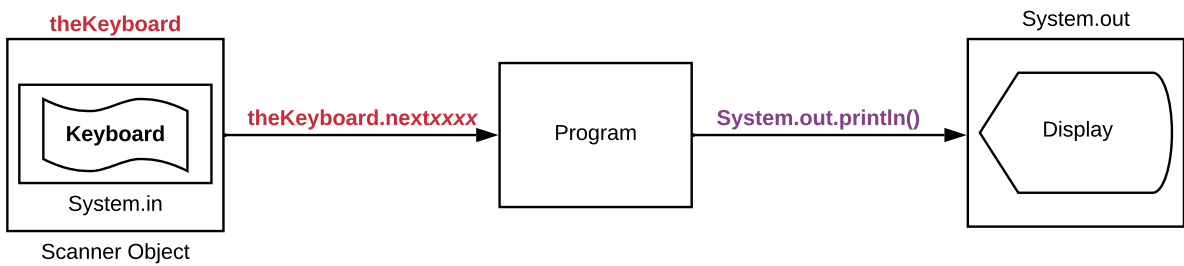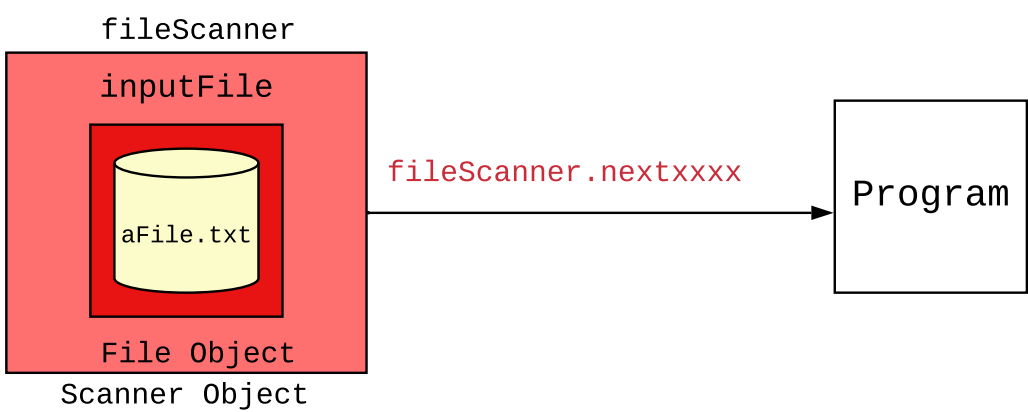# 01 - File_I/O Reading Notes

```
Scanner theKeyboard = new Scanner(System.in);
```



```
File inputFile = new File("path-to-the-File");
Scanner fileScanner = new Scanner(inputFile);
```


The path for the file is assumed to be relative to the high-level folder for application project.

In class, the high-level folder is the project folder (*01-name-name*).

Any file you want to use with a relative path should be in the high-level folder

"`aFile.txt`" - assumed to be in the high-level folder for the app

"`./data/numbers.txt`" - `.` start where you are (high-level folder)
                        `/data` then go to the data folder
                        `/numbers.txt` then the file called numbers.txt

When a File object or Scanner object is defined, several data structures are created in memory to support.

When you are done with a File or Scanner object for a File you should `close` the file to cause the data structures to be removed from memory.

If you don't `close` a File or Scanner for a File when you are done - the data structures and the memory is unusable - known as a "resource leak" or "memory leak".

File and Scanner objects MAY be closed automatically when a program end normally (`return` or `exit()`).

If a run-time error causes a program to terminate early - no automatic closing of files occurs.

Some IDEs will flag not closing the file as an error. One way to avoid this if to place your Scanner defintion in a "try-with-resource" statement:

```
try(Scanner fileScanner = new Scanner("input-file-path"))
```


"try-with-resources" will also ensure that any files listed within it are automatically closed in the case of an Exception in the program

It is considered good practice to close the file yourself when you are done or cause it to be closed when you are done using "try-with-resources" if an Exception occurs.

NEVER close a keyboard Scanner yourself. It can cause problems for other methods in your process. Java defined System.in not YOU so you should not close the Scanner for it. Also, Java may want to display messages regarding the end of processing and request input; if you close the keyboard Scanner it can't do that and an Excpetion is thrown.