cse15l-lab-reports

Lab Report #1

Command: `cd`

This navigates to the user’s home directory when no arguments are provided.

```
[user@sahara ~/lecture1]$ cd
[user@sahara ~]$
```

When provided with a directory as an argument, it will make that directory the working directory.

```
[user@sahara ~]$ cd lecture1/
[user@sahara ~/lecture1]$
```

When provided with a file, we see an error since cd requires a directory as its argument.

```
[user@sahara ~/lecture1]$ cd Hello.java
bash: cd: Hello.java: Not a directory
```

Command `ls`

When `ls` is used on its own, it prints out all files and directories in the working directory:

```
[user@sahara ~]$ ls
lecture1
```

When ls is provided a directory, it prints out the files and directories in the provided directory. However, it remains in the same directory, e.g home in this case:

```
[user@sahara ~]$ ls lecture1/
Hello.java  messages  README
[user@sahara ~]$
```

When ls is provided with a file, it lists that file - but only if it exists.

```
[user@sahara ~/lecture1]$ ls Hello.java
Hello.java
```

Command cat

When cat is not provided any arguments, it looks like it waits for input, and outputs whatever the keyboard sends. It only seems to stop when provided with an EOF or a SIGINT from the user.

```
[user@sahara ~/lecture1]$ cat
test <-- user input
test <-- terminal output
^C <-- ctrl+C (sig-int)
[user@sahara ~/lecture1]$
```

When cat is provided a directory, it handles it gracefully. cat does not seem to work with directories:

```
[user@sahara ~]$ cat lecture1/
cat: lecture1/: Is a directory
```

When cat is provided with a file, it outputs the contents of the file to the output stream (default to the terminal when called from the terminal).

```
[user@sahara ~/lecture1]$ cat Hello.java
import java.io.IOException;
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Path;

public class Hello {
  public static void main(String[] args) throws IOException {
    String content = Files.readString(Path.of(args[0]), StandardCharsets.UTF_8);
    System.out.println(content);
  }
```