.
- **toString(): String**: Restituisce una rappresentazione testuale del numero complesso.
- **stringToNumeroComplesso(string): NumeroComplesso**: Un metodo di classe (o statico) che converte una stringa in un oggetto NumeroComplesso.

NumeroComplesso	
- realPart: double	
- complexPart: double	
+ stringToNumeroComplesso(string):	NumeroComplesso
+ NumeroComplesso(double)	
+ NumeroComplesso(double, double)	
+ NumeroComplesso(NumeroComplesso)	
+ getRealPart():	double
+ getComplexPart():	double
+ getModule():	double
+ getAngle():	double
+ toString():	String

Calcolatore

Questa classe rappresenta un numero complesso e sembra fornire operazioni aritmetiche su di esso.

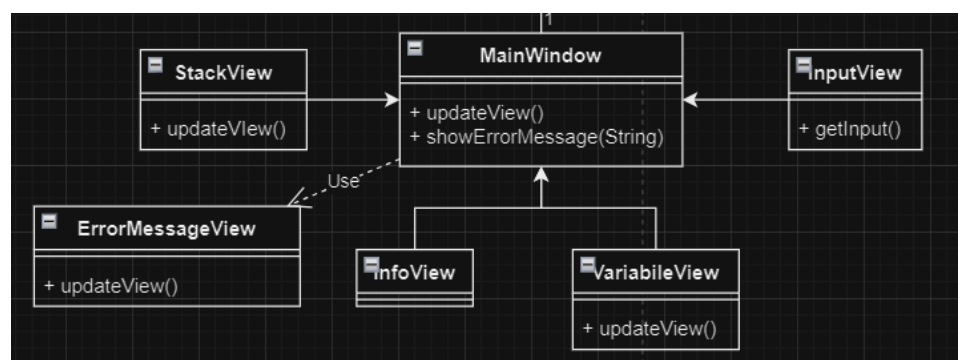
Calcolatore		
+ <u>add</u> (NumeroComplesso, NumeroComplesso):	NumeroComplesso	
+ <u>subtract</u> (NumeroComplesso, NumeroComplesso):	NumeroComplesso	
+ <u>multiply</u> (NumeroComplesso, NumeroComplesso):	NumeroComplesso	
+ <u>divide</u> (NumeroComplesso, NumeroComplesso):	NumeroComplesso	
+ <u>invertSign</u> (NumeroComplesso):	NumeroComplesso	
+ <u>sqrt</u> (NumeroComplesso):	NumeroComplesso	

Metodi:

- **add(NumeroComplesso, NumeroComplesso):** Questo metodo prende due oggetti **NumeroComplesso** come parametri e restituisce un nuovo oggetto **NumeroComplesso** che rappresenta la somma dei due numeri complessi.
- **subtract(NumeroComplesso, NumeroComplesso):** Questo metodo accetta due oggetti **NumeroComplesso** come parametri e restituisce un nuovo oggetto **NumeroComplesso** che rappresenta la differenza tra i due numeri complessi.
- **multiply(NumeroComplesso, NumeroComplesso):** Questo metodo prende due oggetti **NumeroComplesso** come parametri e restituisce un nuovo oggetto **NumeroComplesso** che rappresenta il prodotto dei due numeri complessi.
- **divide(NumeroComplesso, NumeroComplesso):** Questo metodo accetta due oggetti **NumeroComplesso** come parametri e restituisce un nuovo oggetto **NumeroComplesso** che rappresenta il rapporto divisione tra i due numeri complessi.
- **invertSign(NumeroComplesso):** Questo metodo prende un oggetto **NumeroComplesso** come parametro e restituisce un nuovo oggetto **NumeroComplesso** che rappresenta il numero complesso con il segno invertito.
- **sqrt(NumeroComplesso):** Questo metodo prende un oggetto **NumeroComplesso** come parametro e restituisce un nuovo oggetto **NumeroComplesso** che rappresenta la radice quadrata del numero complesso.

MainView

La classe MainView gestisce le informazioni visibili tramite l'interfaccia.



Metodi:

- **updateView():** Questo metodo è responsabile di aggiornare la MainView. Potrebbe essere utilizzato per reperire modifiche nei dati o nello stato dell'applicazione sulla parte visiva.
- **showErrorMessage(String):** Questo metodo accetta una stringa come parametro come messaggio di errore, e gestisce la visualizzazione di questo messaggio nella vista principale. Può essere utilizzato per informare l'utente di errori o problemi nell'applicazione.