The String type

Strings
INFS1609/COMP1400 – Week 3

• The String type is part of Java's class library
• It provides special facilities to make handling character strings easy

Declaring Strings

• Declare a String variable called “greeting” and initialise it to “Hello world!”

String greeting = "Hello world!";

• If there is no initialisation, the default value is the empty string,"

String length

• Often, you need to know the length of a string

int len = greeting.length();

• The value of len will be 12 if greeting is “Hello, world!”
Concatenation

"Hello," + " world" + "!"

evaluates to:

“Hello, world!”

Concatenating Strings and other types

- When strings are concatenated with other types, the other types are automatically converted to strings

“The length of "" + greeting + "\" is " + len

evaluates to:

The length of “Hello, world!” is 13

Protected Character

- If you want to include a quote character inside a string, you must “protect” it by preceding it with a “\"

Printing to the terminal

- Java provides class library for input and output
- “print” outputs a string to the terminal
  
  System.out.print(stringExpression)

- “println” is the same except it appends a newline
  
  System.out.println(stringExpression)
**Concatenation in print statements**

```java
System.out.println("The length of " + greeting + "\" is " + len);
```

**Reading Strings**

```java
import java.io.*;

public String readString(String prompt) throws IOException {
    BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
    System.out.println(prompt);
    String line = reader.readLine();
    return line;
}
```

**Reading Numbers**

```java
import java.io.*;

public void readNum() throws IOException {
    BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
    System.out.println("Type an integer:");
    String line = reader.readLine();
    int i = Integer.parseInt(line);
    System.out.println("Type a floating point number:");
    String line = reader.readLine();
    float f = Integer.parseFloat(line);
}
```