19BQ1A05K5
CSE-D

1. How to implement precedence rules and associativity in java language ? Give an example.

Java operator precedence

Operator precedence is nothing but the order in which the operators in an expression are evaluated

Now let us consider an example

```
int a = 12 - 4 * 2;
```

what will be the value of a whether it is 16 or 4 In such case operator precedence comes into picture.

when two operators share a common operand 4 in the above example in such case operator with highest precedence is evaluated first.

In java, the precedence of '*' is higher than that of '-'. Hence multiplication is performed first then next subtraction is performed finally the value of above expression is '4'.

Example for operator precedence

```
class precedence {
    Public static void main (String args[]) {
     int a = 20, b = 10, c = 1, result;
     result = a-++c-++b;
     System. out. println (result);
    }
}
```

output

7

The operator precedence of ++ is higher than - in the above example.

result = a - ++c - ++b; this can be written as

result = a - (++c) - (++b); The expression in parenthesis is evaluated first

Associativity of operators in Java.

If an expression has two operators with similar precedence, the expression is evaluated according to associativity (either left to right, or right to left) a = b = c; Here the value of c is assigned to b, then b value is assigned to a because the associativity of = is from right to left

Operators	precedence	Associativity
post fix inc & dec	++, --	left to right
Prefix inc & dec and unary	++, --, +, - ~, !	right to left
multiplicative	*, /, %	left to right
additive	+, -	left to right
shift	<<, >>, >>>	left to right
relational	<, <=, >, >= instanceof	left to right
equality	==, !=	left to right
Bit wise AND	&	left to right
Bit wise exclusive and inclusive OR	^, |	left to right
logic AND, OR, ternary	&&, ||,	left to right
assignment	=, +=, -=, *=, /= %=, &=, ^=, |=, <<= >>=, >>>=	left to right

2. Design a class that represents a bank account and construct the methods to

i) Assign initial values

ii) Deposit an amount

iii) Withdrawal amount after checking balance

iv) Display account number and Balance. Do you need to use static keyword for the above bank account program ? explain.

Code

```
import java.util.Scanner;
Public class BankAccount {
    Public static void main (String [] args) {
        Scanner in = new Scanner (System.in);
    BankAccount account = new BankAccount (1000);
        account.deposit (500);
        account.withdraw(50);
        System.out.println ("BankAccount" + account.getNumber ());
    System.out.println("Has a bal of " + account.getBalance ());
    }
    Private double balance;
    Private int accountNumber;
    Private static int lastAccountNumber = 0;

    Public BankAccount (double intialBalance)
    {
        balance = intialBalance;
        accountNumber = lastAccountNumber + 1;
        lastAccountNumber = accountNumber;
    }
```

```
Public void deposit (double depositAmount)
{
   balance += depositAmount;
}
Public boolean withdraw(double withdrawAmount)
{ if (withdrawAmount > balance) {
   System.out.println ("Insufficient funds!!");
    return false;
   } else {
    balance -= withdrawAmount;
    return ture;
   }
  }
  Public int getNumber ()
  {
   return accountNumber;
  }
  public double getBalance ()
  {
   return balance;
  }
}
```

Generally static key word is used to share data between the two objects or more

we need to use static key word in the above program because Account number is constant throughout the program it is not able to change it any where so it is necessary to use static key word in above program.

3. Define a class Electric Bill with the following specifications

class : Electric Bill

Instance Vaiable / data member :

String n - to store a name of Costomer

int units - to store the no. of units Consumed

double bill - to store the amount to paid.

member methods :

Void accept () - to accept the name of the coustmer and no. of units Consumed

Void Calculate () - to Calculate the bill as per the following

Number of units - Rate per unit

First 100 units - Rs. 2.00

Next 200 units - Rs. 3.00

Above 300 units - Rs. 500

A Subcharge of 2.5% charged if the no. of units consumed is above 300 units

Void Print () - TO print the details as follows :

Name of the Coustmer....

No. of units Consumed....

Bill amount....

write a main method to create an object of the class and call the above member methods.

Code

```
// to calculate the electricity bill
import java.util.*;
Public class electricity
{
Public static void main (string args [])
{
Scanner sc = new scanner (system.in);
int u;
double b;
string name, month;
System.out.println ("Enter your name");
name = sc.nextLine ();
System.out.println ("Enter the month");
month = sc.nextLine ();
System.out.println ("Enter the unit consumed");
u= sc.nextInt ();
System.out.println ("The name of the consumer-" name);
system.out.println("the month for which bill is paied"+ month);
system.out.println("the unit consumed by you for the month of
                    "+month+" is "+u);
if (u<=100)
{
b= u*1.80;
System.out.println ("the bill to be paid= "+b);
}
else if ((u>100) && (u<=300))
{
b= u*2.30;
```

```
System.out.println("the bill to be paid = "+b);
}
else if ((u>300)&&(u<=500))
{
b= u*2.80;
system.out.println("the bill to be paid = "+b);
}
else if (u>500)
{
b= u*3.50;
System.out.println("the bill to be paid = "+b);
}
else
system.out.println("wrong input");
}
```

4. Design a class to overload a function check() as follows:

(i) Void check (string str, char ch) - to find and print the frequency of a character in a string

Example:

Input — output

str = "success" number of s present is = 3

ch = 's'

(ii) void check (string s1) - to display only the vowels from string s1, after converting it to lower case,

Example:

Input:

s1 = "computer" output: o u e

Code

```
class characterVowel {
Public void checkstring str, char ch {
int c=0, code, i, s;
str = str. toLowerCase();
int len = Str. length();
for (code = 97; code<122; code++) {
  c=0;
  for (i=0; i< len; i++) {
    ch = str. charAt(i);
    s = (int) ch;
    if (s == code)
      c = c+1;
    }
```

```
ch=(char)code;
if (c!=0)
system.out.println("Frequency of "+ch+" is "+c);
}
}
Public void check (string s1){
int i;
char ch=0, chr=0;
for (i=0; i<s1.length(); i++){
  ch = s1.charAt(i);
if (characters.isUpperCase(ch))
chr = character.toLowerCase(ch);
if ((s1.charAt(i)=='a') || (s1.charAt(i)=='e') || (s1.charAt(i)=='i'
      || (s1.charAt(i)=='o') || (s1.charAt(i)=='u'))
  System.out.println(s1.charAt(i));
}
}
}
```