**Day1: 21-Sep-2024**

* Write simple java program to get person name from command line and print Hello <Name> on console.
* Write menu driven java program for performing Add, Subtract, Multiplication and division of numbers.
* Write menu driven java program with below options
* Calculate simple interest
* Calculate compound interest
* Get 4-digit number from user, write a function to calculate sum of digits of the 4-digit number: Using loop and Using recursion
* Write a function for factorial and use the function for printing factorial series like 1 2 6 24 120………

**Day2: 23-Sep-2024**

* Write a Java program to allot grade student grade based on below conditions

Up to 50% pass class

Up to 60% second class

Up to 75% First class

Above 75% Distinction.

* Write a java program for printing patterns like below

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\* \*\*\*

\*\* \*\*

\* \*

\*\*\*\*\*

\*\*\*

\*

\*\*\*

\*\*\*\*\*

* Write a java program to sort an array of 10 numbers. Get array element from user and use any sorting method you like.

**Day3: 24-Sep-2024**

1. Write a Java program to create Student class with properties with rollNo, name, marks. Write printData method for printing Student. Create Default and parameterized constructor. Generate rollNo automatically using static field.

**Day4: 25-Sep-2024**

Write a menu driven program to add Student objects in an array of students.

1. Add Student

2. Print Student by rollNo

3. Print All students Sorted by Marks

0. exit

Write a menu driven program to add Employee objects in an array of Employees.

1. Add Employee

2. Print Employee by empId

3. Print All Employees Sorted by Salary

0. exit

Created overloaded static methods for printing an array of String and Integers separately. Put these methods in ArrayPrinter class and call from main

ex.

public static void printArray(int[] arr) {}

public static void printArrray(String[] arr) {}

**Day5: 26-Sep-2024**

**1. Write jav program to format given mobile number to expected format as shown below.**

Expected output +91-988-162-8598

Input +91-988-162-8598

Input +919881628598

Input 9881628598

Input 988162859 Error for less than 10 char

2. Implement toString() method in student class.

3.Write java program for billing of items purchased by customer. Create an Item class with itemCode, itemName, price and quantity along with getters and toString mthods.

Create an array of items holding stock of items. Create another array of ietms for customer. When customer buy the Item by item code add purchased item in an array. For every purchase create ietem using Parametrized constructor.

Menus.

1. Display stock

2. But item by item code

3. Display bill

**Day6: 27-Sep-2024**

**Revise all the Enum, Date and Assocaiation examples from class**

1. Create Enum for course separately with values as DAC, DESD, DVLSI. Use Course Enum and Date of Birth in Student class. Sort student by Date of Birth using after/before methods of java.util.Date

2. Create Person class with basic details like addharNumber, name, date of birth, address ( use Adress class for this). Find out whether the person is born before 15th August 1947.

**Day7: 28-Sep-2024**

**Topics Covered: Inheritance, abstract class, final variable, final methods, final class, upcasting, down casting, overriding, co-varient return type. Object class methods like getClass and toString. Wrapper classes boxing and un boxing.**

**Students should revise and try all above concepts**

1. Create account class acount number, accountName, dateOfOpening, amount.

Add below functionality

1. Generate account number on account creation.

2. Add abstract method as double applyInterest() in the account class

3. Create child classes as SavingAccount, CurrentAccount, DepostAccount. Override double applyInterest() in the child classes.

4. Interst percentage of Saving are 3%, Deposit 8%, and Current as 1%

5. Use upcasting and demonstarte Dynamic method dispatch.

6. Make is menu driven.

2. Create Shape class with abstract methods as

1. double calculateArea();

2. double calculatePerimeter();

Override these methods in child classes like Reactange and Circle.

Add additional method in Circle class as void draw() with message( not need to drwa real Circle)

Create circle object and have Shape reference to it.

Call the draw method of circle with specific circle reference after down casting from Shpae reference

Make is menu driven.

**Day8 and 9: 29-Sep-2024 and 30-Sep-2024**

**Topic Covered:** Method overriding, Polymorphism, final class, final var, final methods, Inteface, SAM type, Functional Interface, Default methods, Interafce extension, Interface implemntation.

Exception hierarchy, try, catch, finally, throw, throws, custom exception, multi catcg, try with resource, static block, class loading, many exceptions....

**Assignment 1:**

Create Depositable , Withdrwable , Muturable interfaces with methods. Implement these interfaces in acoount calss hierarchy.

1) Depositable I/F

double depoist( double)

2) Withdrwable

double withdraw( double)

3) Muturable

double calculateMatutyAmount( double)

Implement MinBalnceException for withdraw

**Assignment 2:**

Handle ListFullException exception for insrt method of FixedList ( refer example from class)

**Assignment 3:**

Create Drawable interface with draw() method in it. Implement the interface in Circle and Rectangle. Note. Just print message in draw methods.

**Assignment 4:**

Try overriding methods with co-varient return type

ex. Number add( Number a, Number b)

@Override

Integer add( Number a, Number b)

**Day 10: 01-Oct-2024**

**Topic Covered:** Collection Hierarchy, Map interface, ArrayList: add, rremove, Iterator, Advacnced for, for each , Collections.sort revrse iteration. ConcurrentModification Exception.

Assignment 1:

Create Fruit class with name, color, price and quantity and do following things

1. Implement toString() method

2. Add Constructors

3. Create ArrayList< Fruit> fruitList = new ArrayList<Fruit>();

4. Provide for adding fruit in the list

5. Remove fruit by index

6. Print whole list using advacned for

7. Print whole list using for each

8. Print whole list using forword iterator

9. Print whole list using backword iterator

Assignment 2:

Craete arraylist of 10 double numbers. Calculate sum of all numbers and print. Accept these numbers form user using do while loop.

Try indexOf, lastIndexOf methods of arraylist and go through Java doc for ArrayList methods

**Day 11: 03-Oct-2024**

**Topic Covered:** LocalDate, Program to interface not to implementaion. Convert arry to arraylist. Arraylist to Array. AddAll, Collections.sort(), Comprable and Comparator difference, Natural order, Custom order, ArrayList and LinkedList difference.

1) Create Account class with accNO, accName, dateOfOpening, balances( please note that you need to create new account class and not to use extsing account Hiraechy).

Use LocalDate for date of opening .

Validations:

1. Add validation for date of opening is not future date

2. balance should be more than 1000

Menu:

1. Add account

2. Print all accounts

3. Sort by account no desc

4. Sort by accName

5. Sort by Date of opening

6. Sort by balance desc

**7.** exit

2. Use existing fruit class to create list of fruits and provide below functionality

Menu:

1. Add fruit

2. Sort fruits by color ( natural order)

3. Sort by quantity

**4.** Sort by price

**Day 12: 04-Oct-2024**

**Topic Covered:** LinkedList, Vector, CopyOnWriteArrayList, contains and eqauls, Hascode, HashSet, LinkedHashset, Fail fast and fail safe iteraror.

1. Create simple List using LinkedList of Account implement below menu

Menus

1 Accept A/C details from user

2 Display all Accounts content using Enhanced For loop

3 Accept account No & display details or error message if account Not Found ( Create AccountNotFoundException)

4 Accept From A/C NO and To Accoun ID and to Funds transfer. ( check for min Bal Exception)

5 Accept A/C NO & remove A/C from List

6 Apply interest on all saving A/C

7 Sort accounts as per ascending A/C No.

8 Sort by accountOpeningDate

2. Create application for Job Portal and provide below options.

Create classes JobSeeker with registrationId, name, email, password, dateOfBirth, aadharId, phone number, graduationDate, degree(enum).

Validate user entred graduation date should be in past and date of birth can not be future

Validate for duplicate registration

validate phone number for 10 digit without characters

Menus

1 JobSeeker registration

//TODO Get all details and validate data if it is valid then add in list

2 Update email

//Get old and new email from user

3 Display all job seeker by theie degree type. take degree from user

4 Sort JobSeeker details as per email ( comparable)

5 Sort JobSeeker details as per dob ( Comparator)

6 Sort customer details as per graduationDate ( Comparator)

**Day 13:05/10/2024**

**Library Application**

1. Use HashMap , LinkedHamap and TreeMap ( Create separate tester for each type of Map)

Book details : title,bookType(enum),price,publishedDate ,authorName,quantity

Unique ID : book title

Get data from user and store Books in Hashset

Menus

1 Add book.

2 Display All books

3 Allot book to student ( quantity -1). Find by book title

4 Take book return ( quantity +1) Find by book title

5 Remove book

Handle exceptions for duplicate book entries and BookNotFoundException

**2)**

Create an application for School which will maintain data for all the school students in Hashmap.

RollNo is not unique, There can be mnore than 1 students in school with same roll No.

Division/Section is not unique in the school for 5A, 6B

**Unique key can be a comibination of standard, division and roll no**

Student class will have following properties

rollNo

standard

division

name

Menu:

Add student

Find student using unique mentioned above

Print all students

Note: once it works change the HashMap to TreeMap

**Day 14:07/10/2024**

1) complete the School assignment with Custom key approach as

discussed in class. Refer StudentKey.txt file

2)Practice lambda expressions for Function, BiFunction, Predicate, forEeach on collection

3) Replace all the comparators from Day 7 to Day 12 by lambda expressions and use forEach Iterable method instead of Enhanced for loop

**Day 15:08/10/2024**

**Topics:** Java Streams, char oriented stream and Byte oriented stream. Diff between Reader and writer. Buffered reader /writers. DOS, DIS, OIS, OOS. Serial Version UID and serializable interface

1) Check if the file object is directory or file. If its directory print all the files or directories in it.

2) Create a new file if the files is not present at specified path

3) Read text file using FileReader->Bufferered reader and print on console.

4) Read from one text file and write in other files. Use PrintWriter, Bufferedwriter, FileWriter for writing. User BufferedReader, FileReader for reading. Use try catch with resource

**Day 16: 09/10/2024**

**Topics:**  Stream API, IntegerStream, sorted, filter,map, reduce, summ, collect methods of stream. Optional class and methods

1) Create Apple class with weight, color and taste. Use Stream API for below things. Use method ref for println

Filter by weight

filter by color

filter by color and weight

filter by color, weight and taste

Sort by weight

Sort by color

Remove red apples

Removed Green apples

Convert to Set collection

2) create list of String and use Stream API for below ( use method ref for println

print distinct

print all with uppercase

print which starts with specific letter

concat all strings in the list

filter by ending with some text

**Day 17: 10/10/2024**

**Topics:**  DataOutputStream, DataInputStream, ObjectOutputStream, ObjectInputStream. Stream API- grouppingBy, partioningBy. Compartor.comparing, thenComparing, naturalOrder, reversOrder. Static import.

1) Use Book class from Library application and use DataOuputStream and DataInputStream to write and read books from **binary** file.

Write methods to save book in file and save many books in file. Pass book object and DOS to method for saving book. Resue the method to save muliple books.

Create methods to read one book and read many books until data is available on stream.

Print the data read from file suing forEach and method reference.

2) Create Printer class with serialNo, modelNo, price, printerType, manufacturingDate. Create Enum for PrinterType with values as LASER, INKJET, DOTMATRIX.

Store the printers objects in file using ObjectOutputStream.On program startup read all printers from file and store them in HashMap where serialNo will be key and printer object will be value.

1.Add Printer

2.Update Printer Price

3.Print all Printers

4.Display printers sorted by Prices (user comparing method of Comparator)

5.Filter by printerType ( use filter method of Stram API)

6.Remove a book

7. Show printer by PrinterType ( use goupingBy of Stream)

0.Save in file and Exit

3)Stream API assignments

1) Use existing Apple class and use partitionedBy method to sepate Green and Red apples.

2) Use comparing and thenComparing methos to sort apples by color and weight

3) Convert the list of apples in set using Collect method.

**Day 18: 11/10/2024**

**Topics:**  Externalizable, JVM, ClassLoaders, Metaspace, Method area, Garbage collector. Clonable . System.gc and Runtime class

1) Use Printer class from yesterday assignment and implement Externalizable in it. Perform below operations

Store the printers objects in file using ObjectOutputStream.On program startup read all printers from file and store them in HashMap where serialNo will be key and printer object will be value.

1.Add Printer

2.Update Printer Price

3.Print all Printers

4.Display printers sorted by Prices (user comparing method of Comparator)

5.Filter by printerType ( use filter method of Stram API)

6.Remove a printer

7. Show printer by PrinterType ( use goupingBy of Stream)

0.Save in file and Exit

2) Create multiple object objects fo Students and have multiple references. Make reference null and check what is the out come of GC by implementing finalize method of Object class.

3) Import cloning project and check how clone method works( Existing project)

4) Create simple thread to print sum of number between 1 to 100. Extend your thread class from Thread class and Override run method.