ESE Examination Programming Lab – II
Problem Statement for Batch –S1

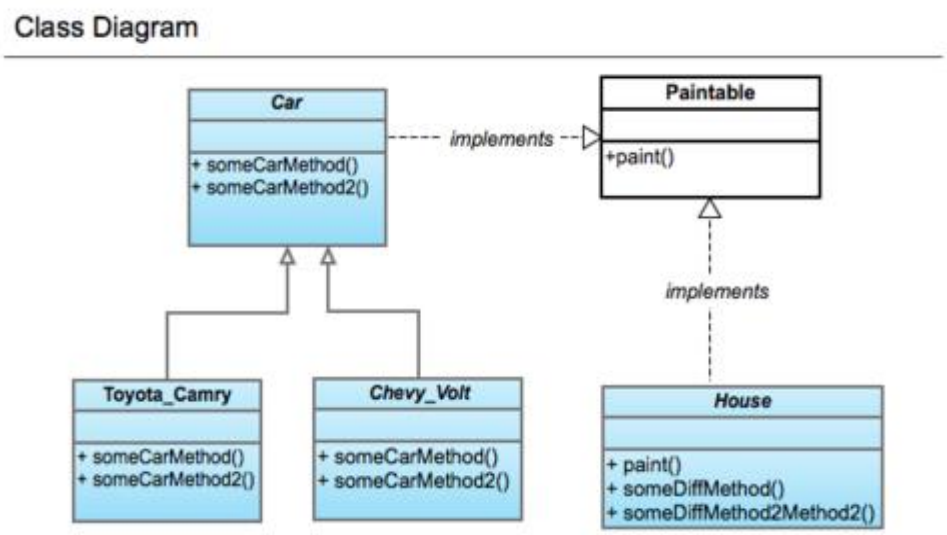
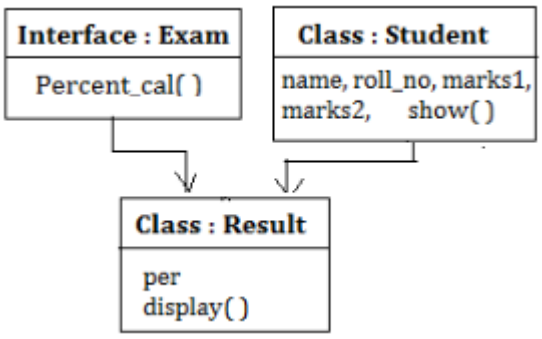
Dt. – 19/06/2021
Time: 9.30am – 12.30pm

NOTE: Assume suitable data if required

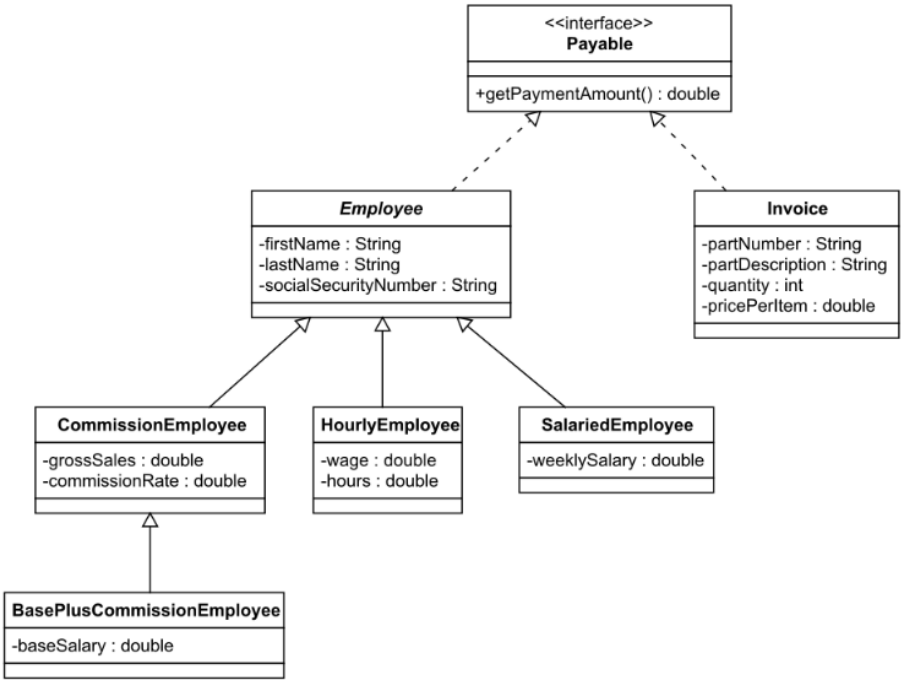
Enrollment Number	Problem Statement
2019BTECS00001 JOSHI PALLAVI SUHAS	Demonstrate how you are using access and non-access specifier/modifiers for solving following problem. Problem statement - Assume that the bank maintains two kinds of account of its customers, one called saving account and the other current account. The saving account provides compound interest and withdrawal facility but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge imposed. Create a class Account that stores customer name, account number and type of account. From this class Current-Account and Saving-Account to make them more specific to their requirements. Includes necessary methods in order to achieve the following task: a) Accept deposit from customer and update balance. b) Display the balance for account c) Display banks opening and closing balance and check this balance is same after performing all transaction made on that period. d) Compute and deposit interest. e) Permit withdrawal and update balance f) Check for minimum balance, impose penalty if necessary and update balance. g) Create and assign system generated account number to customer. f) Check no two customers contain same details, if same mark account as duplicate and make sure that duplicate type is not allowed to change. Also delete previous account and update all details in new account. (Show use of static, final, abstract and access specifier/modifiers). If required assume suitable data and mention your assumption in program - either in comment form or call method to print assumption on console.
2019BTECS00002 INGALE PARTHESH MANDAR	Suppose you are implementing java based software for Walchand College Library System and suggested to use user defined classes with static, non-static inner class, anonymous class strategy for implementing different functionality. For above scenario demonstrate different types of inner classes used for creating application and show how to access data member and methods in inner class. Also show how to access inner class method. First draw class diagram that represents application functionality using outer class and inner class (include all types of inner classes). Assume suitable data,

	functionality of library system.
2019BTECS00003 PRAJWAL BALASAHEB YADAV	Implement a student database application for CSE department such that objects elements are arranged in order, no duplicate entries allowed. Implement above scenario using suitable collection framework that fulfils above both conditions. Also based on user choice create sub collection for SY, TY and BTECH students.
2019BTECS00004 NALE YASH BABASAHEB	Write a program to handle stock of a wholesale grocery shop using interface. Maintain available stock of dairy product, wafers, vegetables, beverages separately. Check the current demand by customers and place an order if stock is not sufficient. Update the stock accordingly. Interface includes all common methods also stock information is related to interface not individual class assume that grocery shop handle only limited orders. Assume required and suitable data.
2019BTECS00005 SUTAR ASHISH AMOL	i) Write a program to write employee class information into external resource file and to read employee class state information from external resource file. Also show how different types of exception are arises during execution of file handling program. Exception - NullPointerException, IllegalArgumentException, IllegalStateException, EOFException, IndexOutOfBoundException, FileNotFoundException, SecurityException. ii) Implement a Java program using IO stream classes that takes bank-space separated data (one record per line, assume data – name, roll-number, mark, grade) from a text file and inserts this record into another text file if line number is even.
2019BTECS00006 SOLANKE SHITAL ARUN	i) Suppose in Number.txt file some random number are stored (-ve/+ve/fractional number). Using JAVA program find how many numbers are –ve, +ve and fractional number. ii) Design a java class to represent a bank account. Include the following members: Data Members- Name of the depositor, account number, type of account and balance in account. Methods - to assign initial values, to deposit an amount, to withdraw an amount after checking balance and to display account details.
2019BTECS00007 REVE DATTATRAY PARMESHWAR	i) Suppose you are implementing application for student class with methods (only three methods) 1-getdata() for getting student required information, 2-display() for show student information and 3-estimate() for calculating student percentage based in 5 subject marks. Let application want give information about student class (what it contains and how to use class) at the time of creating student class object if user

	want information flag is true. (Hint: anonymous class). ii) Write a java program to write a user input in external resource text file created through program using IO stream class.
2019BTECS00008 KHANDEKAR AMITKUMAR ASHOK	i) Create few threads in main function and implement this thread such that thread execution ends after exiting main thread. (Join method). ii) Implement a synchronized thread application for banking operation – debit and credit operation.
2019BTECS00009 MANE SONAL SUBODH	Write a java program for bank data database using Map interface of java collection framework. In bank costumer name may repeated with some unique constraint like first name, last name, date of birth but compulsory an account number is different. Also perform bank operation. Also display total balance available at bank open and at the time of closing bank and display all account name in descending order of available balance.
2019BTECS00010 PAWAR KAJAL JITENDRA	Create simple thread using Thread class and implementing Runnable class and show all information about thread like default name, priority, assign name using constructor, using setName, default priority, assign priority, call thread for execution, difference between calling thread and run method, current thread state etc. Also create multiple thread and show parent thread, child thread, their id, current state, priority, make one thread goes in sleeping state, thread is alive or not. Also show object sharing in number of runnable thread objects. Consider bank application credit and debit.
2019BTECS00011 NAVJYOT NETAJI SAKHALKAR	Suppose you are crating JAVA based student admission software for Walchand College of Engineering Sangli. From user or student software ask information about – name, JEE Mark, 12th standard mark, 10th standard mark. The college authority said to developer, there is no duplication in student record (assume name field is unique) and records are arranged in decreasing order of JEE Mark, 12th standard mark, 10th standard mark. Implement above situation using suitable collection framework, comparable and comparator interface.
2019BTECS00013 MOHD NIFASAT BEG	i) Implement following diagram using abstract class and interface. Assume suitable data member and methods.

	 ii) With suitable real world application demonstrate different types of inner classes and show how to access data member and methods in inner class. Also show how to access inner class method.
2019BTECS00014 SIDDHI BALKRUSHNA LOKHANDE	i) Create a user defined object array using Arrays class. Perform all operation of array using Arrays class. (Add, delete, search, modify, sort, comparing array, sub array creation (index position divisible by 2 or 7), copying array onto another array). Show how to traverse array using enumeration/iterator/enhanced loop. ii) Write a program to read and write primitive data type into/from external resource file.
2019BTECS00015 KHARAT SHRADDHA SANJAY	i) Implement following diagram to demonstrate abstract class and interface. 

	ii) Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
2019BTECS00016 RAUT AKASH SANJAY	Assume you are implementing JAVA based application for OLA cab organization. The organization authority provide requirement – 1) main class contain functionality- INQUIRY for availability SUV and SEDAN type, TAKE-ON-RENT, ESTIMATE/CHECK FARE and FEEDBACK. 2) For inquiry class contain take information for user and show available vehicle with type. 3) For TAKE-ON-RENT, its ask user for vehicle with driver or without driver. In both case methods contain its own local class and it takes information from user. 4) FEEDBACK is only allowed for valid customer based on unique class identity generated for each customer.
2019BTECS00017 ATTAR MUSKAN RAJU	i) Comparable and Comparator – write a java program to demonstrate comparable interface and comparator class. Distinguish two collections are different or not, based on single data member and multiple data member. Also use insertion order given by comparable or comparator. ii) Write a Java Program to define a class, describe its constructor, overload the Constructors and instantiate its object. Also demonstrate how to use inner static and non-static classes in defined class and in other external class.
2019BTECS00018 PATANGE RUSHIKESH KEDARNATH	Implement following scenario represented in diagram.



Suppose you are creating java based software SALARY application for a company. SalarySoftware class contain the main method. SalarySoftware class uses Employee as a subclass. There are three types of employee, 1) commission employee works on commission, 2) salaried employee works on weekly salary and 3) hourly employee works on clock hour basis. Each employee has a different salary. Invoice is the final document for representing final amount. If required assume suitable data.