**OBJECT ORIENTED PROGRAMMING 2**

**BBIT EC**

**PREET TAILOR**

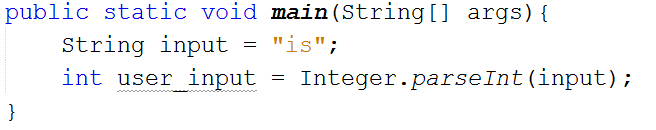
**094790**

# **LAB 1**

# .......................................................................................................................

# Section 1

Create java class and write the codes shown below. Run your program

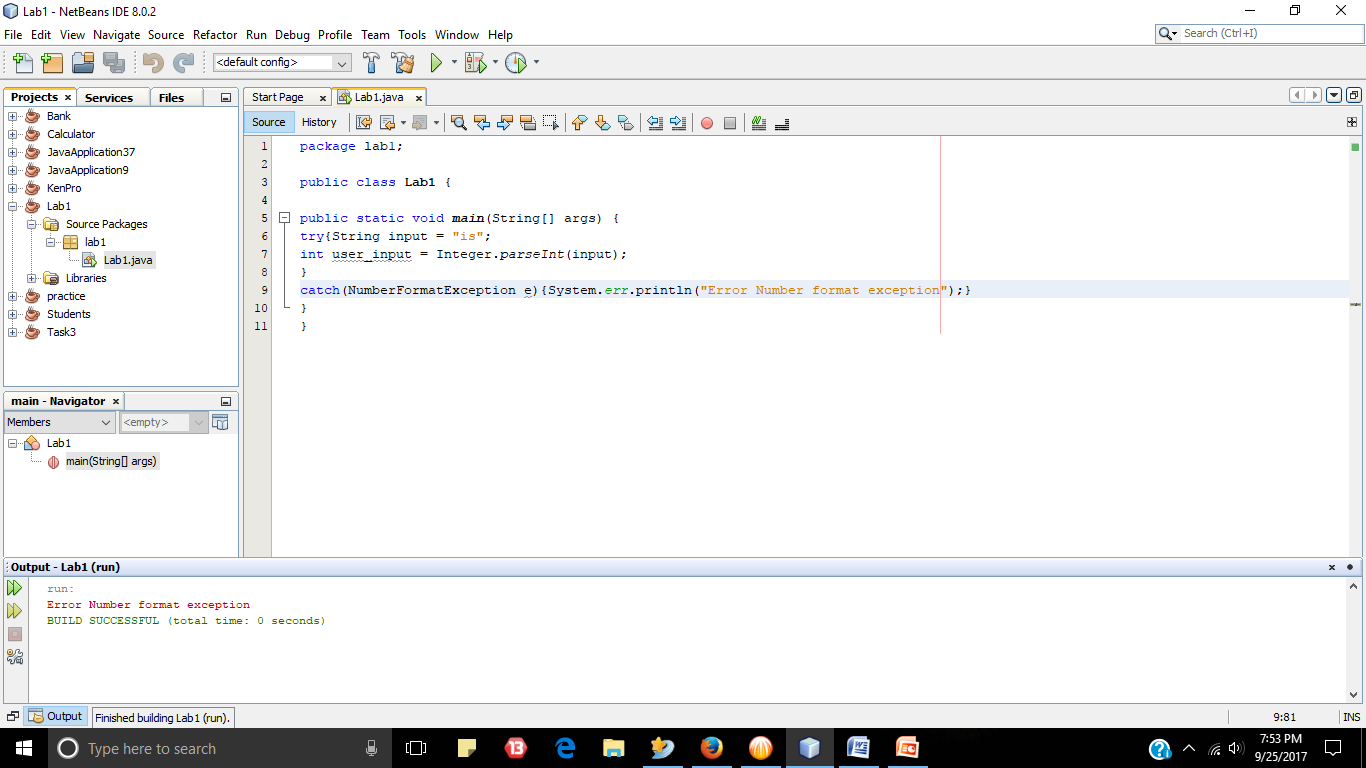


You will probably see an error as a result of an exception

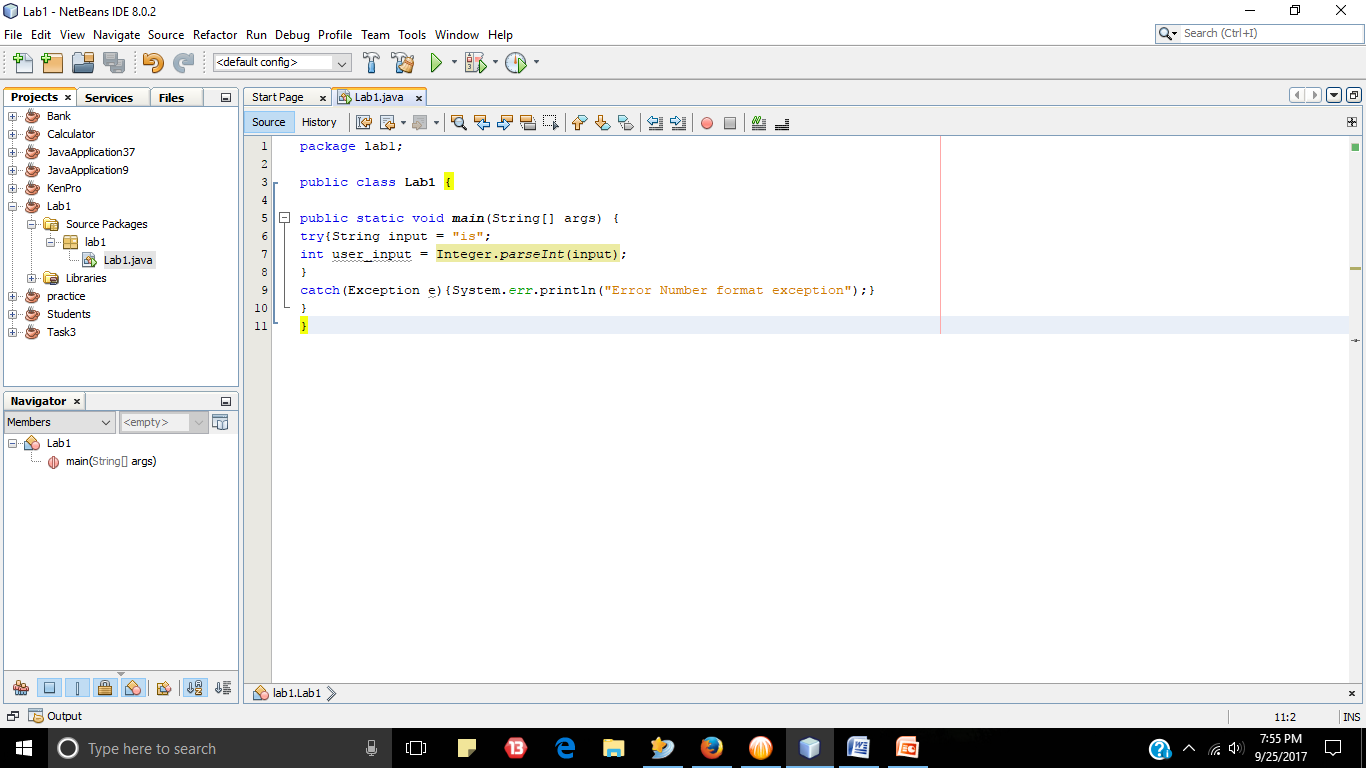
**What is the name of the exception?** NumberFormatException

Handle the exception as follows;

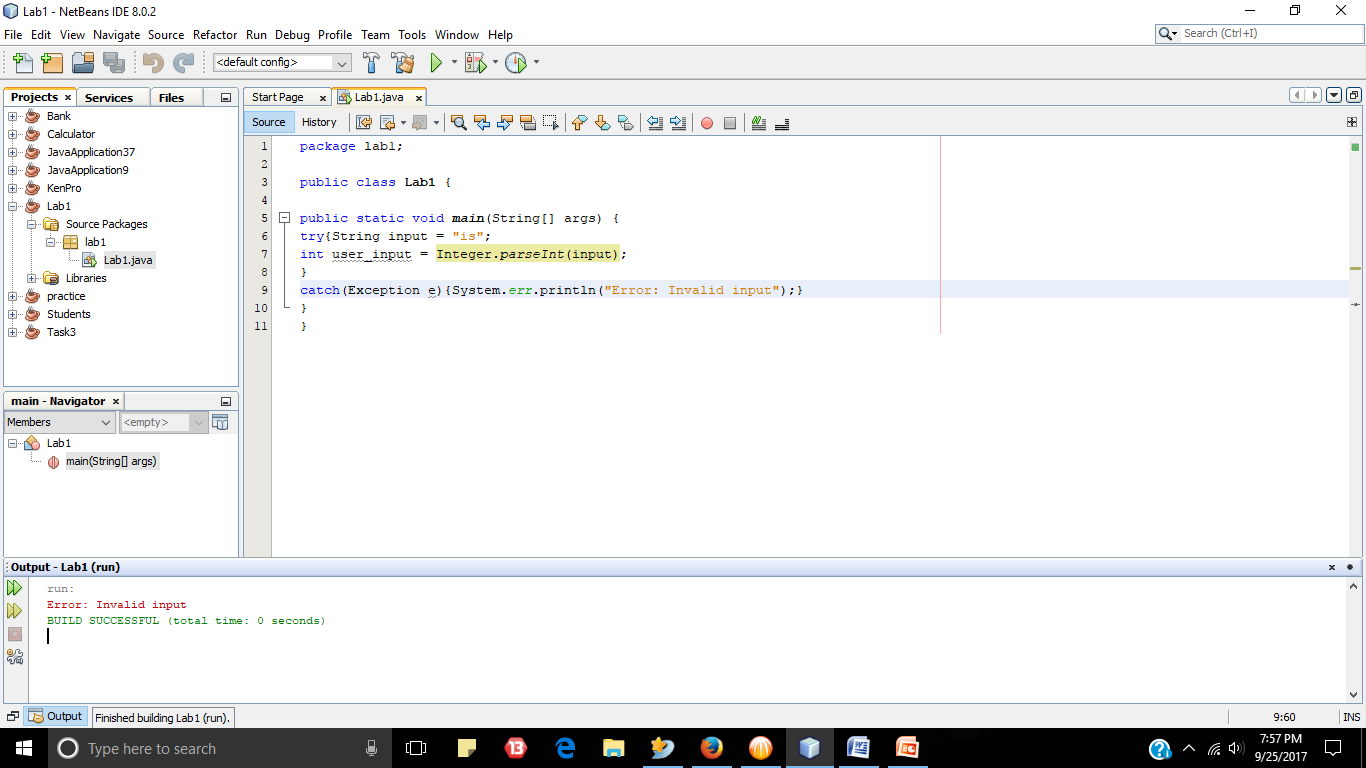
1. In scenario 1, catch the specific exception as indicated by the error



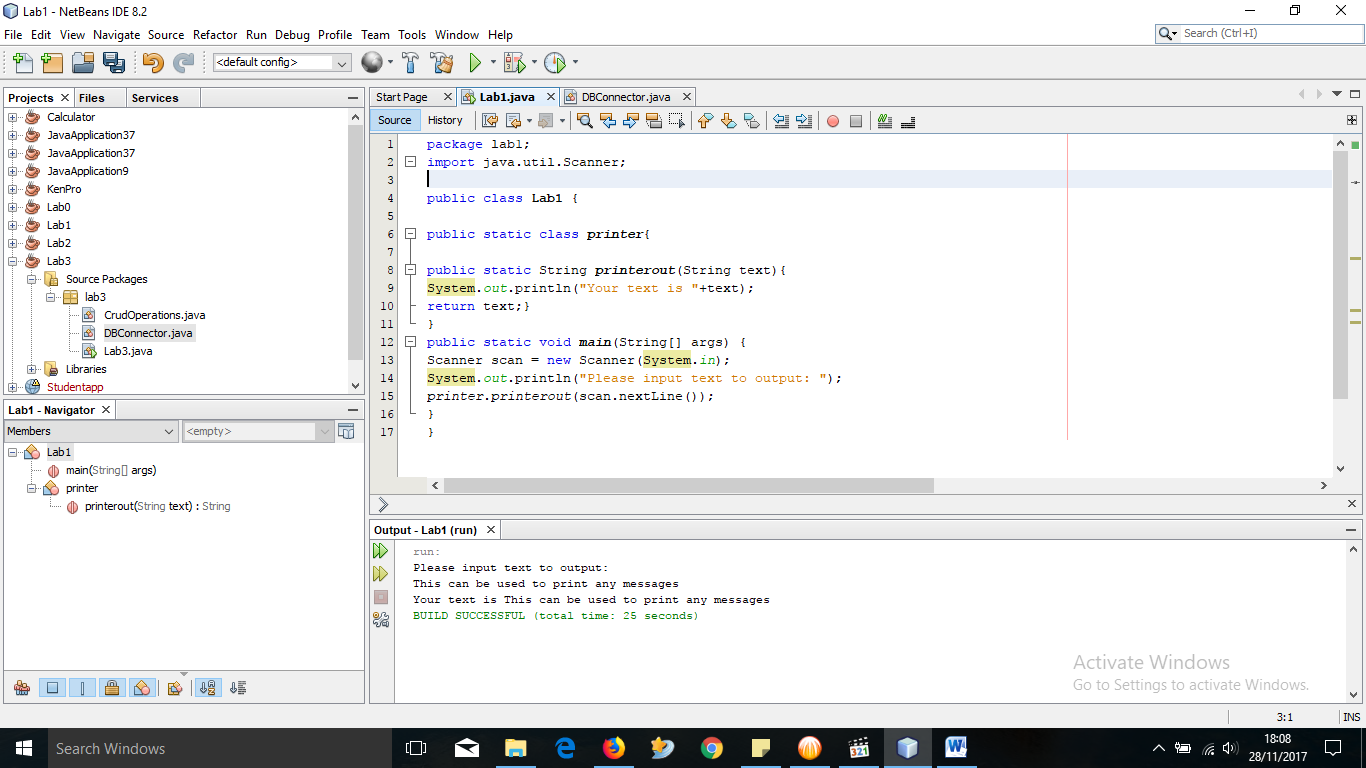
1. In scenario 2, catch a general exception



In each of the scenario (1 & 2) above, show how you generate a friendly message to be shown to users of your program.



Create an inner class with one static method called ***printer***. Printer has one parameter, *text*, printer method prints whatever text received in *text.* You can use this class to print any messages.

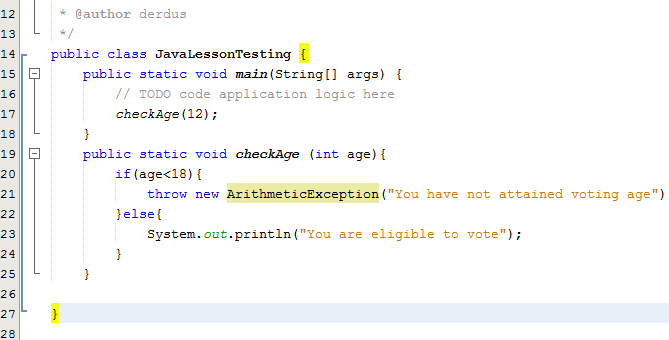


# Section 2

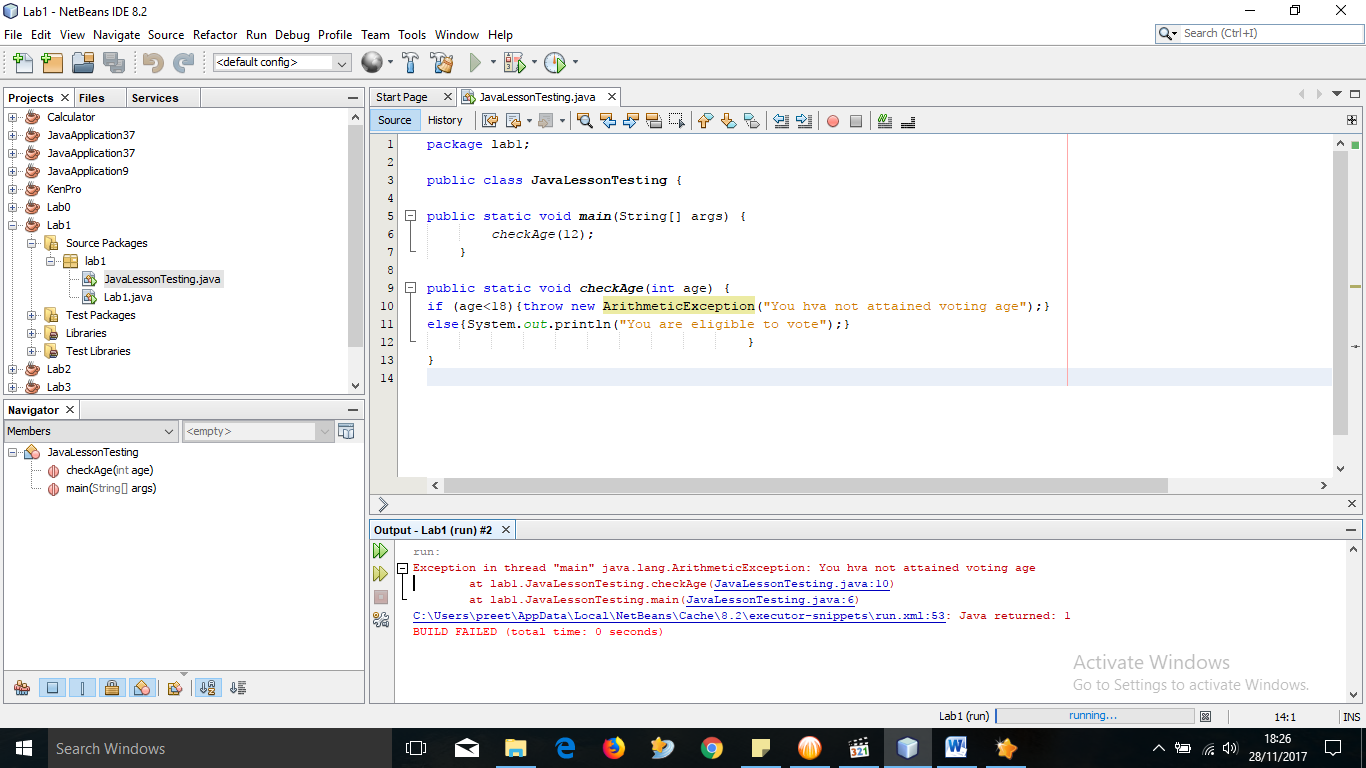
**In this section, we investigate how we can explicitely throw an exception. Proceed as follows**

**Suppose that a kenyan below 18 years cannot and that the rest can. So we want to throw an exception of type ArithmeticException , if the age of a person is below 18 and then show this error with a message**

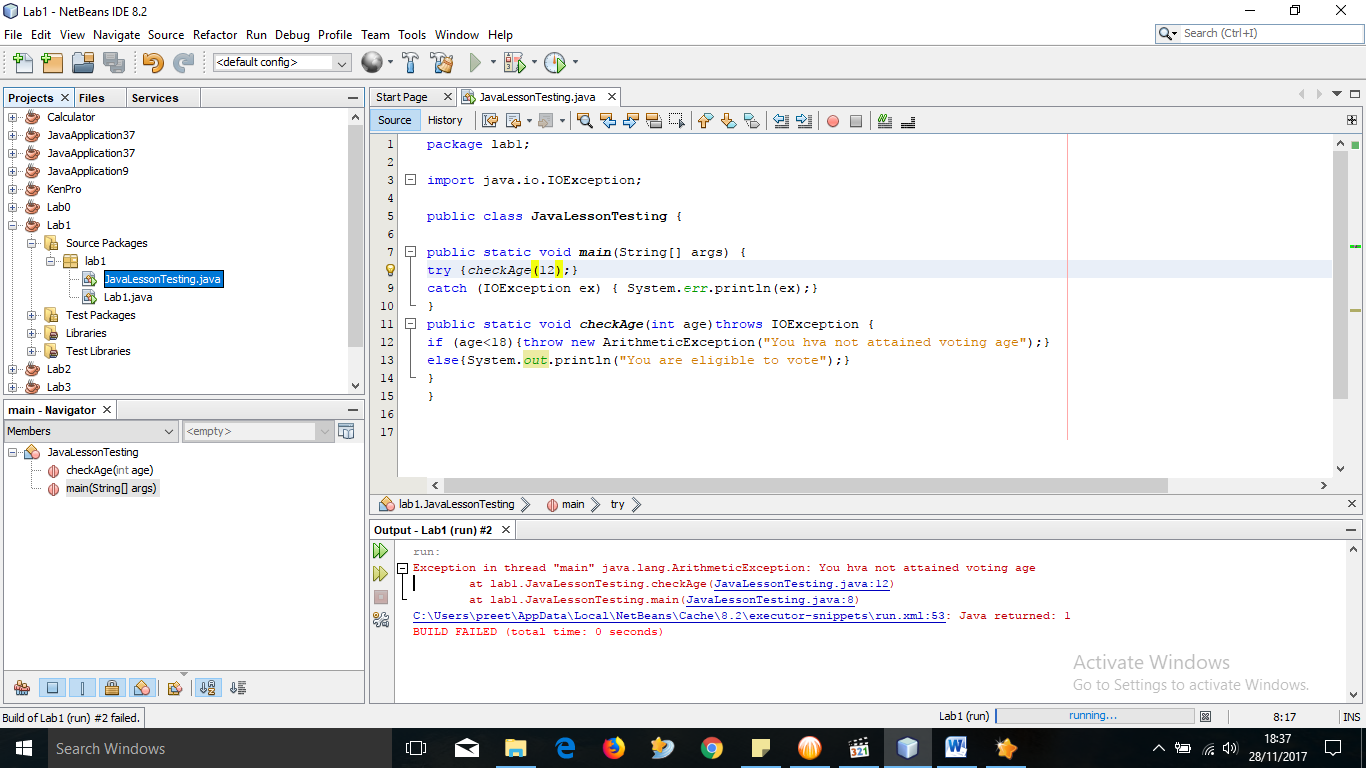
**Create a class and methods as shown below**



**Run the program and record your observation**



**Show how you the method checkAge can throw an IOException using the *throws* key, then catch the exception.**

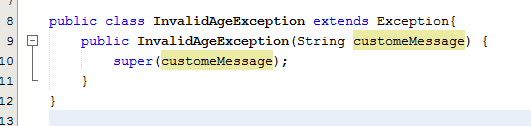


# Section 3

**In section 2, we just assumed that being below 18 years could be put in ArithmeticException exception. This may not correct, so we want to create our own exception name and then throw it.**

**We will call this exception InvalidAgeException**

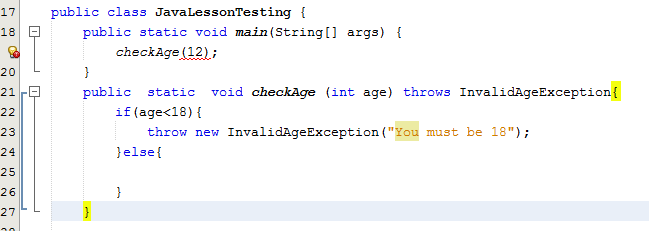
**Create a java class InvalidAgeException that extends Exception class. In the constructor of this class, initilaize exception error message in the parent class. This is shown below**



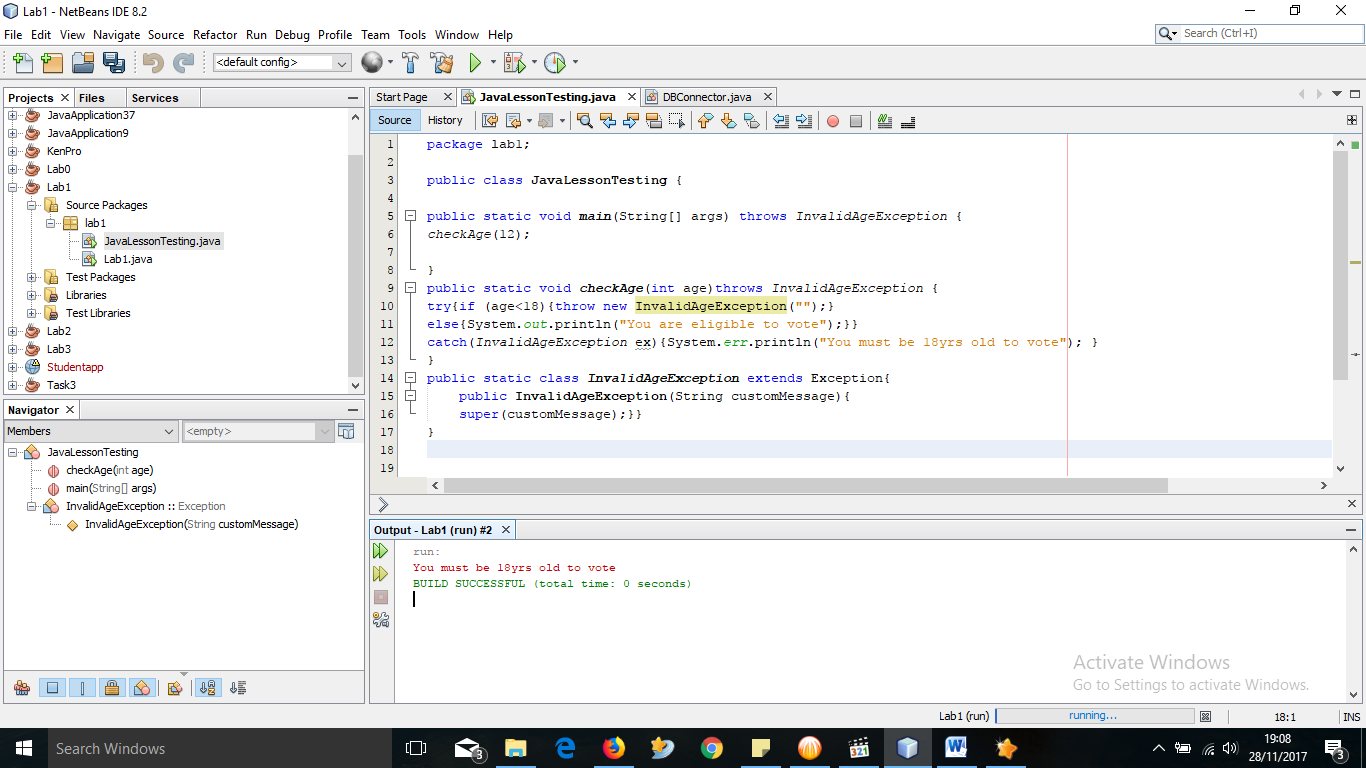
**Explain what is happening in line 10.**

“super()” calls the default constructor of the super class. If you don't define a constructor your class always has a invisible default constructor, which doesn't require any parameters. You are calling the constructor of the Object class in this case.

**In another class, show how you throw the InvalidAgeException if age of a person is below 18. See sample code below**



**Proceed and catch the exception. Show how you show a user a friendly message by accessing the exception message and printing it.**



**Why is it advisable that you catch the general exception and why must you do it at the end.**

So that you can catch errors that you missed and improve on the code. It is advisable to do it at the end so that the program is allowed to run and is after already known exceptions.

***-This lab is complete-***