

12. Integer Conversion



Purpose: Performance tune a component of an application

The method `checkInteger` was found to be a bottleneck in a production application. Your task is to improve the performance of the `checkInteger` method by a factor of 3. The class `GenerateData` will generate three representative test data sets.

The `checkInteger` method has been included in the same class as the test harness. The functional requirements are;

1. input will be a `java.lang.String`
2. the `String` should represent a valid integer
3. the value of the integer must be greater than 10
4. the range shall be between 2 and 100000 inclusive
5. The first digit is 3.

This current implementation satisfies the requirements.

```
public boolean checkInteger(String testInteger) {
    try {
        Integer theInteger = new Integer(testInteger);
        return
            (theInteger.toString() != "") && //not empty
            (theInteger.intValue() > 10) && //greater than ten
            ((theInteger.intValue() >= 2) &&
            (theInteger.intValue() <= 100000)) && //2>=X<=100000
            (theInteger.toString().charAt(0) == '3'); //first digit is 3
    } catch (NumberFormatException err) {
        return false;
    }
}
```

Hints:

1. Make sure you identified the real bottleneck.
2. Make sure you are fixing the real bottleneck.
3. The profiler `-Xprof` is preferred for this task (why?).
4. Keep a copy of the code so you can roll back if need be.