EXCEPTION HANDLING

1. **Difference between exception and error.**

    - Exception: Able to handle the interrupt.
    - Error: Not able to handle the interrupt.

2. **Types of exception:**

    - Checked | compile time (Ex: Java).
    - Unchecked | run time (Ex: Kotlin - created to replace java).
    - Customized (Our own exception).

3. **Types of error:**

    - Compile time error | syntactical error.
    - Run time error: There is no compile time issue here. But execution normally stops. We should handle those situation with the help of exception handling.
    - Logical error | bugs: There is no compile time or run time issues here. But output we are expecting is not coming.

4. **Real time examples for exception handling:**

    - Medical Domain: There is surgery scheduled for a patient. During the surgery, one of the medical software programs stopped executing.
    - Banking Domain: Suppose I want to withdraw some money that is more than the amount I have in my account.
    - Online Shopping Domain: I ordered 5 self-help books from BookSpringIndia. Upon delivery, I received books that were mismatched.

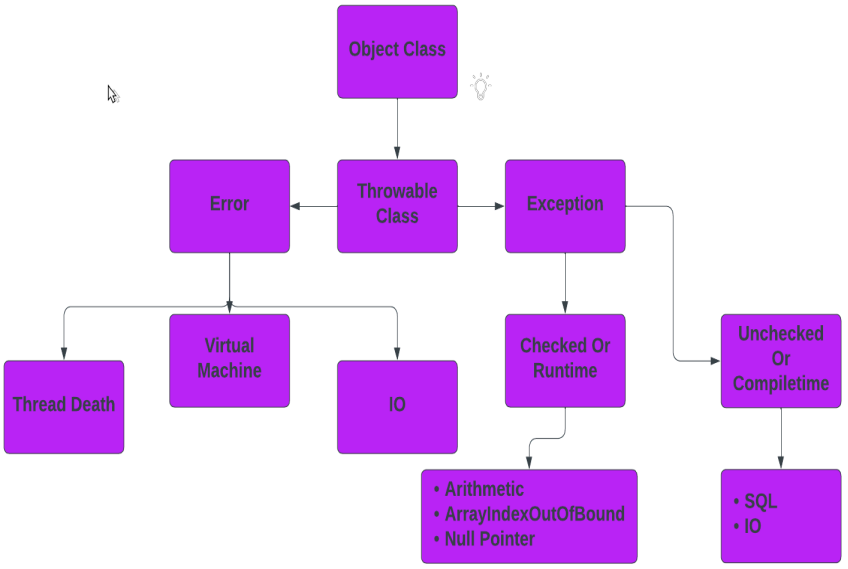
5. **Keywords to handle the exception:**

- Try.
- Catch.
- Throw.
- Finally.
- Throws (Few programming languages).

6. **Try:**

- It will try to execute the statement.
- If everything works fine it will execute the remaining statements.
- If something was wrong it will try to execute the catch block.

7. **Exception hierarchy:**



8. **Catch:**

- Catch block will be executed if only there is a exception.
- Catch block will catch the exception thrown by try.

9. Throw:

- Throw keyword will throw the exception to catch block.

10. Finally:

- If we got the exception or not, it will execute the finally block.
- It makes sense whenever we want to close the resource.

11. Quick activity during the session:

a) Can I have try block with only finally block?

- Yes we can have.
- With the help of try block we can open any resource and with the help of finally block we can close the resource which was opened.

```java
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class TryWithFinally {
    Run | Debug
    public static void main(String[] args) throws NumberFormatException, IOException {
        int productId;
        BufferedReader br = null;
        try{
            br = new BufferedReader(new InputStreamReader(System.in));
            productId = Integer.parseInt(br.readLine());
            System.out.println(productId);
        }
        finally{
            br.close();
        }
    }
}
```

```
PROBLEMS 69  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

C:\Users\VARSHINISRI\Downloads\COURSE>javac TryWithFinally.java

C:\Users\VARSHINISRI\Downloads\COURSE>java TryWithFinally
101
101
```

b) Can I have finally block inside nested try block? Yes.

MismatchedBookOrder.java > MismatchedBookOrder > handleOrder(int)

```java
public class MismatchedBookOrder{
    public static void handleOrder(int orderQuantity){
      try{
        System.out.println("Received " + orderQuantity + " books from BookSpringIndia.");
        try{
          throw new Exception(message:"Books don't match the order!");
        }
        finally{
          System.out.println(x:"Contacting BookSpringIndia for a replacement...");
        }
      }
      catch (Exception e){
        System.out.println("Error: " + e.getMessage());
        System.out.println(x:"Contacting BookSpringIndia for a replacement...");
      }
    }
    Run | Debug
    public static void main(String[] args){
      handleOrder(orderQuantity:5);
}
}
```

PROBLEMS 71 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
C:\Users\VARSHINISRI\Downloads\COURSE>java MismatchedBookOrder
Received 5 books from BookSpringIndia.
Contacting BookSpringIndia for a replacement...
Error: Books don't match the order!
Contacting BookSpringIndia for a replacement...
```

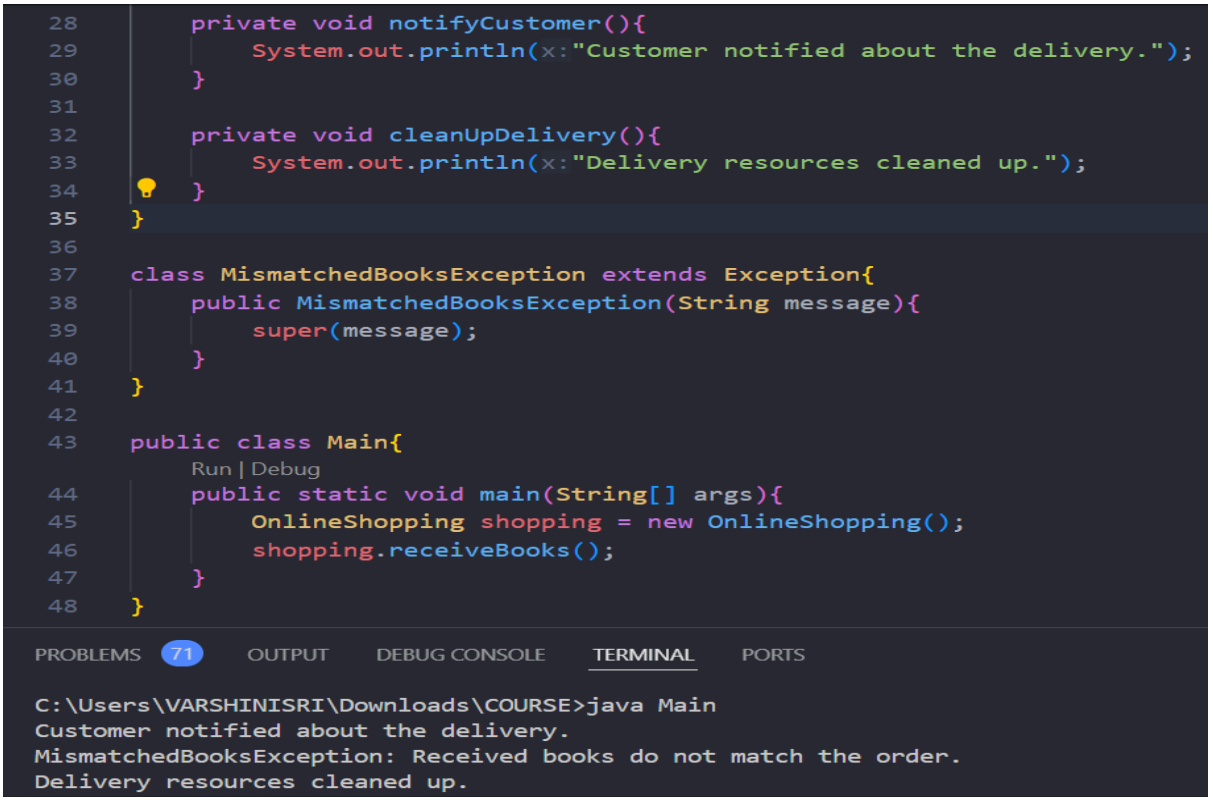
c) Can I have more than 1 finally block?

- We cannot have more than one finally block associated with a single try block.
- But we can have nested try-catch-finally blocks.

```
Main.java > OnlineShopping
class OnlineShopping{
    public void receiveBooks(){
        try{
            verifyBooks();
        }
        catch(MismatchedBooksException e){
            System.out.println("MismatchedBooksException: " + e.getMessage());
        }
        finally{
            cleanUpDelivery();
        }
    }

    private void verifyBooks() throws MismatchedBooksException{
        try{
            int numberOfBooksReceived = 3;
            int numberOfExpectedBooks = 5;

            if(numberOfBooksReceived != numberOfExpectedBooks){
                throw new MismatchedBooksException(message:"Received books do not match the order.");
            }
        }
        finally{
            notifyCustomer();
        }
    }
```



```
    private void notifyCustomer(){
        System.out.println(x:"Customer notified about the delivery.");
    }

    private void cleanUpDelivery(){
        System.out.println(x:"Delivery resources cleaned up.");
    }
}

class MismatchedBooksException extends Exception{
    public MismatchedBooksException(String message){
        super(message);
    }
}

public class Main{
    Run | Debug
    public static void main(String[] args){
        OnlineShopping shopping = new OnlineShopping();
        shopping.receiveBooks();
    }
}
```

```
PROBLEMS 71   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

C:\Users\VARSHINISRI\Downloads\COURSE>java Main
Customer notified about the delivery.
MismatchedBooksException: Received books do not match the order.
Delivery resources cleaned up.
```

12. Practised things:

a) Try with multiple catch:

```java
public class TryWithMultipleCatch {
    public static void main(String[] args) {
        int num1 = 0;
        int num2 = 5;
        int rollno[] = new int[5];
        String str = null;
        try{
            System.out.println(str.length());
            num2 = 6/num1;
            System.out.println(rollno[5]);}
        catch(ArithmeticException e){
            System.out.println("Cannot divide by zero.");}
        catch(ArrayIndexOutOfBoundsException e){
            System.out.println("Stay in your limit.");}
        catch(Exception e){
            System.out.println("Something went wrong." + e);}}}
```

```
Microsoft Windows [Version 10.0.22621.3593]
(c) Microsoft Corporation. All rights reserved.

C:\Users\VARSHINISRI\Downloads\COURSE>javac TryWithMultipleCatch.java

C:\Users\VARSHINISRI\Downloads\COURSE>java TryWithMultipleCatch
Something went wrong.java.lang.NullPointerException: Cannot invoke "String.length()" because "<local4>" is null
```

b) Throw keyword:

```java
public class ThrowKeyword {
    public static void main(String[] args) {
        int num1 = 6;
        int num2 = 0;
        try{
            num2 = 3/num1;
            if(num2 == 0){
                throw new ArithmeticException("I don't want to print zero");
            }
        }
        catch(ArithmeticException e){
            num2 = 3/1;
            System.out.println("That's the deafult output "+e);
        }
        catch(Exception e){
            System.out.println("Something went wrong.."+e);
        }
        System.out.println(num2);
    }
}
```

```
C:\Users\VARSHINISRI\Downloads\COURSE>java ThrowKeyword
That's the deafult output java.lang.ArithmeticException: I don't want to print zero
3
```

c) Custom exception:

CustomExceptions.java > CustomExceptions > main(String[])

```
1   class CustomException extends Exception{
2       public CustomException(String string){
3           super(string);}}
4   public class CustomExceptions {
        Run | Debug
5       public static void main(String[] args) {
6           int num1 = 6;
7           int num2 = 0;
8           try{
9               num2 = 3/num1;
10              if(num2 == 0){
11                  throw new CustomException(string:"I don't want to print zero");
12              }
13          }
14          catch(ArithmeticException e){
15              num2 = 3/1;
16              System.out.println("That's the deafult output "+e);
17          }
18          catch(CustomException e){
19              System.out.println("Something went wrong.."+e);
20          }
21          System.out.println(num2);}}
```

PROBLEMS 71 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
C:\Users\VARSHINISRI\Downloads\COURSE>java CustomExceptions
Something went wrong..CustomException: I don't want to print zero
0
```

d) Ducking exception using throws:

DuckingExceptionUsingThrows.java > DuckingExceptionUsingThrows > main(String[])

```
1   class A{
2       public void show() throws ClassNotFoundException{
3           Class.forName(className:"Library");
4       }
5   }
6
7   public class DuckingExceptionUsingThrows {
        Run | Debug
8       public static void main(String[] args) {
9           A obj = new A();
10          try{
11              obj.show();
12          }
13          catch(ClassNotFoundException e){
14              System.out.println(e);
15          }
16      }
17  }
```

PROBLEMS 69 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Microsoft Windows [Version 10.0.22621.3593]
(c) Microsoft Corporation. All rights reserved.

C:\Users\VARSHINISRI\Downloads\COURSE>javac DuckingExceptionUsingThrows.java

C:\Users\VARSHINISRI\Downloads\COURSE>java DuckingExceptionUsingThrows
java.lang.ClassNotFoundException: Library
```