What does this mean?

age = 25
CS 302 Objectives

Instruction and experience in the use of an object-oriented programming language. Program design; development of good programming style; preparation for other Computer Science courses.

Course Learning Objectives

Students successfully completing this course will:

- be able to write computer programs in a high-level programming language.
- analyze problems and formulate algorithms
- create robust, user-friendly, well-structured and well-documented Java programs
- read basic Java programs to determine their purpose
- have a basic understanding of how computers work

In short, read and write Java programs.
Course Website

http://pages.cs.wisc.edu/~cs302/
A Computer

What does this mean?

age = 25
What does this mean?

```c
int age;
age = 25;
```
What does this mean?

```c
int age;
age = 25;
age = age + 1;
```
What does this mean?

```cpp
int i;
int j;
i = 1;
j = 2;
i = j;
```
Does this mean the same thing?

```cpp
int i;
int j;
i = 1;
i = j;
j = 2;
```
Questions

(Degrees Fahrenheit – 32) x 5 / 9 = Degrees Celsius

What symbols have different meanings in Java?

What changes must be made to implement this equation in Java?
Demonstration
What does this do?

\[ k = i; \]
\[ i = j; \]
\[ j = k; \]
Online Course Pages

Information Sheet
Work at Home
Piazza (where to offer and get answers)
Coding Style
Labs (CodeLab & Team Labs)
Labs 15% of grade (75 lab points)

Team Lab
- meets in 1350cs or 1370cs on Tuesday or Wednesday each week.
- 14 team labs, each Team Lab is worth 5 lab points.

CodeLab
- Online tool for individually solving hundreds of small problems.
- 500+ exercises, each 10 exercises is worth 1 lab point.
Review

(Degrees Fahrenheit – 32) x  \(\frac{5}{9}\) = Degrees Celsius

What symbols have different meanings in Java?

What changes must be made to implement this equation in Java?
My List

- X vs *
- = vs assignment
- value is stored on the left hand side of assignment (=) operator
- Variables: name areas of computer memory, declare before use, declare type of data, initialize
- Variable names: start with letter, include letters numbers and _, but no spaces
- Conventions: camelCasing, spell out names
- Semicolon at the end of statements
Creating, Compiling, and Running Programs

Java Visualizer

Creating, Compiling, and Running Programs

Source code (developed by the programmer)

```java
public class Welcome {
    public static void main(String[] args) {
        System.out.println("Welcome to Java!");
    }
}
```

Bytecode (generated by the compiler for JVM to read and interpret)

```
... 
Method Welcome()
  0 aload_0
  ...
Method void main(java.lang.String[])
  0 getstatic #2 ...
  3 ldc #3 <String "Welcome to Java!">
  5 invokevirtual #4 ...
  8 return
... 
```

"Welcome to Java" is displayed on the console

```
Welcome to Java!
```
Block

{
  //beginning of a block of code

}  //end of a block of code
public class HelloWorld {

    public static void main(String[] args) {
        double degreesFahrenheit = 212.0;
        double degreesCelsius;

        degreesCelsius = (degreesFahrenheit - 32) * (5 / 9.0);

        System.out.println( degreesCelsius);
    }
}
Primitive Data Types

Commonly used:

int   whole numbers
double floating point numbers

Other integer data types: byte, short, long
Other floating point types: float
Programming Errors

Syntax
- compiler error

Logic
- program runs but provides wrong values

Runtime
- program crashes or throws exception
Output

From `java.lang` package
- automatically included in every Java program.

```
System.out.print( "hello");
System.out.println( "hello");
```