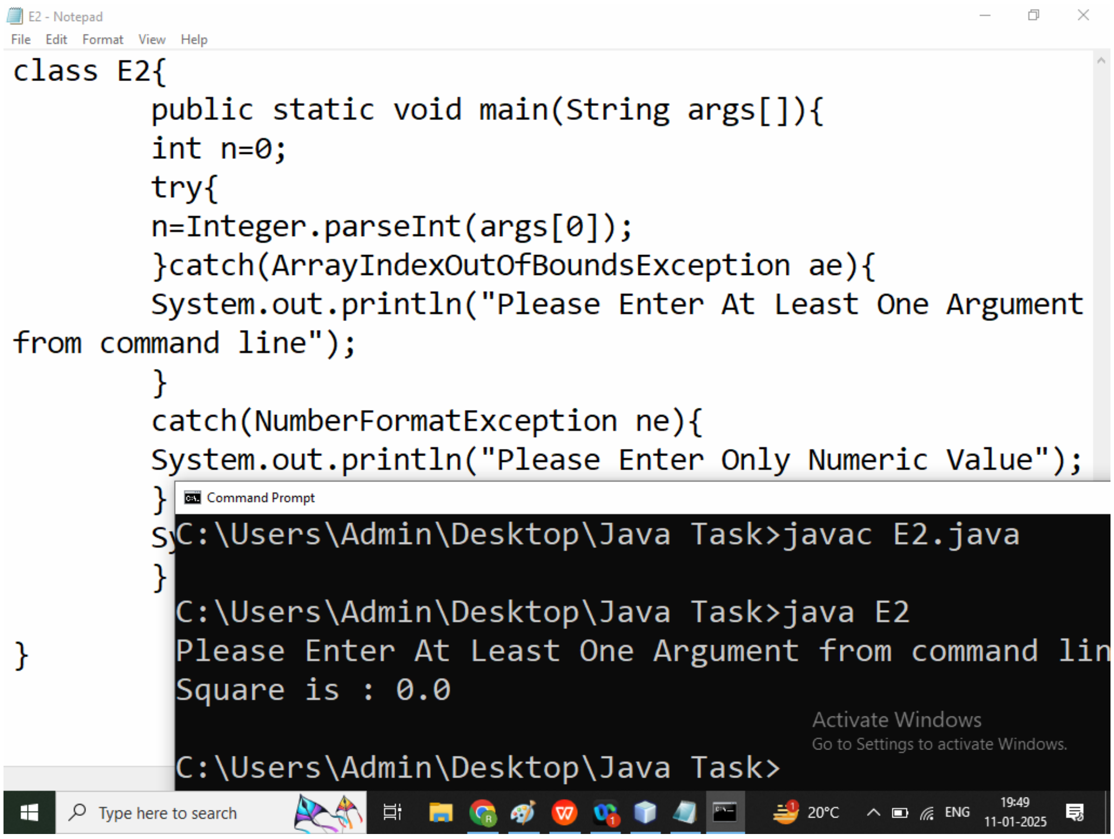
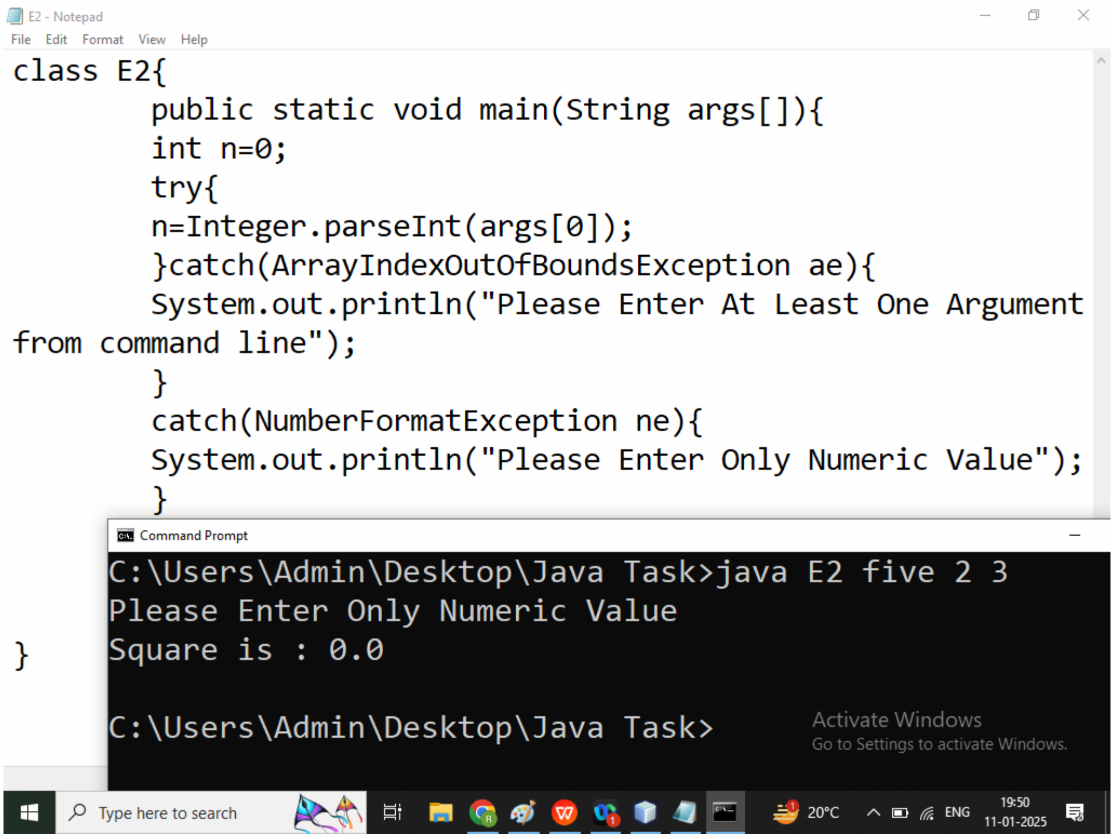
Q1. Explain Command Line Argument in java Programming?
Ans: Command Line Argument is used to allow to pass information to a program at the time of execution of the Program

Syntax:
java ClassName Hello Hi How Are You
java ClassName 1 2 3 4 5
Java ClassName



```
class E2{
        public static void main(String args[]){
        int n=0;
        try{
        n=Integer.parseInt(args[0]);
        }catch(ArrayIndexOutOfBoundsException ae){
        System.out.println("Please Enter At Least One Argument
from command line");
        }
        catch(NumberFormatException ne){
        System.out.println("Please Enter Only Numeric Value");
        }
        S
        }
}
```

```
C:\Users\Admin\Desktop\Java Task>javac E2.java

C:\Users\Admin\Desktop\Java Task>java E2
Please Enter At Least One Argument from command lin
Square is : 0.0

C:\Users\Admin\Desktop\Java Task>
```

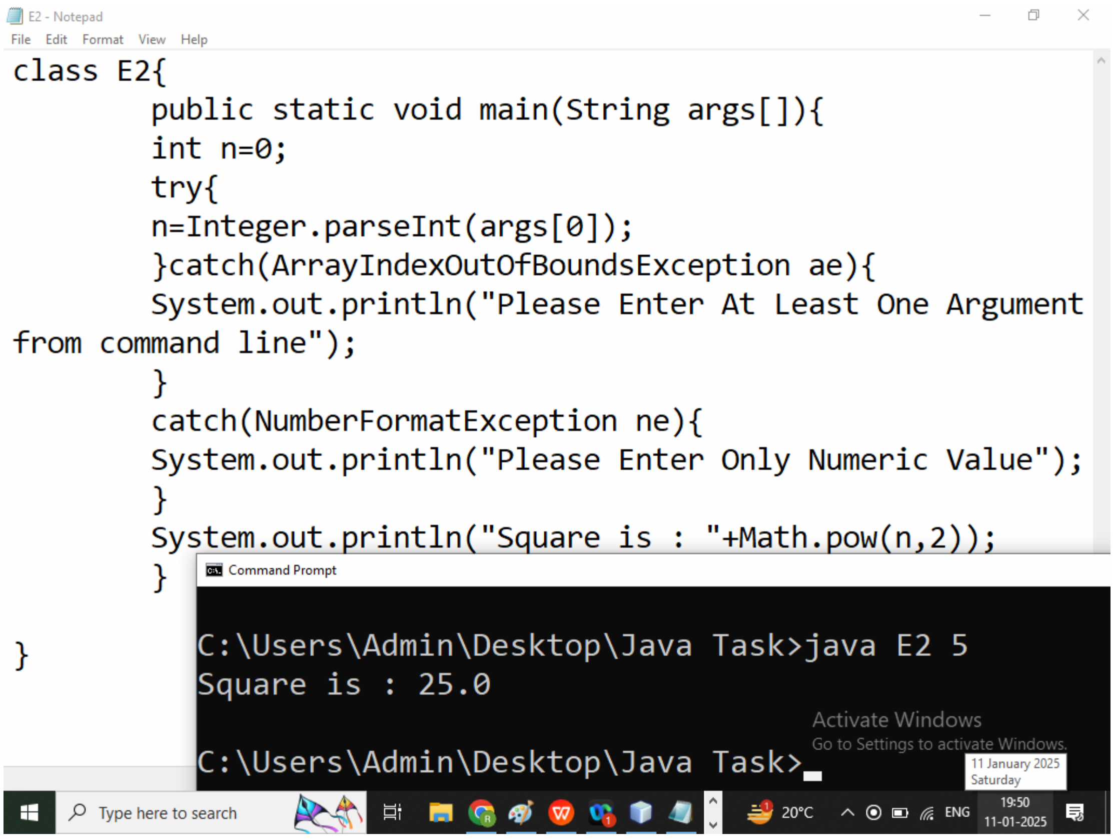


E2 - Notepad

```
class E2{
		public static void main(String args[]){
		int n=0;
		try{
		n=Integer.parseInt(args[0]);
		}catch(ArrayIndexOutOfBoundsException ae){
		System.out.println("Please Enter At Least One Argument
from command line");
		}
		catch(NumberFormatException ne){
		System.out.println("Please Enter Only Numeric Value");
		}

}
```

Command Prompt

```
C:\Users\Admin\Desktop\Java Task>java E2 five 2 3
Please Enter Only Numeric Value
Square is : 0.0

C:\Users\Admin\Desktop\Java Task>
```



E2 - Notepad

```
class E2{
		public static void main(String args[]){
		int n=0;
		try{
		n=Integer.parseInt(args[0]);
		}catch(ArrayIndexOutOfBoundsException ae){
		System.out.println("Please Enter At Least One Argument
from command line");
		}
		catch(NumberFormatException ne){
		System.out.println("Please Enter Only Numeric Value");
		}
		System.out.println("Square is :  "+Math.pow(n,2));
		}

}
```

Command Prompt

```
C:\Users\Admin\Desktop\Java Task>java E2 5
Square is : 25.0

C:\Users\Admin\Desktop\Java Task>
```

```
class E2{
		public static void main(String args[]){
		int n=0;
		try{
        n=Integer.parseInt(args[0]);
        }catch(ArrayIndexOutOfBoundsException ae){
        System.out.println("Please Enter At Least One Argument from command line");
        }
        catch(NumberFormatException ne){
        System.out.println("Please Enter Only Numeric Value");
        }
        System.out.println("Square is : "+Math.pow(n,2));
        }

}
```

```
class E2{
        public static void main(String args[]){
        int n=0;
        try{
        n=Integer.parseInt(args[0]);
        }catch(ArrayIndexOutOfBoundsException ae){
        System.out.println("Please Enter At Least One Argument from command line");
        ae.printStackTrace();

        }
        catch(NumberFormatException ne){
        System.out.println("Please Enter Only Numeric Value");
        ne.printStackTrace();

        }
        System.out.println("Square is : "+Math.pow(n,2));
        }

}
```

Q1. Explain User Defined Exception class or Customize Exception class in java Programming?
Ans: In java if you are not satisfied from the pr-defined exception then you also create your own Exception class/ User Defined Exception/ Customize Exception

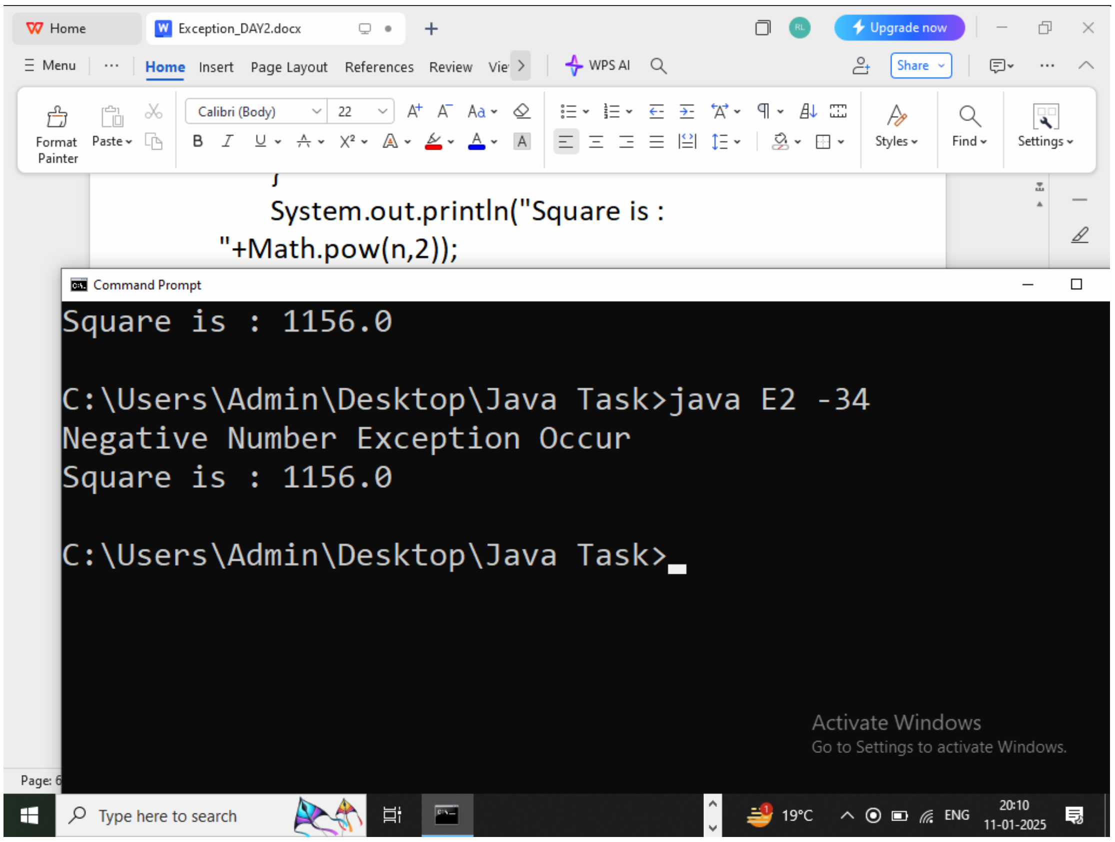
You can create a User defined Exception or Customize Exception by extending Throwable or Exception Class

Steps to create a User Defined Class in java
1. Create a Custom Class/ User Defined Class for the exception by extending Throwable or Exception(define Member data String)
2. Define a Parameterized Constructor and method to return message
3. Throw the Exception from where you handle it.
4. Handle The Exception using try catch

```
class NegativeException extends Throwable{
private String msg;
public NegativeException(String msg){
this.msg=msg;
}

public String getMsg(){
return msg;
}

}
```

```
class E2{
    public static void main(String args[]){
    int n=0;
    try{
    n=Integer.parseInt(args[0]);
    if(n<0){
    NegativeException x=new
NegativeException("Negative Number
Exception Occur");
    throw  x;
    }
    }catch(ArrayIndexOutOfBoundsException
ae){
    System.out.println("Please Enter At Least
One Argument from command line");
    ae.printStackTrace();

    }
    catch(NumberFormatException ne){
    System.out.println("Please Enter Only
Numeric Value");
    ne.printStackTrace();

    }
    catch(NegativeException y){
    System.out.println(y.getMsg());
    }
    System.out.println("Square is :
"+Math.pow(n,2));
    }
```

}



```
System.out.println("Square is :
"+Math.pow(n,2));
```

```
Square is : 1156.0

C:\Users\Admin\Desktop\Java Task>java E2 -34
Negative Number Exception Occur
Square is : 1156.0

C:\Users\Admin\Desktop\Java Task>
```