Java Exception Handling

how to solve the problem with run-time error

OODP 2022

Java Exceptions - Try...Catch

W3School

- The try statement allows you to define a block of code to be tested for errors **while it is being executed**.
- The catch statement handles errors.
- The try and catch keywords come in pairs:

```
try {
  //  Block of code to try
}
catch(Exception e) {
  //  Block of code to handle errors
}
```

```
public class MyExceptionClass {
  public static void main(String[ ] args) {
    int[] myNumbers = {1, 2, 3};
    System.out.println(myNumbers[10]); // error!
  }
}
```

```
Exception in thread "main"
java.lang.ArrayIndexOutOfBoundsException: 10
        at MyClass.main(MyClass.java:4)
```

What is the problem?

```
public class MyExceptionClass {
  public static void main(String[] args) {
    try {
      int[] myNumbers = {1, 2, 3};
      System.out.println(myNumbers[10]);
    } catch (Exception e) {
      System.out.println("Something went wrong.");
    }
  }
}
```

Modify the given example and run the code.

output: Something went wrong.

Finally : The finally statement lets you execute code, after try...catch, regardless of the result

```
public class MyClass {
  public static void main(String[] args) {
    try {
      int[] myNumbers = {1, 2, 3};
      System.out.println(myNumbers[10]);
    } catch (Exception e) {
      System.out.println("Something went wrong.");
    } finally {
      System.out.println("The 'try catch' is finished.");
    }
  }
}
```

```
Something went wrong.
The 'try catch' is finished.
```

The throw keyword : It allows you to create a custom error.

The throw statement is used together with an **exception type**. There are many exception types available in Java:

- ArithmeticException,
- FileNotFoundException,
- ArrayIndexOutOfBoundsException,
- SecurityException, etc:

```
public class MyThrowClass {
  static void checkAge(int age) {
    if (age < 18) {
      throw new ArithmeticException("Access denied - You must be at least 18 years old.");
    }
    else {
      System.out.println("Access granted - You are old enough!");
    }
  }
  public static void main(String[] args) {
    checkAge(15); // Set age to 15 (which is below 18...)
  }
}
```

What is the problem?

```
Exception in thread "main" java.lang.ArithmeticException: Access denied - You must be at least 18 years old.
        at MyThrowClass.checkAge(MyThrowClass.java:4)
        at MyThrowClass.main(MyThrowClass.java:11)
```

Useful Example with Exception

from **Jia Book**

```
import  java.io.*;
public class CopyTextFile {
  public static void main(String[] args) {
    if (args.length >= 2) {
      try {
          BufferedReader in = new BufferedReader(new FileReader(args[0]));
          PrintWriter out = new PrintWriter(new BufferedWriter(new FileWriter(args[1])));
          String  line;
          while ((line = in.readLine()) != null) {
            out.println(line);
          }
          out.flush();
          out.close();
      } catch (IOException e) {}
    }
  }
}
```

```
java CopyTextFile input.txt output.txt
```

Useful Example with Exception

from **Jia Book**

```
public class Maximum {
  public static void main(String[] args) {
    if (args.length >= 2) {
      int i1 = Integer.parseInt(args[0]);
      int i2 = Integer.parseInt(args[1]);
      System.out.println("The maximum of " + i1 + " and " + i2 + " is: " +
                         ((i1 >= i2) ? i1 : i2));
    } else {
      System.out.println("Usage: java Maximum integer1 integer2");
    }
  }
}
```

```
Java Maximum b a
NumberFormatException occured
```

What is the problem?

Try to add statements.
from **Jia Book**

```
public class Maximum2 {
  public static void main(String[] args) {
    if (args.length >= 2) {
      try {
        int i1 = Integer.parseInt(args[0]);
        int i2 = Integer.parseInt(args[1]);
        System.out.println("The maximum of " + i1 + " and " + i2 + " is: " +
                           ((i1 >= i2) ? i1 : i2));
      } catch (NumberFormatException e) {
        System.out.println("Invalid input value: " + e.getMessage());
        System.out.println("The input values must be integers.");
      }
    } else {
      System.out.println("Usage: java Maximum integer1 integer2");
    }
  }
}
```

```
Java Maximum b a
Invalid input value: For input string: "a"
The input values must be integers.
```