



OPPS MINI PROJECT  
COLLAGE MANAGEMENT

**GROUP NO. 22**

BY:

BT21CSE079 MAMULLA PRANAY

BT21CSE077 ROBIN K. BANSAL

BT21CSE106 ANANDAPU. CHAITANYA KUMAR

# ABOUT THE PROJECT

This project is about creating a Collage Management using Java. This project applies main concepts of opps such as inheritance, abstraction, polymorphism, Interface and File handling

## **Objective:**

To store the detail of various students,faculty,staff across various fields via the technique of file handling in JAVA.

**Through this mini project we will be able to:**

## **1 For Faculty Class**

- a. Add the details or record of a new Faculty.
- b. View the details or record of a particular Faculty.
- c. Remove the details of a Faculty if needed.
- d. Update any part of the record of a particular Faculty if needed or any wrong detail is entered by chance.
- e. To show Faculty list , salary ,department

## **2 For Student Class**

- a. Add the details or record of a new Student.
- b. View the details or record of a particular Student.
- c. Remove the details of a Student if needed.
- d. Update any part of the record of a particular Student if needed or any wrong detail is entered by chance.

- e. we also used gpa ,credit score

### **3 For Staff Class**

- a. Add the details or record of a new Faculty.
- b. View the details or record of a particular Faculty.
- c. Remove the details of a Faculty if needed.
- d. Update any part of the record of a particular Faculty if needed or any wrong detail is entered by chance. To show Faculty list , salary ,department

### **Code of Project:**

**We have similar code for three management**

**1)Faculty Class**

**2)Student Class**

**3)Staff Class**

Here is over view of total code:-

```
J Main.java > Main
1 > import java.util.InputMismatchException; ...
8
9 > class FileReadWriteException extends Exception { ...
21 > class InvalidInputException extends Exception { ...
33
34 // general operations
35 > interface GeneralOperations { ...
46
47 // faculty Management
48 > class FacultyManagement ...
104 > abstract class FacultyBase { ...
132 > class Faculty extends FacultyBase ...
229 > class FacultyOperations implements GeneralOperations { ...
377 > class FacultyIO { ...
572
573 // student Management36
574 > class StudentManagement { ...
629 > abstract class StudentBase { ...
656 > class Student extends StudentBase { ...
751 > class StudentIO { ...
947 > class StudentOperations implements GeneralOperations { ...
1095
1096 // staff Management
1097 > class StaffManagement { ...
1152 > abstract class StaffBase { ...
1175 > class Staff extends StaffBase { ...
1257 > class StaffIO { ...
1443 > class StaffOperations implements GeneralOperations { ...
1584
1585 💡 main function
1586 > public class Main { ...
```

We have used following libraries:-

```
1  import java.util.InputMismatchException;
2  import java.util.Scanner;
3  import java.io.BufferedReader;
4  import java.io.File;
5  import java.io.FileReader;
6  import java.io.FileWriter;
7  import java.io.IOException;
```

Exception code used is

```
class FileReadWriteException extends Exception {
    public String message;
    public FileReadWriteException(String message) {
        this.message = message;
    }
    @Override
    public String toString() {
        return message;
    }
}
class InvalidInputException extends Exception {
    public String message;
    public InvalidInputException(String message) {
        this.message = message;
    }
    @Override
    public String toString() {
        return message;
    }
}
```



```

        default:
            System.out.println("Unknown Operation");
    }

} catch (InputMismatchException error) {

    System.out.println("\n");
    System.out.println("Error: Invalid input given");
    System.out.println("");

} catch (InvalidInputException error) {

    System.out.println("\n");
    System.out.println("Error: " + error.toString());
    System.out.println("");

} catch (FileNotFoundException error) {

    System.out.println("\n");
    System.out.println("Error: " + error.toString());
    System.out.println("");

} catch (Exception error) {

    System.out.println("\n");
    System.out.println("Error: " + error.toString());
    System.out.println("");

} finally {
    sc.close();
}
}
}

```

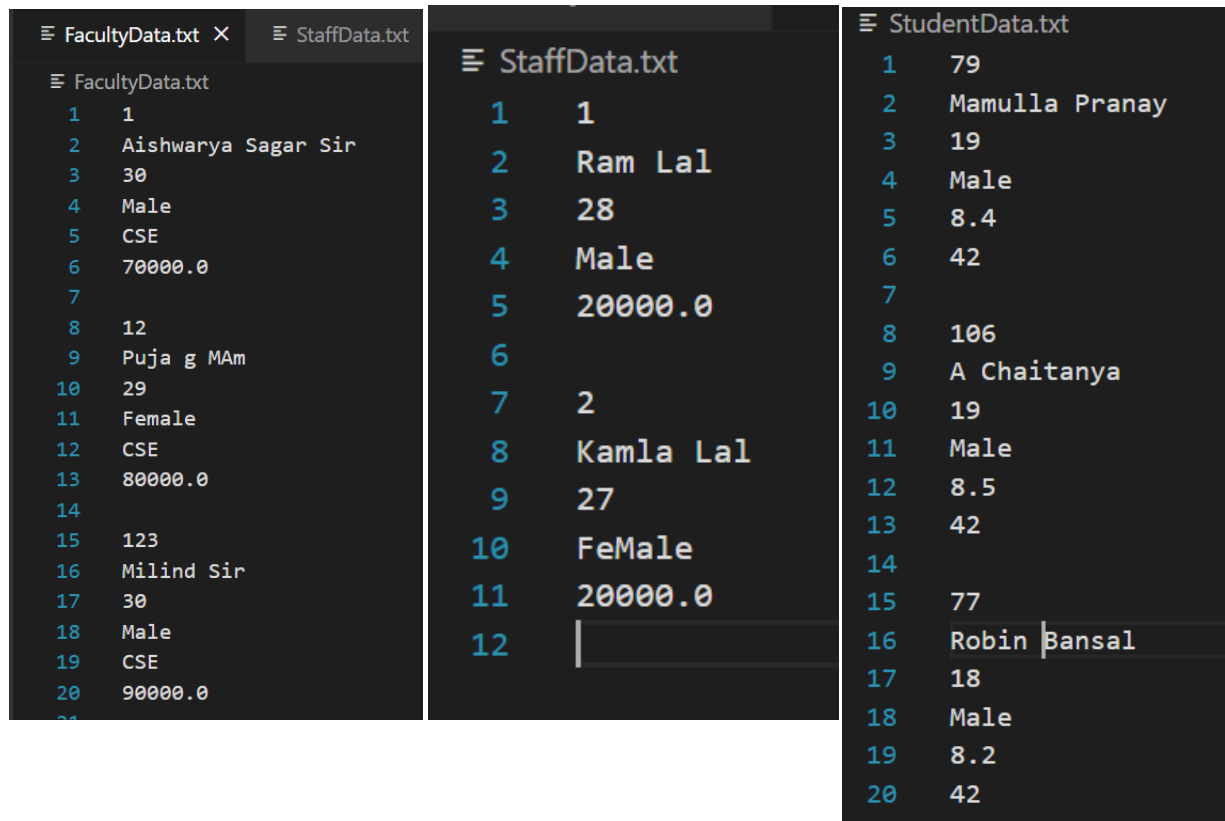
The data enter get stored in their respective .txt files

Like:-

for faculty data save in FacultyData.txt

for Student data save in StudentData.txt

for Staff data save in StaffData.txt



The image displays three separate text files, each containing a list of data entries. The first file, FacultyData.txt, lists faculty members with their IDs, names, salaries, and genders. The second file, StaffData.txt, lists staff members with their IDs, names, salaries, and genders. The third file, StudentData.txt, lists students with their IDs, names, ages, and genders. Each entry is numbered sequentially from 1 to 20.

```
FacultyData.txt
1 1
2 Aishwarya Sagar Sir
3 30
4 Male
5 CSE
6 70000.0
7
8 12
9 Puja g MAm
10 29
11 Female
12 CSE
13 80000.0
14
15 123
16 Milind Sir
17 30
18 Male
19 CSE
20 90000.0

StaffData.txt
1 1
2 Ram Lal
3 28
4 Male
5 20000.0
6
7 2
8 Kamla Lal
9 27
10 FeMale
11 20000.0
12

StudentData.txt
1 79
2 Mamulla Pranay
3 19
4 Male
5 8.4
6 42
7
8 106
9 A Chaitanya
10 19
11 Male
12 8.5
13 42
14
15 77
16 Robin Bansal
17 18
18 Male
19 8.2
20 42
```



## OUTPUT

```
.....Welcome to IIIT Nagpur Management System.....
```

1. Faculty Management
2. Student Management
3. Staff Management

```
Select Option: █
```

**After Selecting any one option:-**

```
----- Manage Faculty -----
```

1. Add Faculty
2. Update Faculty
3. Search Faculty
4. Delete Faculty
5. Show All Faculty
6. Exit

```
Select Option: █
```

# New data Added

```
----- Manage Faculty -----
1. Add Faculty
2. Update Faculty
3. Search Faculty
4. Delete Faculty
5. Show All Faculty
6. Exit

Select Option: 1

---- Enter Faculty Information ----
ID: 123
Name: ramu
Age: 30
Gender(Male/Female): Male
Department: cse
Salary: 200000

-----///----- New faculty added -----///-----

----- Manage Faculty -----
1. Add Faculty
2. Update Faculty
3. Search Faculty
4. Delete Faculty
5. Show All Faculty
6. Exit

Select Option:
```

# Updating existed data

```
----- Manage Faculty -----
1. Add Faculty
2. Update Faculty
3. Search Faculty
4. Delete Faculty
5. Show All Faculty
6. Exit

Select Option: 2

Enter faculty ID to update: 123

---- Edit Faculty Information ----
Name (ramu): ram
Age (30): 32
Gender (Male): Male
Department (cse): cse
Salary (200000.0): 200000

-----///----- Faculty data updated -----///-----

----- Manage Faculty -----
1. Add Faculty
2. Update Faculty
3. Search Faculty
4. Delete Faculty
5. Show All Faculty
6. Exit

Select Option:
```

**Thank you**

