1. A Java FileExist Application accept the complete path of a file from the user and display whether the file exist or not.
2. A Java LineCount Application accept the complete path of a file from user and count the number of lines in that file and print it on the console.
3. A Java WordCount Application:

accept the path of a file from user and count the number of words in that file and print it on the console.

1. A Java grep Application: There is a useful utility in Unix called grep. You can use grep to list the lines in a text file that contain a certain strings. You will write the Java version of grep The user will input from the command line the search string and file name The output of your program will be the line numbers AND the lines in which the string occurs. If it does not find the string then a message is displayed to show that the string was not found.
2. create a class "MyDate". create 5 different objects of it. add them in a ArrayList. store ArrayList in file. Now open a file, read ArrayList and display all objects.
3. Write a Java program that prompts the user to enter a text file name and displays the number of vowels and consonants of that file. Use the set to store the vowels 'A', 'E', 'I', 'O', and 'U'.
4. "Write a Program in Java using class and object to input a Date in ddmmyyyy 8-digit format and print it in:

1) dd/mm/yyyy format

2) dd, month name, yyyy format

Example

1) Enter any date in 8 digits (ddmmyyyy) format: 02052013

Date in dd/mm/yyyy format = 02/05/2013

Date in dd, month name, yyyy format = 02 May, 2013

2) Enter any date in 8 digits (ddmmyyyy) format: 12111963

Date in dd/mm/yyyy format = 12/11/1963

Date in dd, month name, yyyy format =

12 November, 1963

3) Enter any date in 8 digits (ddmmyyyy) format: 252013

Wrong Input

4) Enter any date in 8 digits (ddmmyyyy) format: 29022013

1. Write a Java application that displays today's date. You want the date to be displayed in the following format: week day, month, date, and year.

More specifically you want your Java application to display the date in the following manner:

Tuesday, October 12, 2010

1. Design a Java application to generate 100 random Integers between 1 and 49 and write them in a text file named boy.txt, one number per line.

Then read the 100 Integers from the test.txt text file and place them into a Java collection. The application should then (in this order):

- eliminate duplicates

- sort the collection

- display the sorted collection on the screen

Name your Java file Result.java

1. Write a java program using class and object to accept a date in the string format dd/mm/yyyy. Check whether the date entered is valid or not. If it is valid, then input a certain number of days. Then calculate and print the future date after adding the given number of days if the future date is valid. If the date entered is invalid, then display a proper error message.

Example

Enter the date in (dd/mm/yyyy) format: 18/02/2014

Entered Date: 18/02/2014

Enter number of days after which future date is to be found: 2

Future Date : 20/2/2014

1. Write a program to create a file called CreateData.dat if it does not already exist. It will then write 100 integers created randomly into the file using binary I/O.

Hint

- The program creates the datafile if it does not exist, and overwrites it if it does.

- The program writes using binary I/O 100 random integers to the file.

1. Ask the user to enter a month (in numeric form), a day and a two-digit year.

Calculate the month times the day.

If the month times the day equals to the year, then display the date is magic. Otherwise, display the date is not magic.

1. Develop a program to maintain a list of homework assignments. When an assignment is assigned, add it to the list, and when it is completed, remove it from the list. You should keep track of the due date. Your program should provide the following services:

1. Add a new assignment.

2. Remove an assignment.

3. Provide a list of the assignments in the order they were assigned.

4. Find the assignment with the earliest due date.

1. you are to write a program that gives its users three basic options: reverse, convert, and compare. The program should ask the user to input the file name for the file in question (including the extension, .txt). If the user selects reverse the program should open the text file and read its content into a stack of characters. The program should then pop the characters from the top of the stack and save them in a second text file. The order in the second file should be the reverse of the order in the first file.  
     
   If the user selects convert, the program should open a text file and read its content into as queue of characters. The program should then dequeue the characters, convert it to uppercase, and store it in a second file.  
     
   If the user selects compare, the program ask the user for two files and should then open the two text files and read their contents into two separate queues. The program should determine if the files are identical by comparing characters in the queues. When two nonidentical characters are encountered, the program should display a message indicating that the files are not the same. Otherwise let the user know that the files are identical.  
     
   You should use your own test files to make sure the program performs as specified but you do not have to include them in your submission. You are at liberty to ask the user for the file name to be examined.