

Assignment 1

- Finish the hash table
 - from the previous practical
 - keys String
 - values Object
 - methods at least
 - Object get(String key)
 - void set(String key, Object value)
 - iterator over keys
- add the forEachValue method, which applies an operation (supplied as an argument) on each value in the table
 - design the method so a lambda expression can be used as the operation
 - design a suitable functional interface

Assignment 2

- create a simple calculator that reads the input in the Reverse Polish notation (postfix notation)

1 2 3 + +
=> 6

- reads from the std input
- prints out to the std output
- only the int type
- expressions are separated by a new-line

Tests...

Test 1

- What is printed out?

```
public class Greeter {  
    public static void main (String[] args) {  
        String greeting = "Hello world";  
        for (int i = 0; i < greeting.length(); i++) {  
            System.out.write (greeting.charAt(i));  
        }  
    }  
}
```

- A Hello world
- B nothing
- C something else

Test 2

- What is printed out

```
public class Slasher {  
  
    public static void main(String[] argv) {  
  
        String fullClassName = "cz.cuni.mff.java.io.Slasher";  
  
        String fileName =  
            fullClassName.replaceAll(".", "/") + ".java";  
  
        System.out.println("The class " + fullClassName +  
            " must be in the file " + fileName);  
  
    }  
}
```



Slides version PJ06.en.2019.01

This slides are licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/).