CSC 243 - Java Programming

Strings and Text I/O

char, Character, and String

- A *char* is a primitive type
- A *Character* is a wrapper class for *char*
- A *String* is a class known to the compiler
  - Represents a sequence of characters
  - String literals are implemented as instances of String
  - String objects are immutable

Simple String Methods

- `length()`: Returns the number of characters in the string
- `charAt(index)`: Returns the character at the specified index
- `concat(s1)`: Returns a new string that concatenates this string with string s1
- `trim()`: Returns a new string with whitespace characters trimmed at both sides

Mutable Character Sequences

- StringBuilder is a non-thread safe, mutable sequence of characters
- StringBuffer is a thread safe, mutable sequence of characters
- Common operations
  - `append`
  - `insert`
  - `delete`
  - `replace`

Command Line Arguments

- The argument to `main` is an array of Strings entered on the command line
- The following program echos each command line argument on a new line

```
public class Echo {
  public static void main (String[] args) {
    for (String s: args) {
      System.out.println(s);
    }
  }
}
```

Command Line I/O

- Standard Streams
  - `System.in` standard input
  - `System.out` standard output
  - `System.err` standard error
- The standard input and standard output are used for text based console programs

Text Based Input

- `java.util.Scanner` is a simple text scanner object and can be constructed to use the standard input stream

```
Scanner input = new Scanner(System.in);
```

- Common java.util.Scanner methods
  - `close()`
  - `next()`
  - `nextLine()`
  - `nextInt()`
  - `close()`

File Based Input

- `java.util.Scanner` can also be constructed to read from a File object

```
File file = new File("filename");
Scanner input = new Scanner(file);
while (input.hasNextLine()) {
  System.out.println(input.nextLine());
}
```

Writing to a File

- java.io.PrintWriter can be used to create a file and write data to a text file

```
File file = new File("file.txt");
PrintWriter output = new PrintWriter(file)
output.print("Hello world");
output.close();
```