Security Injections @Towson

Security Injections, Java CS1 - Integer Error

1. Background | 2. Code Responsibly | 3. Laboratory Assignment
4. Security Checklist | 5. Discussion Questions

Code Responsibly - How can I avoid an Integer Error?

1. *Know your limits:* Familiarize yourself with the ranges available for each data type. Since the size of data types may be compiler and machine dependent, it is a good idea to run the IntegerCheck program shown below to show you the limits of each variable type.

```
import java.util.Scanner;
public class IntegerCheck {
  public static void main(String[] args) {
    Scanner scan = new Scanner(System.in);
    int i;
    byte b;
    short sh;
    long lon;
    System.out.println("Valid integer is between " + Integer.MIN_VALUE + " and "
+ Integer.MAX_VALUE );
    System.out.println("Valid byte    is between " + Byte.MIN_VALUE + " and " +
 Byte.MAX_VALUE );
    System.out.println("Valid short   is between " + Short.MIN_VALUE + " and " +
Short.MAX_VALUE );
    System.out.println("Valid long    is between " + Long.MIN_VALUE + " and " +
 Long.MAX_VALUE );
  }
}
```

2. *Choose your data types carefully:* Many programming languages include multiple data types for storing integer values. If you have any concerns about the integer values that you will be using, learn about the options available in the language you are using, and choose integer types that are large enough to hold the values you will be using.
3. *Validate your input:* Check input for ranges and reasonableness before conducting operations (more on this in future modules.)

Answer the following questions:

Question 1:

How can you avoid an integer error in your program?

- [x] ✓ Know the smallest and largest allowable values for each data type in the programming language you are using
- [ ] Always pick float or double as the data type for numbers
- [x] ✓ Check your input for reasonable values before conducting mathematical operations

(Hint: read the code responsibly section above to answer this question.)

Go To Next Section

This project is supported by the National Science Foundation under grants DUE-1241738 and DUE -0817267. Any opinions, findings, conclusions, or recommendations expressed are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Powered by a modified version of Class2Go