# File Handling

DESCRIPTION

**Project Objective:**

As a developer, write a program to read, write, and append to a file.

**Background of the problem statement:**

As a developer, write Java code to read, write, and append to a file.

**You must use the following:**

- Eclipse/IntelliJ: An IDE to code for the application.
- Java: A programming language.
- Git: To connect and push files from the local system to GitHub.
- GitHub: To store the application code and track its versions.

**Following requirements should be met:**

- The versions of the code should be tracked on GitHub repositories.
- The code should work properly.

**Output:-**

```
package com.anand.assistedproject.training;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Scanner;
import java.util.StringTokenizer;

public class FileHandlingproject{

	public static void main(String[] args) {
		@SuppressWarnings("resource")
		Scanner strInput = new Scanner(System.in);
	String choice,cont = "YES";

	while( cont.equalsIgnoreCase("YES") ) {
		System.out.println("\t\t Employee Information System\n\n");

			System.out.println("1 ===> Add New Employee Record ");
			System.out.println("2 ===> View All Employee Record ");
```

```
System.out.println("3 ===> Delete Employee Record ");
System.out.println("4 ===> Search Specific Record ");
System.out.println("5 ===> Update Specific Record ");

System.out.print("\n\n");
System.out.println("ENTER YOUR CHOICE: ");
choice = strInput.nextLine();

if( choice.equals("1") ) {
		try {
					AddRecord();
				} catch (IOException e) {
					// TODO Auto-generated catch block
					e.printStackTrace();
				}
} else if( choice.equals("2") ) {
		try {
					ViewAllRecord();
				} catch (IOException e) {
					// TODO Auto-generated catch block
					e.printStackTrace();
				}
} else if( choice.equals("3") ) {
		try {
					DeleteRecordByID();
				} catch (IOException e) {
					// TODO Auto-generated catch block
					e.printStackTrace();
				}
}  else if( choice.equals("4") ) {
		try {
					SearchRecordbyID();
				} catch (IOException e) {
					// TODO Auto-generated catch block
					e.printStackTrace();
				}
}  else if( choice.equals("5") ) {
		try {
					updateRecordbyID();
				} catch (IOException e) {
					// TODO Auto-generated catch block
					e.printStackTrace();
				}
}

System.out.println("DO YOU WANT TO CONTINUE :YES/NO");
cont = strInput.nextLine();
```

```
            }

        }

          public static void AddRecord() throws IOException {

                BufferedWriter bufferwriter = new BufferedWriter( new
FileWriter("records.txt",true) );
                @SuppressWarnings("resource")
                Scanner strInput = new Scanner(System.in);

                String ID, name, age, addr;

                System.out.print("Enter the Employee ID: ");
                ID = strInput.nextLine();
                System.out.print("Enter the Employee Name: ");
                name = strInput.nextLine();
                System.out.print("Enter the Employee Age: ");
                age = strInput.nextLine();
                System.out.print("Enter the Employee Address: ");
                addr = strInput.nextLine();

                bufferwriter.write(ID +","+ name+ ","+ age+ ","+ addr);
                bufferwriter.flush();
                bufferwriter.newLine();
                bufferwriter.close();

 }




        public static void ViewAllRecord() throws IOException {
                BufferedReader br = new BufferedReader( new FileReader("records.txt") );

                String record;

                System.out.println(" ------------------------------------------------------------ ");

                System.out.println("|   ID              Name                          Age
                Address                  |");

                System.out.println(" ------------------------------------------------------------ ");

                while( ( record = br.readLine() ) != null ) {

                        StringTokenizer st = new StringTokenizer(record,",");
```

```
                System.out.println("|   "+st.nextToken()+"      "+st.nextToken()+"
"+st.nextToken()+"                          "+st.nextToken()+"      |");

        }

        System.out.println("|                                                    |");

        System.out.println(" ------------------------------------------------------------ ");
        br.close();

    }



public static void DeleteRecordByID() throws IOException {
                @SuppressWarnings("resource")
                        Scanner strInput =  new Scanner(System.in);
                String ID, record;


                File tempDB = new File("records_temp.txt");
                File db = new File("records.txt");


                BufferedReader br = new BufferedReader( new FileReader( db ) );
                BufferedWriter bw = new BufferedWriter( new FileWriter( tempDB ) );


                System.out.println("\t\t Delete Employee Record\n");

                System.out.println("Enter your Employee ID: ");
                ID =  strInput.nextLine();


                while( ( record = br.readLine() ) != null ) {


                        if( record.contains(ID) )
                                continue;

                        bw.write(record);
                        bw.flush();
                        bw.newLine();

                }

                br.close();
```

```
                bw.close();

                db.delete();

                tempDB.renameTo(db);

        }


    public static void SearchRecordbyID() throws IOException {
                String ID,record;
                @SuppressWarnings("resource")
                        Scanner strInput = new Scanner(System.in);

                BufferedReader br = new BufferedReader( new
FileReader("records.txt") );

                System.out.println("\t\t Search Employee Record\n");


                System.out.println("Enter the Employee ID: ");
                ID = strInput.nextLine();

                System.out.println(" ------------------------------------------------------------ ");
                System.out.println("|   ID                Name
    Age                 Address                 |");
                System.out.println(" ------------------------------------------------------------ ");

                while( ( record = br.readLine() ) != null ) {

                        StringTokenizer st = new StringTokenizer(record,",");
                        if( record.contains(ID) ) {
                                System.out.println("|   "+st.nextToken()+"
    "+st.nextToken()+"              "+st.nextToken()+"                  "+st.nextToken()+"
|");
                        }



                }

                System.out.println("|                                                    |");
                System.out.println(" ---------------------------------------------- ");

                br.close();
                }
```

```
	public static void updateRecordbyID() throws IOException {
			String newName, newAge, newAddr, record, ID,record2;

			File db = new File("records.txt");
			File tempDB = new File("records_temp.txt");

			BufferedReader br = new BufferedReader( new FileReader(db) );
			BufferedWriter bw = new BufferedWriter( new FileWriter(tempDB) );

			@SuppressWarnings("resource")
				Scanner strInput = new Scanner(System.in);

			System.out.println("\t\t Update Employee Record\n\n");
			/**/
				System.out.println("Enter the Employee ID: ");
				ID = strInput.nextLine();
				System.out.println(" ---------------------------------------------------
----- ");
				System.out.println("|   ID                Name
	Age                 Address                 |");
				System.out.println(" ---------------------------------------------------
----- ");

				while( ( record = br.readLine() ) != null ) {

					StringTokenizer st = new StringTokenizer(record,",");
					if( record.contains(ID) ) {
						System.out.println("|   "+st.nextToken()+"
	"+st.nextToken()+"          "+st.nextToken()+"                  "+st.nextToken()+"
|");
					}

				}
				System.out.println("|                                                 |");
				System.out.println(" ---------------------------------------------------
----- ");

			br.close();
			/**/
			System.out.println("Enter the new Name: ");
			newName = strInput.nextLine();
			System.out.println("Enter the new Age: ");
			newAge = strInput.nextLine();
			System.out.println("Enter the new Address: ");
			newAddr = strInput.nextLine();

			BufferedReader br2 = new BufferedReader( new FileReader(db) );
```

```
                        while( (record2 = br2.readLine() ) != null ) {
                                if(record2.contains(ID)) {
                                        bw.write(ID+ ","+newName+ ","+ newAge+ ","+
newAddr);
                                } else {

                                        bw.write(record2);
                                }
                                bw.flush();
                                bw.newLine();
                        }

                        bw.close();
                        br2.close();
                        db.delete();
                        boolean success = tempDB.renameTo(db);
                        System.out.println(success);

                }


}
```