

AP COMPUTER SCIENCE A – READING FROM THE CONSOLE

- 1) Compile and run the program `ConsoleReaderTest.java` and see what it does.
- 2) Look at the program:

```
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.IOException;
/**/
public class ConsoleReaderTest {
    /**/
    public static void main ( String [] arg ) {
        /**/
        int n;
        BufferedReader br;
        String s;
        /**/
        System.out.println();
        br=open();
        s=readString(br,"string = ? ");
        n=readInt(br,"    int = ? ");
        close(br);
        /**/
        System.out.println();
        System.out.println("string = " + s);
        System.out.println("    int = " + n);
        /**/
        return;
    }
    /**
     * Open a BufferedReader connected to the console.
     */
    private static BufferedReader open () {
        /**/
        BufferedReader rv;
        /**/
        rv=new BufferedReader( new InputStreamReader( System.in ) );
        /**/
        return rv;
    }
}
```

need to import classes that are not in java.lang

the console

AP COMPUTER SCIENCE A – READING FROM THE CONSOLE

```

/*
 * Read a non-empty String from the console.
 */
private static String readString ( BufferedReader br, String prompt ) {
    /**/
    String rv;
    /**/
    while ( true ) {
        rv=null;
        System.out.print(prompt);
        try { rv=br.readLine(); }
        catch ( IOException ioe ) {}
        if ( !rv.equals("") ) break;
    }
    /**/
    return rv;
}

```

The `IOException` (error) must be caught because it is not a `RuntimeException`. But no error will occur when connected to the console.

```

/*
 * Read an int from the console.
 */
private static int readInt ( BufferedReader br, String prompt ) {
    /**/
    int rv;
    String s;
    /**/
    while ( true ) {
        s=null;
        System.out.print(prompt);
        try { s=br.readLine(); }
        catch ( IOException ioe ) {}
        if ( s.equals("") ) continue;
        rv=0;
        try {
            rv=Integer.parseInt(s);
            break;
        }
        catch ( NumberFormatException nfe ) {}
    }
    /**/
    return rv;
}

```

If cannot convert `s` to an int

If can convert `s` to an int

It is not necessary to catch the `NumberFormatException` (it is a `RuntimeException`), but we need to here.

```

/*
 * Close the connection to the console.
 */
private static void close ( BufferedReader br ) {
    /**/
    try { br.close(); }
    catch ( IOException ioe ) {}
}

```

AP COMPUTER SCIENCE A – READING FROM THE CONSOLE

- 3) Make a program `ConsoleReaderTest2.java` starting with `ConsoleReaderTest.java` which reads the `string`, and which reads a positive `int n`. Use a `while` loop in `main` to read the positive `int`. If the `int` is not positive, then re-prompt the user. After that, print to the console the `string` `n` times. For example, when running the program, the output to the console should look like

```
string = ? steve  
int = ? 6
```

```
string = steve  
string = steve  
string = steve  
string = steve  
string = steve  
string = steve
```

where the bold print is user input.