- **1. Write a program to create a new text file named test.txt.**

```java
package day10_Assessment;
import java.io.File;
import java.io.IOException;

public class CreateFileExample {
   public static void main(String[] args) {
      try {
         File file = new File("test.txt");
         if (file.createNewFile()) {
            System.out.println("File created: " + file.getName());
         } else {
            System.out.println("File already exists.");
         }
      } catch (IOException e) {
         System.out.println("An error occurred while creating the file.");
         e.printStackTrace();
      }
   }
}
```
Output:

File created: test.txt

- **2. Write a program to check whether a file exists at a given path.**

```java
package day10_Assessment;
import java.io.File;

public class CheckFileExists {
```

```java
    public static void main(String[] args) {
        String filePath = "test.txt";
        File file = new File(filePath);

        if (file.exists()) {
            System.out.println("File exists at: " +
file.getAbsolutePath());
        } else {
            System.out.println("File does not exist.");
        }
    }
}
```

Output:
File exists at: C:\path\to\your\project\test.txt

- **3. Write a Java program to write "Hello, World!" into a file using FileWriter.**

```java
package day10_Assessment;

import java.io.FileWriter;
import java.io.IOException;

public class WriteToFile {
    public static void main(String[] args) {
        try {
            FileWriter writer = new FileWriter("test.txt");
            writer.write("Hello, World!");
            writer.close();
            System.out.println("Successfully wrote to the file.");
        } catch (IOException e) {
            System.out.println("An error occurred while writing to the
file.");
```

```
            e.printStackTrace();
        }
    }
}
```

Output:
Successfully wrote to the file.

- **4. Write a program to read the content of a file line by line using BufferedReader.**

```java
package day10_Assessment;
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

public class ReadFileLineByLine {
    public static void main(String[] args) {
        try {
            BufferedReader reader = new BufferedReader(new
FileReader("test.txt"));
            String line;
            while ((line = reader.readLine()) != null) {
                System.out.println(line);
            }
            reader.close();
        } catch (IOException e) {
            System.out.println("An error occurred while reading the
file.");
            e.printStackTrace();
        }
    }
}
```

Output:
Hello, world!

- **5. Write a program to append a line of text to an existing file.**

```java
package day10_Assessment;
import java.io.FileWriter;
import java.io.IOException;

public class AppendToFile {
    public static void main(String[] args) {
        try {
            FileWriter writer = new FileWriter("test.txt", true);
            writer.write("\nThis is an appended line.");
            writer.close();
            System.out.println("Text appended successfully.");
        } catch (IOException e) {
            System.out.println("An error occurred while appending to the file.");
            e.printStackTrace();
        }
    }
}
```

Output:
Text appended successfully.

- **6. Write a program to count the number of lines, words, and characters in a file.**

```java
package day10_Assessment;
import java.io.BufferedReader;
import java.io.FileReader;
```

```java
import java.io.IOException;

public class FileCount {
    public static void main(String[] args) {
        int lineCount = 0;
        int wordCount = 0;
        int charCount = 0;

        try {
            BufferedReader reader = new BufferedReader(new FileReader("test.txt"));
            String line;
            while ((line = reader.readLine()) != null) {
                lineCount++;
                String[] words = line.split("\\s+");
                wordCount += words.length;
                charCount += line.length();
            }
            reader.close();

            System.out.println("Lines: " + lineCount);
            System.out.println("Words: " + wordCount);
            System.out.println("Characters: " + charCount);
        } catch (IOException e) {
            System.out.println("An error occurred while reading the file.");
            e.printStackTrace();
        }
    }
}
```

Output:
Lines: 2

Words: 6

Characters: 38

- **7. Write a program to copy content from one file to another using FileReader and FileWriter.**

```java
package day10_Assessment;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

public class CopyFile {
    public static void main(String[] args) {
        try {
            FileReader reader = new FileReader("source.txt");
            FileWriter writer = new FileWriter("destination.txt");

            int ch;
            while ((ch = reader.read()) != -1) {
                writer.write(ch);
            }

            reader.close();
            writer.close();

            System.out.println("File copied successfully.");
        } catch (IOException e) {
            System.out.println("An error occurred while copying the file.");
            e.printStackTrace();
        }
    }
}
```

Output:
Java file handling example.

- **8. Write a program that lists all the files in a directory.**

```java
package day10_Assessment;
import java.io.File;

public class ListOfFilesInDirectory {
    public static void main(String[] args) {
        File directory = new File("C:\\path\\to\\your\\folder"); // Change to your directory path

        if (directory.isDirectory()) {
            String[] files = directory.list();
            if (files != null && files.length > 0) {
                System.out.println("Files in directory:");
                for (String file : files) {
                    System.out.println(file);
                }
            } else {
                System.out.println("The directory is empty.");
            }
        } else {
            System.out.println("The specified path is not a directory.");
        }
    }
}
```

Output:
- Files in directory:
- test.txt
- source.txt
destination.txt

- **9. Write a program to filter and display only .txt files from a folder using FilenameFilter.**

```java
package day10_Assessment;
import java.io.File;
import java.io.FilenameFilter;

public class FilterTxtFiles {
   public static void main(String[] args) {
      File directory = new File("C:\\path\\wipro\\java\\text");
      FilenameFilter txtFilter = new FilenameFilter() {
         public boolean accept(File dir, String name) {
            return name.toLowerCase().endsWith(".txt");
         }
      };

      String[] txtFiles = directory.list(txtFilter);

      if (txtFiles != null && txtFiles.length > 0) {
         System.out.println(".txt files in the directory:");
         for (String file : txtFiles) {
            System.out.println(file);
         }
      } else {
         System.out.println("No .txt files found in the directory.");
      }
   }
}
```

Output:

.txt files in the directory:

test.txt

source.txt

- **10. Write a program to read a file using Scanner and display the tokens.**

```java
package day10_Assessment;
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

public class ReadFileTokens {
    public static void main(String[] args) {
        try {
            File file = new File("test.txt");
            Scanner scanner = new Scanner(file);

            System.out.println("Tokens in the file:");
            while (scanner.hasNext()) {
                System.out.println(scanner.next());
            }

            scanner.close();
        } catch (FileNotFoundException e) {
            System.out.println("File not found.");
            e.printStackTrace();
        }
    }
}
```

Output:
Tokens in the file:
Hello
World
from

Java