Custom Exception trong Java

Trong java đã định nghĩa sẵn các exception class như ArithaturesException, NullPulumException,...v...v... những exception này sẽ được throw ra khi gặp các trường hợp cụ thể nào đó. Ví dụ nếu lấy một số nào đó chia cho 0, bạn sẽ tạo ra một ArithaturesException.

Trong bài trước chúng ta đã học cách sử dụng throw để ném ra exception, ở bài này ta sẽ tiếp tục dùng đến nó để nén ra các custom exception mà ngừoi dùng tự định nghĩa.

Chúng ta có thể tự tạo ra các exception class của riêng mình và ném exception đó bằng cách sử dụng từ khóa **throw**. Những exception này được gọi là user-defined exception hoặc custom exception. Trong bài viết này, chúng ta sẽ học cách tạo một custom exception và throw nó vào một điều kiện cụ thể.

Để có thể học tốt bài này, bạn nên có kiến thức cơ bản về [try-catch block](https://freetuts.net/try-catch-trong-java-xu-ly-exception-2252.html" \t "_blank" \o "try-catch block) và [throw trong java](https://freetuts.net/throw-exception-trong-java-2275.html" \t "_blank" \o "throw trong java).

**1. Ví dụ về user-defined exception trong Java**

Hãy bắt đầu bằng việc xem xét một ví dụ dưới đây.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | /\* This is my Exception class, I have named it MyException   \* you can give any name, just remember that it should   \* extend Exception class   \*/  class MyException extends Exception{     String str1;     /\* Constructor of custom exception class      \* here I am copying the message that we are passing while      \* throwing the exception to a string and then displaying      \* that string along with the message.      \*/     MyException(String str2) {      str1=str2;     }     public String toString(){      return ("MyException Occurred: "+str1) ;     }  }    class Example1{     public static void main(String args[]){      try{          System.out.println("Starting of try block");          // I'm throwing the custom exception using throw          throw new MyException("This is My error Message");      }      catch(MyException exp){          System.out.println("Catch Block") ;          System.out.println(exp) ;      }     }  } |

Output:

|  |  |
| --- | --- |
| 1  2  3 | Starting of try block  Catch Block  MyException Occurred: This is My error Message |

Bạn có thể thấy rằng tôi đã tự throw ra một **user-defined exception** bằng câu lệnh throw new MyException("This is My error Message").

**Chú ý rằng:**

* User-defined exception nghĩa phải extend từ Exception class.
* Exception được ném ra bằng cách sử dụng từ khóa throw.

2. Một ví dụ khác về user-defined exception

Throw có thể sử dụng bên trong một phương thức hay một khối lệnh nào đó. Khác với throw, throws lại dùng ngay khi bạn khai báo một phương thức, cẩn thận đừng để bị nhầm lẫn nhé.

Ở ví dụ này, chúng ta đang ném ra ngoại lệ từ một method. Trong trường hợp này, chúng ta nên sử dụng throws nếu không bạn sẽ gặp lỗi biên dịch: “unhandled exception in method”.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31 | class InvalidProductException extends Exception  {      public InvalidProductException(String s)      {          // Call constructor of parent Exception          super(s);      }  }    public class Example1  {     void productCheck(int weight) throws InvalidProductException{      if(weight<100){          throw new InvalidProductException("Product Invalid");      }     }        public static void main(String args[])      {          Example1 obj = new Example1();          try          {              obj.productCheck(60);          }          catch (InvalidProductException ex)          {              System.out.println("Caught the exception");              System.out.println(ex.getMessage());          }      }  } |

**Output:**

|  |  |
| --- | --- |
| 1  2 | Caught the exception  Product Invalid |

Trên là một vài ví dụ và cách tự tạo ra một exception trong Java.