

# In-class exercises (lecture 9)

## Exercise 1: Timing

**Claim: Java NIO is faster than traditional IO (FileReader, BufferedReader)**

- Open today's code > timing package > Main.java
- Read the code then run main.
- Which approach was fastest on your machine?

**Type your answer in the chat**

## Exercise 2: File processing and command line args

**Write a program that reads country\_codes.csv.** Pick an approach to reading files from the iostreams package (MainTryWithResources.java example is recommended).

Instead of "hard coding" the file path, **allow the user to specify the file path.** It will look something like this:

```
public static void main(String[] args) {  
    String filePath = args[0];  
    // Read the file
```

Informally test your program in these two ways:

- Add the command line argument to IntelliJ:
  - Go to Run > Edit configurations
  - Select the file containing your main method from the files listed under "Applications" in the left pane. You may have to run the file at least once using the little green arrow button to get it to show up under Applications.
  - Add a file path to the "Program arguments" field e.g. country\_codes.csv
  - Click Apply then run the file
- Run the program from the command line
  - You must have run the file containing the main method within IntelliJ at least once.
  - In your system terminal (not IntelliJ):
    - `cd path/to/project_folder/build/classes/java/main`
    - `java packagename.FileName your_arg_here`

**What do you notice about the file path?**

## Exercise 3: Basic regex example

In today's code, open scanner > Regex.java. Notice that the last line in the while loop calls three different Matcher methods: matches, lookingAt, and find.

Run the code and try entering different strings. Take a look at the output from the three methods and see if you can work out what each one does.

## Exercise 4: File I/O and regex

*We may not have time to get to this in class. If that happens, use this exercise for practice outside of class time. Sample solutions will be posted right after class.*

Your task is to finish the country code processor started in exercise 2. Your program should do the following:

- The user enters the file path at the command line as a command line arg (not using Scanner).
- Read the file line by line.
- Use regex to put each country into a HashMap
  - Key: country code
  - Value: country name
- Final values should not contain any extra quotes
- Write tests