**HOME TASK 2**

**Name: Daniella Lamptey**

**Matriculation Number: 7216548**

Q1**: What class and method would you use to read a few pieces of data that are at known positions near the end of a large file?**

A1: The `**Files.newByteChannel**` method in Java returns a **SeekableByteChannel** instance, enabling the capability to read or write bytes to a file with flexibility. The term seekable indicates that you can freely move the position within the file to any desired location before conducting read or write operations. In practical terms, utilizing **SeekableByteChannel** grants direct access to various sections of a file. By employing the seek () method, you can set the position to the beginning, middle, or end of the file, facilitating efficient reading, or writing operations on specific portions without the need to process the entire content sequentially.

Q2: **When invoking format, what is the best way to indicate a new line?**

A2: The **System.out.format()** method supports format specifiers, and the **%n** specifier is used to insert the system-dependent line separator. This ensures that your output is properly formatted with the correct newline character for the platform.

Q3**: How would you determine the MIME type of a file?**

A3: With the probeContentType method of the java.nio.file.Files class, we can get the type of the file that we gave in the path name which we passed in as the parameter to the method.

Q4: **What method(s) would you use to determine whether a file is a symbolic link?**

A4: By using the **Files.isSymbolicLink()** method to check if a file is a symbolic link.