

Worksheet 1

Together (Language basics)

Native “numbers”

Java (as well as most languages) is a bit like a calculator. Just type the inputs on the slides

Java (as well as most languages) is a bit like a calculator. Typing very simple calculations is already a Java expression. Try:

- `-3 * 2`
- `2147483647 + 1`

What happens? Why?

Now try the decimal numbers examples in `jshell`. What is different?

Functions

Let's play with `Integer.parseInt`. What does this return? (pay attention to the type)

- `Integer.parseInt("1")`
- `Integer.parseInt("-2")`
- `Integer.parseInt("one")`

What about edge cases?

- `Integer.parseInt("2147483647")`: still working?
- How about `Integer.parseInt("2147483648")`?

Now type `Integer.parseInt`. Press the tabulation key once. What happened? Press it a second time. Anything new?

Finally, let's give `Math.max` a try. Try to predict the output (type included) of:

- `Math.max(2, 7)`
- `Math.max(-2., 2)`
- `Math.max(7, 3, 6)`

Procedures

Try:

- `Thread.sleep(100)`
- `Thread.sleep(500)`
- `Thread.sleep(2000)`

What do you think does the argument represents? What about

- `System.exit(0)`
- `System.exit(1)`

Let's try operators

Modular arithmetic

Evaluate:

- `8 / 2`
- `7 / 2` (is the result what you expected?)
- `7. / 2`
- `7 % 2`

How could we write a test to detect even (and, conversely, odd) numbers with this?

Binary logic

What is

- `3 && 5` ?
- `3 & 5` ? Why?
- `3 ^ 5`

Pick a two-digits hexadecimal numbers. `&` it with 255. What is the result? Why?

Comparisons

- Evaluate `3 == 4`. What does `jshell` return ? Is that an expression or a statement?
- What about `4 == 4`?

Statements and Expressions

Casts

- Create a `short` variable representing the current year in your game (fancy a Renaissance FPS, anyone? please just don't use anything before the Roman empire collapse — 476).
- Now cast it to `byte`: what happens?

- Store the resulting value into a **byte** variable. Do you get your value back when casting this second variable back to **short**? What's going on?
- Create a **byte** variable representing the current month in the same calendar. What happens if you cast it to **short**? And if you cast a second time back to **byte** on to of the value obtained after the first **short** cast (sprinkle parenthesis liberally)?

Printing vs. evaluating

Print the name of your character (use `System.out.println`). Is the output of `jshell` any different from when you simply have it evaluate the variable containing it? How?

Can you copy the content of this variable to a new variable? Can you do the same with the printing statement (`String copy = System.out.println(characterName)`)? Why?

Assignments

- Create an **int** variable to represent the number of lives in your future game and initialize it to 3.
- Create a **String** variable to represent the name of your character without initializing it right away.
- Now set it to its appropriate value (sorry, “Zelda” and “Pikachu” are taken).
- Finally, set it to the value `null`.

Can you create a variable of type `void`? Why?

Conditional statements

- Choose an integer between 1 and 100 and store it to a variable called **secret**.
- Ask your neighbour to guess it and store what their answer as second variable called **playerInput**.
- Write an **if** conditional statement to print whether the initial secret number was less or more than their guess, or if they guessed right.

Individual practice

More about comparisons

Computing new values

Let us create a **String** value containing an at sign (`@`) immediately before the name of your character. Store the value you get to a new variable called **nick**.

- Look up the documentation for the `String` class and find how to compute a new string skipping the first characters from a given input string.
- Use it to print your character's name using the `nick` variable instead of the original variable containing it.
- Store the corresponding value to a new variable called `computedCharacterName`.

Some more comparisons

- What about `characterName == characterName`?
- And what about `characterName == computedCharacterName`?
- How can you explain what you observe?

Loops

The ASCII set

Use casts to perform conversions between a character and its code (both to and from).

Now, iterate over the `[0, 255]` range to print each character next to its code. You got yourself an ASCII table!

Collatz conjecture

Pick any integer number you want and save it as `int currentValue`. Now apply the current rule:

- if `currentValue` is even, divide it by 2
- otherwise, multiply it by 3 and add 1

Repeat a couple times. What do you observe ? Can you make a conjecture ? When will it stop ? Write a `while` loop verifying it.

Rot13

Cæsar's Rot13 was a clever (well at the time at least) scheme to cipher messages. From a given input message, he would replace all letters with the letter 13 ranks after in the alphabet (wrapping at 'z' of course).

Find a way to compute the replacement letter for a given initial letter.

Now declare a `String` variable holding a message and write a loop to print its Rot13 encoding (you want to look `.charAt()` up).