02: Getting to Know Java with JShell

What is JShell

- JShell is a feature of Java that allows us to program one line at a time.
- It's great for new programmers because it allows us to write one line and immediately see the results.
- Let's search for the JShell application.

Type your first line of Java

```
System.out.println("Hello campers!")
```

What will this line do?

Type your first line of Java

```
System.out.println("Hello campers!")
```

What will this line do?
The line “System.out.println” will output information the screen.

Math with Java

“$+$” is the symbol for addition.

```
2 + 2
```

What will this output?

Math with Java

“+” is the symbol for addition.

```
2 + 2
```

What will this output?
4

Math with Java

“-” is the symbol for subtraction.

```
7 - 4
```

What will this output?

Math with Java

“-” is the symbol for subtraction.

```
7 - 4
```

What will this output?
3

Math with Java

“*" is the symbol for multiplication.

```
3 * 5
```

What will this output?

Math with Java

“*" is the symbol for multiplication.

```
3 * 5
```

What will this output?
15

Math with Java

“/” is the symbol for division.

```
12 / 3
```

What will this output?

Math with Java

“/” is the symbol for division.

```
12 / 3
```

What will this output?
4

Math with Java

"/" is the symbol for division.

```
6 / 4
```

What will this output?

Math with Java

"/" is the symbol for division.

```
6 / 4
```

What will this output?
Wait a second??? Why isn't this 1.5? Any time you have a whole number divided by another whole number, you will always get another whole number.
Is this a bug or a feature?

Math with Java

```
2 + 2 * 5
```

What will this output?

Math with Java

```
2 + 2 * 5
```

What will this output?

12. Java understand order of operations. How might we write this to evaluate to be 20?

Math with Java

```
(2 + 2) * 5
```

What will this output?

Math with Java

```
(2 + 2) * 5
```

What will this output?
If you aren't sure about the order of operations, you can wrap things in parentheses.

Strings

Strings are text data. We signify that something is text using double quotes. Write your first name in double quotes.

```
"YourFirstNameHere"
```

What will this output?

Strings

Strings are text data. We signify that something is text using double quotes. Write your first name in double quotes.

```
"YourFirstNameHere"
```

What will this output?
It repeats your first name.

Strings

Concatenation is when we join two strings together. We use the "+" operator to join two strings. (It's the same symbol as addition.)

```
"YourFirstNameHere" + "YourLastNameHere"
```

What will this output?

Strings

Concatenation is when we join two strings together. We use the "+" operator to join two strings. (It's the same symbol as addition.)

```
"YourFirstNameHere" + "YourLastNameHere"
```

What will this output?
If it looks weird, that's because Java won't put spaces between the two strings.

Strings

Concatenation is when we join two strings together. We use the "+" operator to join two strings. (It's the same symbol as addition.)

```
"YourFirstNameHere" + " " + "YourLastNameHere"
```

What will this output?

Variables

Often, you will want to store the results of your expression. For that, we will need variables.

```
int x = 42
```

Because 42 is an integer, we use the keyword **int** as the variable type. Java requires variable types to store variables. We can now use x in equations.

Variables

What will this output?

```
x * 2
```

Variables

What will this output?

```
x * 2
```

84

Variables

What will this print?

```
x
```

Variables

What will this print?

```
x
```

42. Our variable didn't change! Consistency is a good thing.

Variables

We assign values to variables using the “=” sign.

```
x = x - 2
```

What will this output?

Variables

We assign values to variables using the "=" sign.

```
x = x - 2
```

What will this output?

40.

Variables

What will this print?

```
x
```

Variables

What will this print?

```
x
```

40. But I thought x was 42? Nope. We changed it using the “=” sign.

Types

Types are important to Java. All variables are stored with a type. **int** is for integers. **double** is for **double-precision floating point numbers.**

```
double pi = 3.1415
```

There is another type called **float** for **single-precision floating point numbers.** Don't use this.

Variables

What will this print?

```
6.0 / 4.0
```

Variables

What will this print?

```
6.0 / 4.0
```

1.5 Because we used a decimal point, Java believes these are floating point numbers. In Computer Science, 6 has a slightly different meaning than 6.0, even though they are the same value.

Types

Types are important to Java. All variables are stored with a type. **String** is for text data. Notice that "String" begins with an upper case letter.

```
String name = "YourFirstNameHere"
              + " " + "YourLastNameHere"
```

What will **name** become?

Thoughts on Types

There are many types in Java. Here are 4 common types.

- **int** is used for integer values (think whole numbers).
- **double** is used for floating point numbers. These numbers use a decimal point.
- **String** is used for text data, such as your name or a message.
- **boolean** is used to store **true** and **false**.

Thoughts on types

Now that they are off the screen, what are the four Java types that you should remember?

Thoughts on types

Now that they are off the screen, what are the four Java types that you should remember?
int, double, String, and boolean

Thoughts on types

Of those four types, which one is written with an upper case letter?

Thoughts on types

Of those four types, which one is written with an upper case letter? It's `String`. The other three are always written with a lowercase letter.

Fill the remaining time.

Geometry time.

- Imagine that we have a circle.
- The diameter of a circle is 5.
- The area of a circle is $\pi \times \mathit{radius}^2$.
- What is the area? Try to solve this using using the command line.

Answer

- The answer is approximately 19.6. Did you get this?
- You must first divide the diameter by 2.
- $pi * (5.0/2) * (5.0/2)$

Fill the remaining time.

Solve the following.

```
2 + 2 * 2 - 2
5 + 2 * 4
10 / 2 - 3
8 + 12 * 2 - 4
6 - 3 * 2 + 7 - 1
```

Evaluate.

```
Evaluate each of the following.
4  = 2 + 2 * 2 - 2
13 = 5 + 2 * 4
2  = 10 / 2 - 3
28 = 8 + 12 * 2 - 4
6  = 6 - 3 * 2 + 7 - 1
```

Evaluate.

The use of parentheses have the highest levels of precedence.

```
Evaluate.
 2 + 2  *  2 - 2
(2 + 2) *  2 - 2
 2 + 2  * (2 - 2)
(2 + 2) * (2 - 2)
```

Evaluate.

The use of parentheses have the highest levels of precedence.

```
4  =  2 + 2  *  2 - 2
6  = (2 + 2) *  2 - 2
2  =  2 + 2  * (2 - 2)
0  = (2 + 2) * (2 - 2)
```