

Assignment 1

Part1: Implement a menu-driven Java program (like fib or factorial) to implement these input methods in java (command line args, Scanner, BufferedReader, DataInputStream, Console)

Part2: Implement a simple menu driven calculator in java to implement add, sub, mul, div, sqrt, power, mean, variance. Implement a separate Calculator class to include all related function inside that class. (mean calculation : program reads numbers from the keyboard, summing them in the process until the user enters the string "end". It then stops input & displays the avg. of numbers)

Assignment 2

Part 1: Write a Java program that declares two arrays named 'even' and 'odd'. Accept numbers from the user and move them to respective arrays depending on whether they are even or odd.

Part 2: Implement a Java function that finds two neighbouring numbers in an array with the smallest distance to each. The function should return the index of the 1st number.

Part 3: Write a Java program to convert an array into ArrayList and vice versa.

Assignment 3

Write a menu-driven Java Program to study the concepts of classes, and array of objects/arraylist, instance members, and constructors in Java.

Assignment description: Create a Student class describing attributes of a student like prn, name, DoB, marks etc. Create an array of objects of Student class and perform operations like Add students, Display, Search. (by prn, by name, by position), Update/Edit and Delete.

Assignment 4

Write a menu-driven Java Program for the following:

There are 52 cards in a deck, each of which belongs to one of four suits and one of 13 ranks.

Represent a deck of cards as an array of

Objects (*you may use the Vector class)

1. Use integers to encode the ranks and suits.
2. Have suitable default & parameterized constructors.
3. all data members to have private access.
4. The class 'Card' to have the following methods:

createDeck(), printCard(), printDeck (),sameCard(),compareCard(), sortCard(), findCard() which

searches through an array or vector of Cards to see whether it contains a specific card, dealCards() function: to print 5 random cards from the existing deck.

Assignment 5

Part 1 - Implement the generic Shapes class as an interface s so that we can implement concrete classes like circle, triangle, rectangle class from it.

Part 2 - Refer the excel sheet for salary calculation. In this exercise, take an abstract class which is defined below and develop two classes. The abstract class represents the basic building block for employees in a personnel database. The code is shown below:

```
abstract class Employee
private String name, address;
protected int basic salary;
```

The class contains three instance variables: an employee's name, address and basic yearly salary.

Aim of this exercise

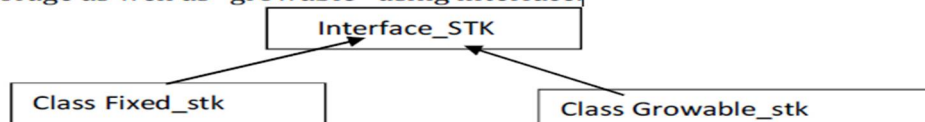
Generate concrete classes from an abstract class:

- Copy the code above
- into the file Employee.java in a folder. Make this class public.
- Write the code for a class NormalEmployee which extends the class above. This class should have a single method which calculates the monthly () salary for an employee. Compile the class.
- Write the code for a class BonusEmployee which extends the class Employee.java. This class describes an employee who has a monthly bonus added to their monthly salary. Compile the class
- Create a fourth file which tests the implementation of NormalEmployee and BonusEmployee files by creating suitable objects.

COMPANY NAME			
Address of the Company			
Pay Slip for Jun-21			
Name		UAN	
Employee ID		PF No	
Designation		ESI No	
Department		Bank Name	
DOJ		Bank A/C No	
Gross Wage	₹21000		
Total Working Days	30	Paid Days	29
LOP days	1	Leaves Taken	
Earnings		Deductions	
Basic Wage	₹9135	EPF	₹1370
HRA	₹3654	ESI / Health Insurance	₹152
Conveyance Allowances	₹1547	Professional Tax	₹0
Medical Allowances	₹1208	Loan Recovery	₹0
Other Allowances	₹4756		
Total Earnings	₹20300	Total Deductions	₹1523
Net Salary			₹18778
Employer Signature		Employee Signature	

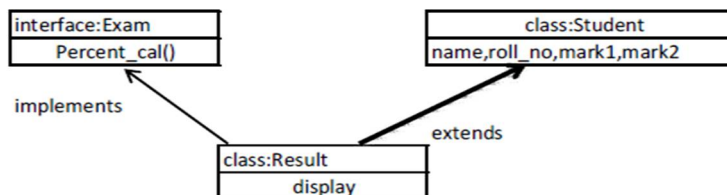
Assignment 6

Part 1: An implementation of IntStack (integer stack) that uses fixed storage as well as "growable" using interface.



Create a user defined package "pkg_Stack" where the interface is stored. The other two complete classes will need to import the package 'pkg_Stack' and then use it.

Part 2: Program to implement the following Multiple Inheritance.



Assignment 7

Assignment description: A rational number is a number in the form of $\frac{a}{b}$ where a and b are integers and $b \neq 0$. Rational numbers can be added, subtracted, multiplied, and divided. Write a Java application that will be able to add, subtract, multiply, divide, compare, convert to floating point, and find absolute value for rational numbers.

Your program should be written in Object Oriented Programming style. The program should accept two rational numbers from the user from keyboard (**Through command line arguments**) and output results of operations to console. Your program should solve operations efficiently and **be able to recover** from bad inputs. Use exception handling mechanism so as not to crash the program.

Example Inputs:

1234 / 5678 and 8765 / 4321
0 / 1 and 34 / 675
apple / 23 and 23 / 0

Assignment 8

Part 1:

Write a Java Program to find the factorial of 'n' integers (as command line arguments CLA). Write your own exception "FactorialException" to validate integer values to be in certain range. Sample call:
java ExceptionDemo 8 - 6 14 abcd 5

- 1) Static main method invokes another method "factorial()"
- 2) CLA which are strings but interpreted as integer values.
- 3) The user-defined exception class

FactorialException should have proper constructors / overridden toString() method to display exception message along with the wrong input number that had generated the exception.

NumberFormatException (ill-formed CLA)

And

Your Exception class

FactorialException (n<0 and n>15)

Part 2:

Define an exception class called "NOMATCHEXCP" that is thrown when the string from the keyboard is not equal to "India". Write a Demo program using try-catch block that shows the use of this user-defined exception. Class NOMATCHEXCP should have a parameterized constructor and the exception message generated should show the line number and the erroneous String that was inputted by the user.

Assignment 9

WAP demonstrating the Database Connectivity (MySQL/Oracle) in Java.

Assignment 10

Write a Java Program for Thread Creating and use of its various methods.

Assignment 11

Write a Java FX program for GUI.