

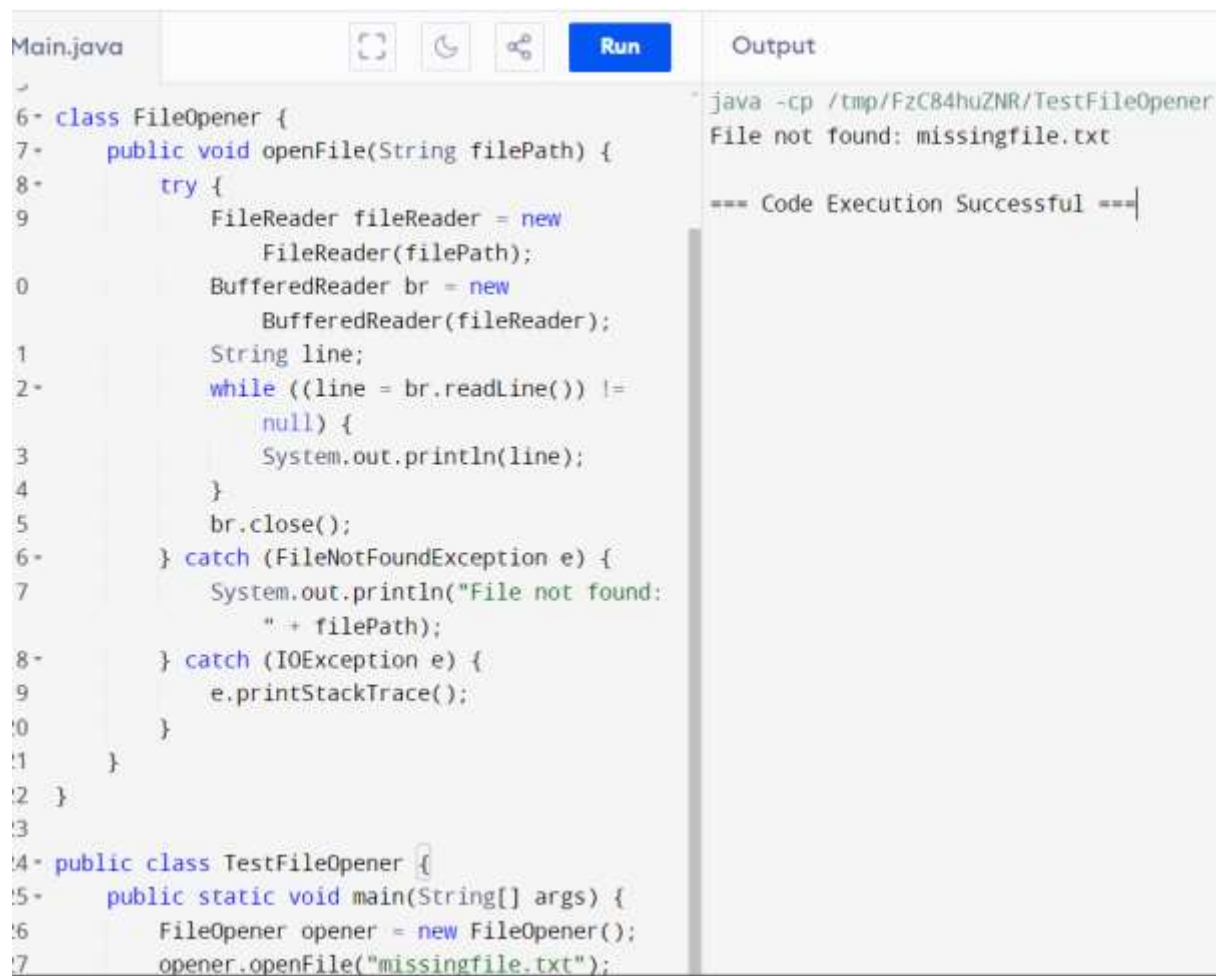
1.

ain.java	Run	Output
<pre>class Counter { public int count = 0; public void increment() { count++; } public int getCount() { return count; } } public class Test { public static void main(String[] args) { Counter counter = new Counter(); while (counter.getCount() < 10) { counter.increment(); } System.out.println("Counter reached:" + counter.getCount()); } }</pre>	<div>Run</div>	<pre>java -cp /tmp/Bloyp9HSbf/Test Counter reached:10 === Code Execution Successful ===</pre>

2.

Main.java	Run	Output
<pre>1 class Employee { 2 private String name; 3 public Employee(String name) { 4 this.name = name; 5 } 6 public String getName() { 7 return name; 8 } 9 } 10 public class Test { 11 public static void main(String[] args) { 12 Employee e = new Employee("John"); 13 String result=e.getName(); 14 System.out.println(result); } }</pre>	<div>Run</div>	<pre>java -cp /tmp/VTRjWZUfD4/Test John === Code Execution Successful ===</pre>

3.



The screenshot shows a Java IDE with a file named 'Main.java'. The code defines a 'FileOpener' class with an 'openFile' method that attempts to read a file line by line. It also includes a 'TestFileOpener' class with a 'main' method that creates an instance of 'FileOpener' and calls 'openFile' with the path 'missingfile.txt'. The IDE's 'Output' window shows the command 'java -cp /tmp/FzC84huZNR/TestFileOpener' and the output 'File not found: missingfile.txt', followed by a success message '=== Code Execution Successful ==='.

```
Main.java
6- class FileOpener {
7-     public void openFile(String filePath) {
8-         try {
9-             FileReader fileReader = new
                FileReader(filePath);
10-             BufferedReader br = new
                BufferedReader(fileReader);
11-             String line;
12-             while ((line = br.readLine()) !=
                null) {
13-                 System.out.println(line);
14-             }
15-             br.close();
16-         } catch (FileNotFoundException e) {
17-             System.out.println("File not found:
                " + filePath);
18-         } catch (IOException e) {
19-             e.printStackTrace();
20-         }
21-     }
22- }
23-
24- public class TestFileOpener {
25-     public static void main(String[] args) {
26-         FileOpener opener = new FileOpener();
27-         opener.openFile("missingfile.txt");
28-     }
29- }
```

```
Output
java -cp /tmp/FzC84huZNR/TestFileOpener
File not found: missingfile.txt

=== Code Execution Successful ===
```

4.

n.java



Run

Output

```
class FileOpener {
    public void openFile(String filePath) {
        try {
            FileReader fileReader = new
                FileReader(filePath);
            BufferedReader br = new
                BufferedReader(fileReader);
            String line;
            while ((line = br.readLine()) !=
                null) {
                System.out.println(line);
            }
            br.close();
        } catch (FileNotFoundException e) {
            System.out.println("File not found:
                " + filePath);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}

public class TestFileOpener {
    public static void main(String[] args) {
        FileOpener opener = new FileOpener();
        opener.openFile("missingfile.txt");
    }
}
```

```
java -cp /tmp/FzC84huZNR/TestFileOpener
File not found: missingfile.txt
```

```
=== Code Execution Successful ===
```