Java Assignments

Day1 -2

Q1. Solve the following questions.

1. Accept 3 numbers from command line arguments. If number is prime, then print the table of the number. Other wise divide number by 10 and display output
2. Write a program to maintain student information. For each student, store following details: studid, name, m1, m2 and m3 (marks of 3 subjects).
   Accept information for 2 students and display it as follows.

```
Student Details:
_____________
Student Id
Name: Divya
M1 :   78
M2:    86
M3:    89
```

3. Write a menu driven program to maintain student information.
   Modify the Student class created in previous assignment.
   Add a member GPA in student class, add a function in student class to return GPA of a student

   calculateGPA()

   gpa=(1/3)*m1+(1/2)*m2+(1/4)*m3

   Create an array to store Multiple students.

   1. Display All Student

   2. Search by id

   3. Search by name

   4. calculate GPA of a student

   5. Exit

Day3-4

Q2. Inheritance, polymorphism, exception handling.

- Write a java application to maintain information for xyz sports club. The

  sports club wants to store employee details. For each employee store the following details:
  id, name, mobile no, emailid, Department, designation, Date of joining.

- Employees are of 2 types salaried employee or contract employee.
  If employee is a salaried employee, then store basic salary
  If it is contract employee, then store no. of hrs worked and per hour rate.

Calculate net salary by using following formula

- Salaried Employee

  Basic+DA+HRA-pf

  DA-10% of basic

  HRA – 15% of basic

  Pf – 12% of Basic

- Contract Employee

  Hr rate * no. of hrs worked

Also store information about members of club
For each member store id, name, mobile no, emailid, type of membership, amount paid.
For employee and members, id is autogenerated

Day 5

Q8. Write a Java program to store Student information studid, name, degree, email in a ArrayList at the time of exit write data in a file in csv format

Q9. Write another program to read data from file created in Q8. and store it in the ArrayList and dipsplay all the Students information on the screen.

Q10. Modify code written in Q8. By using ObjectOutputStream to write data to file

Q11. Modify code written in Q9. By using ObjectInputStream to read data from file and display it.

Day 6

Q12.

Design a file to store word and meaning. Write a multithreaded program to accept 5 words through command line arguments and search meaning of all words from a file.

Q13. Write a Java application that will accept two filenames (text files) as command line

arguments and use two threads to read contents from the two text files. Each of the threads should sleep for a random time after displaying filename with each line.

Day 7

Q4. Use employee class created in sports club assignment.

- Use HashSet to store multiple employee objects. Add required functions in the appropriate class.

  - No duplicate employee entries are allowed.

  - 2 Employee objects are same if their id is same

- Use TreeSet to store multiple Employee objects. Add required functions in the appropriate class.

Q5. Create account class for each account store account id, pin, name, balance, type,account opening date and status. For each new account assign status as “unlocked”. Do not accept status from user.

- If accountdata.dat file exists then read data from file and store it in ArrayList if file doesnot exists then create a empty ArrayList.
- Account object should match if account number and pin number matches.
- While withdrawing amount check that minimum balance should be 5000. Otherwise raise exception InsufficientBalanceException and handle it.
- For change pin allow only if old pin number matches. Maximum 3 attempts are there to enter correct pin number. Otherwise lock the account and change the status to lock and throw an exception LockedAccountException.
- While exiting from program store data in Accountdata.dat

Write a program to store list of Accounts objects in a ArrayList and perform following operations.

1. Add new Account
2. Close account
3. Withdraw amount
4. Deposit amount
5. Check balance
6. Change pin
7. Exit

2. Create a Product class with Product Id & Product Name. Write a program to accept information of 10 products and store that in Hash Table. Search a particular product in the Hash Table. Remove a particular product id and product name from the Hash Table.

Day 8

Q14. Write a java program to add, delete, modify, and display all products, display product based on productid, display all product with qty > given qty.

Create product table in data base to store data. (Use JDBC)

Q15. Create user table in database to store username, address, mobile and email. Add 10 records in the table

Write a java program to accept username and address from user check whether user exists in user table. If exists, then display details of user on the screen and if user not found then accept user details and store it in the table (Use JDBC)

Extra assignments for practice

1. Write a java application that will create and start two threads. One thread will read a text file (Number.txt) containing five positive integers one on each line. The second thread should calculate factorial of the number read by the first thread and print the message on the screen as "Number : numb Factorial : ans". The two threads should work in synchronization.
2. Write a java program to store List of Commonly found trees in each city.

   The List should be sorted on city names.
   Accept name of city and list of tree names from user. Perform following tasks:

   a) Find list of trees for a city
   b) Delete List of a particular city
   c) Add new entry in treemap
   d) Check whether city name already exists
   e) Display all city names and List of trees (Use Iterator and foreach)

- Add following options in Q5. assignment
  f) Add a new tree in existing list
  g) Accept tree name from user. Display all cities in which the tree exists

3. Create account class for each account store account id, pin, name, balance, type,account opening date and status. For each new account assign status as “unlocked”. Do not accept status from user.
    - If accountdata.dat file exists then read data from file and store it in ArrayList if file doesnot exists then create a empty ArrayList.
    - Account object should match if account number and pin number matches.
    - While withdrawing amount check that minimum balance should be 5000. Otherwise raise exception InsufficientBalanceException and handle it.
    - For change pin allow only if old pin number matches. Maximum 3 attempts are there to enter correct pin number. Otherwise lock the account and change the status to lock and throw an exception LockedAccountException.
    - While exiting from program store data in Accountdata.dat

Write a program to store list of Accounts objects in a ArrayList and perform following operations.

1. Add new Account
2. Close account
3. Withdraw amount
4. Deposit amount
5. Check balance
6. Change pin
7. Exit