**Concepts used in program**

**Class UserDetails**🡪

Defined instance variables to be used in the methods present in the class.

Setdata() method

Scanner function to take input from the user and store the values to the instance variables

“this” keyword to refer to instance variables.

getdata() method

System.out.println to make text to be visible on screen.

**Class option1🡪**

Retrievefiles() method

Java inbuilt file class, an abstract representation of file and directory pathname. It provides functions to create delete read write and search files inside the directory whose path is defined inside the method.

If-else conditional statement🡪 to run certain operations at the fulfilment of certain conditions. In this program user input are taken to select which operation to be performed next.

.isDirectory() this function returns true if the file mentioned by the abstract filename path in file class is directory or else returns false.

.listFiles() method inside file class it returns an array of Files denoting the files in a given abstract pathname if the path name is a directory else returns null.

Array.sort() in this program it is used without arguments to sort array of file names inside th e directory.

new Comparators() passed inside Array.sort() method to control the order of data inside that array.

.compareTo operator inside compare method this will compare strings lexicographically.

For-each loop is used to print all the elements stored in the array separately as a single entity, in this program it will print list of files that was stored in the directory that were earlier sorted and arranged in ascending order.

**Class option2 🡪**

Fileops() method

If-else statement here it will perform add, search and delete operation on the files present the directory as per the user choice. It also exit the program on user choice.

Try-catch block is used to handle printStackTrace() type exception, The printStackTrace() method of Java.lang.Throwable class used to print a throwable along with other details like class name and line number where the exception occurred.

For-loop to print the file name that was entered by the user if it is present inside the directory.

**Class Functions🡪**

Initialized instance variable func, this has been initialized to print the welcome page at first irrespective of the user input.

Method Welcomepage() this is just to print a welcome page

Method StartFunctions() t

Method Option3()

This method with if-else conditional statement and according to user input runs the operations in option 1 & 2, displays welcome page and exit the program.

**Main function()**

This only calls welcomepage function and take user in put to operate option 3.

**Algorithm**

Class UserDetails{

Declare string type instance variables name, mail and mob.

Method setData(){

Take user input for their full-name, Email ID and mobile no: and store it in variable name, mail and mob.

}

Method getData(){

Print all details entered by user in method setData()

}

}

Class Option1{

Initialize integer type variable retrieval with value 0.

Method RetrieveFiles(){

If retrieval=1🡪take user input for path of directory to be used.

Store that user input in string type variable named path.

Create object named dir of java class File with path as argument.

If dir.isDirectory() is true🡪

Store file names of dir in array named docs.

Run Array.sort on docs and comparator object.

Comparator comparing file names lexicographically

Print “file names in ascending order”

For-each loop over each element of array docs to print name of each file in the user given directory.

Else🡪 directory given by user is empty.

Class Option2{

Declare integer type variable opt2.

Method Fileops(){

If opt2 =1🡪take user input for the file name to be created.

try{

create new file with the user given name.

Print “File created”

}

Catch{

Print “Unable to create this file”

Invoking printStackTrace() exception.

}

If opt2 =2🡪 take user input for the file name to be deleted.

If file.delete() runs🡪 print “file deleted : ” [user given file name]

Else🡪print error occurred.

If opt2=3🡪

Create a file type directory defined by a path using java predefined File class.

Store files or folders inside array named flist of string type.

Initialize a local int type variable flag with value 0.

If flist is empty🡪print “Directory is empty”

Else 🡪 run a for-loop for all the files in the directory.

For file that matches with user given name🡪 print “file found” & turn flag=1 If flag=0🡪print “file not found”.

If opt2 =0🡪 exit program.

If opt2 has any other value 🡪 print “invalid entry”.

}

}

Class Functions{

Integer type variable func with initial value 2

Method WelcomePage{

Print welcome page data;

}

Method StartFunctions{

Create object for Class UserDetails

Call setData and getData methods of UserDetails class

Create object for Class Option1

Take user input

Run RetrievalFiles functions as per user input

Create object for Class Option2

Take user input

Run Fileops functions as per user input

}

Method Option3{

If func=2 🡪run WelcomePage method.

If func=1 🡪run StartFunctions method.

If func=0 🡪 display an ending message and exit the program.

For any other value of func 🡪 error display Invalid input.

}

}

Public class Project1{

Method Main(){

Create object for class Functions.

Run method WelcomePage.

Take user input for deciding value of variable func.

Run method Option3.

}

}