

Assignment - 2

30 Paul Thomas S3 CSB

Write a java program to read characters from console using Buffered-Reader Class

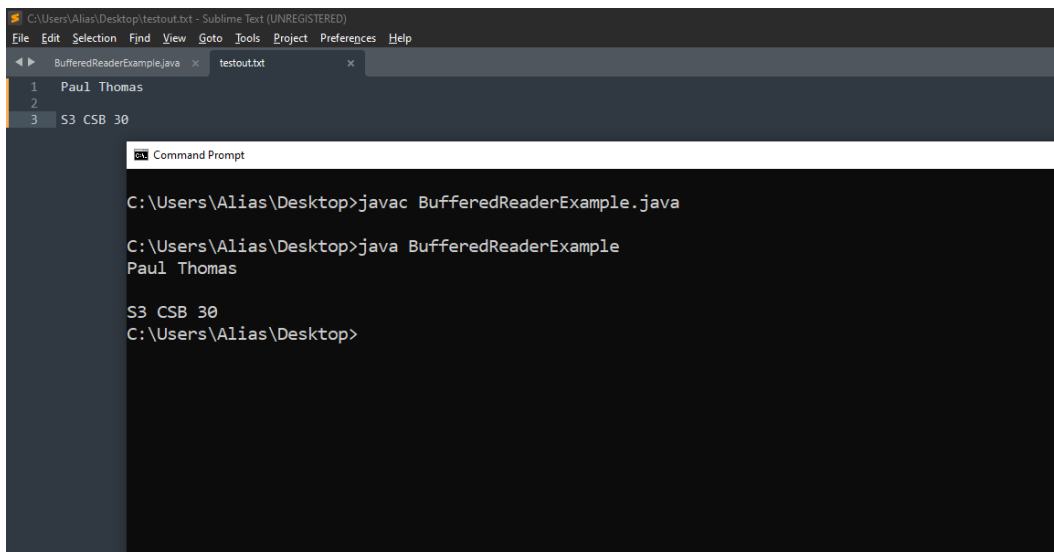
Program

```
import java.io.*;

public class BufferedReaderExample {
    public static void main(String args[])throws Exception{
        FileReader fr=new FileReader("testout.txt");
        BufferedReader br=new BufferedReader(fr);

        int i;
        while((i=br.read())!=-1){
            System.out.print((char)i);
        }
        br.close();
        fr.close();
    }
}
```

OUTPUT



The screenshot shows a Sublime Text editor window with the file 'BufferedReaderExample.java' open. The code in the editor matches the program shown in the previous block. Below the editor, a Command Prompt window is open, showing the compilation and execution of the program. The output of the program is 'Paul Thomas' followed by 'S3 CSB 30' on the next line.

```
C:\Users\Alias\Desktop>javac BufferedReaderExample.java

C:\Users\Alias\Desktop>java BufferedReaderExample
Paul Thomas

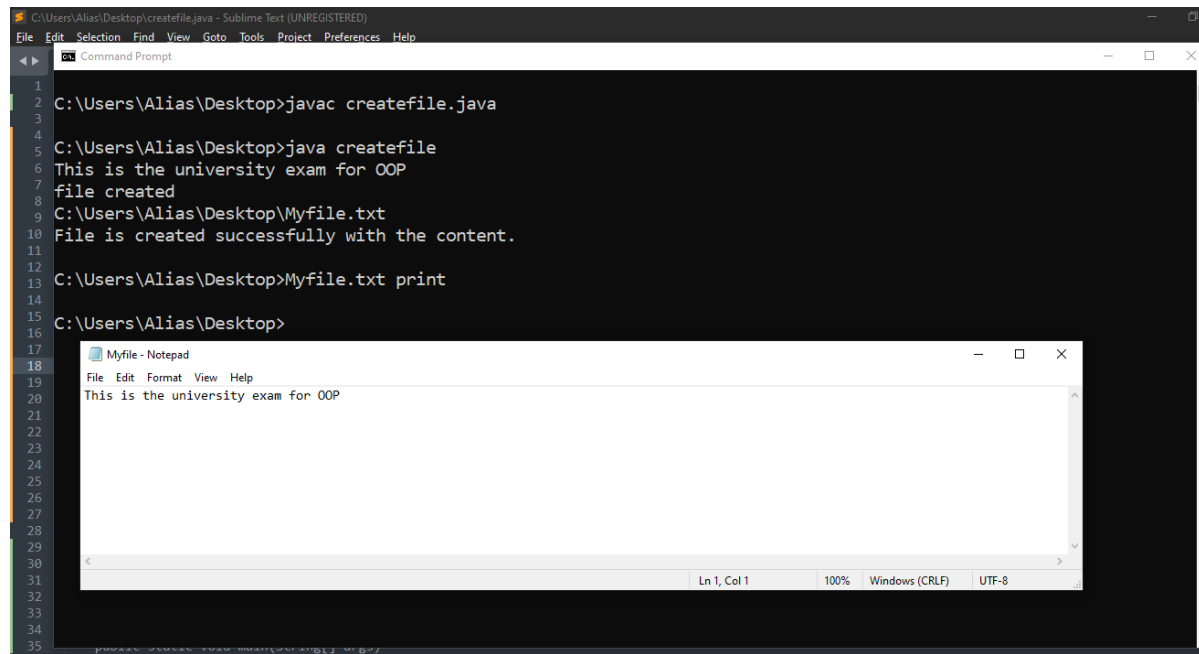
S3 CSB 30
C:\Users\Alias\Desktop>
```

Write a Java Program to create a new file called "Myfile.txt" and write the statement "This is the university exam for OOP"

Program

```
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
public class createfile {
    public static void main(String[] args) {
        String text = "This is the university exam for OOP";
        File file = new File("Myfile.txt");
        boolean result;
        try{
            result = file.createNewFile();
            FileWriter fWriter = new FileWriter("Myfile.txt");
            fWriter.write(text);
            System.out.println(text);
            fWriter.close();
            if(result){
                System.out.println("file created \n"+ file.getCanonicalPath());
                System.out.println("File is created successfully with the content.");
            }
            else{
                System.out.println("File already exist at location:
"+file.getCanonicalPath());
            }
        }
        catch (IOException e){
            e.printStackTrace();
            System.out.print(e.getMessage());
        }
    }
}
```

OUTPUT



The screenshot shows a Sublime Text editor window with the file `C:\Users\Alias\Desktop\createfile.java`. The code in the editor is as follows:

```
1 C:\Users\Alias\Desktop>javac createfile.java
2
3
4
5 C:\Users\Alias\Desktop>java createfile
6 This is the university exam for OOP
7 file created
8
9 C:\Users\Alias\Desktop\Myfile.txt
10 File is created successfully with the content.
11
12 C:\Users\Alias\Desktop>Myfile.txt print
13
14
15 C:\Users\Alias\Desktop>
16
17
```

Below the editor, a Command Prompt window is open, displaying the output of the Java program. The output is:

```
1 C:\Users\Alias\Desktop>javac createfile.java
2
3
4
5 C:\Users\Alias\Desktop>java createfile
6 This is the university exam for OOP
7 file created
8
9 C:\Users\Alias\Desktop\Myfile.txt
10 File is created successfully with the content.
11
12 C:\Users\Alias\Desktop>Myfile.txt print
13
14
15 C:\Users\Alias\Desktop>
```

Additionally, a Notepad window titled "Myfile - Notepad" is open, showing the content of the file `Myfile.txt`:

```
File Edit Format View Help
This is the university exam for OOP
```

The status bar at the bottom of the Notepad window indicates "Ln 1, Col 1", "100%", "Windows (CRLF)", and "UTF-8".

Write a simple program by extending appropriate class to demonstrate the working of threads in Java

```
class MultithreadingDemo extends Thread {
    public void run()
    {
        try {
            System.out.println(
                "Thread " + Thread.currentThread().getId()
                + " is running");
        }
        catch (Exception e) {

            System.out.println("Exception is caught");
        }
    }
}

public class Multithread {
    public static void main(String[] args)
    {
        int n = 8;
        for (int i = 0; i < n; i++) {
            MultithreadingDemo object
                = new MultithreadingDemo();
            object.start();
        }
    }
}

class MultithreadingDemo extends Thread {
    public void run()
    {
        try {
            System.out.println(
                "Thread " + Thread.currentThread().getId()
                + " is running");
        }
        catch (Exception e) {
```

```

        System.out.println("Exception is caught");
    }
}
}
public class Multithread {
    public static void main(String[] args)
    {
        int n = 8;
        for (int i = 0; i < n; i++) {
            MultithreadingDemo object
                = new MultithreadingDemo();
            object.start();
        }
    }
}

```

```

14 }
15 put C:\Users\Alias\Desktop>javac Multithread.java
16
17
18 C:\Users\Alias\Desktop>java Multithread
19 Thread 14 is running
20 Thread 18 is running
21 Thread 13 is running
22 Thread 15 is running
23 Thread 16 is running
24 }
25 Thread 19 is running
Thread 12 is running
Thread 17 is running

C:\Users\Alias\Desktop>

```

Write a Java program to demonstrate use of JLabel and JButton by adding them to a frame.

```
import javax.swing.*;
public class btnlabExample {
public static void main(String[] args) {
    JFrame f = new JFrame("Label and Button Example");
    JLabel l1,l2;
    l1=new JLabel("First Label.");
    l1.setBounds(50,50, 100,30);
    JButton b = new JButton("Click Here");
    b.setBounds(50,100,95,30);
    f.add(l1);
    f.add(b);
    f.setSize(400,400);
    f.setLayout(null);
    f.setVisible(true);
}
}
```

