# Lab: Exceptions and Error Handling

In case **Java Zip File** format is required you need to make a .zip file of the package in which the main class is present, together with all other classes if there are any, and submit it in [Judge](https://alpha.judge.softuni.org/contests/3325).

## Number in Range

Write a program to **enter an integer** in a **certain range.** Until the number is not **valid**, enter a **new number** from the **console**. When the number is **valid** - end the program.

### Input

* Read a **range** - two numbers, separated by a space.
* On the next line, read the **String**.

### Output

* Print the range in the following format: "**Range: [{startIndex}...{endIndex}]**".
* When an **invalid number** is entered, **String** or the number is **out of range**, print "**Invalid number: {num}**".
* When the entered number is **valid**, print "**Valid number: {num}**".

### Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comment** |
| 10 20  5  xx  20 | Range: [10...20]  Invalid number: 5  Invalid number: xx  Valid number: 20 | Set the interval from the console and print it: **Range: [10...20]**. The first String is **5**, which is **invalid** and out of range. Enter a **new** String. The second **String** is **xx**, which is also **invalid**. Enter a **new** String. The third **String** is **20**, which is **valid**. End the program. |
| -5 50  hi  -6  -1 | Range: [-5...50]  Invalid number: hi  Invalid number: -6  Valid number: -1 |  |

## Square Root

Write a program that reads an integer **number** and **calculates** and **prints** its **square** **root** (with **2 digits** after the decimal point). If the number is invalid, print "**Invalid number**". In all cases finally, print **"Goodbye"**. Use try-catch-finally.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 9 | 3.00  Goodbye |
| 20 | 4.47  Goodbye |
| Xx | Invalid  Goodbye |
| -5 | Invalid  Goodbye |

## Enter Numbers

Write a method **readNumber(int start, int end)** that enters an integer number in a given range [**start…end**], **excluding** the numbers **start** and **end**. If an **invalid number** or a **non-number** text is entered, the method should **throw an exception**. Using this method, write a program that enters **10 numbers**: **a1, a2, … a10, such that 1 < a1 < … < a10 < 100**. If the user enters an invalid number, continue while there are 10 valid numbers entered. Print the array elements, separated with **comma and space** ", ".

### Hints

* When the entered input holds a non-integer value, print "**Invalid Number!**".
* When the entered input holds an integer that is out of range, print "**Your number is not in range {currentNumber} - 100**!".

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  3  4  5  6  7  8  9  10  11 | 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 |
| 1  2  1  3  p  4  5  6  7  8  9  10  11 | Your number is not in range (1 - 100)  Your number is not in range (2 – 100)  Invalid Number!  2, 3, 4, 5, 6, 7, 8, 9, 10, 11 |

## Fixing

This program is throwing an ArrayIndexOutOfBoundsException. Using your skills, fix this problem using a try-catch block. (without Judge)

