Name : Ahmad Baseer **Roll no: 124**

Section: C

OOP LAB ASSIGNEMENT

Task#01: Write a class BankAccount in which use an instance variable with a name balance. Write three methods having names of deposit(), withdraw() and display() and use these to manipulate and display balance of an account. Write methods to view and change the information of the Bank Account holder.

Source code :

```
package main;
import java.util.Scanner;
class BankAccount {
Scanner sc=new Scanner(System.in);
private double balance;
private String accountNum;
public BankAccount(String acctNum, double initialBalance)
{
accountNum = acctNum;
balance = initialBalance;
}
public void deposit(double amount)
{
double newBalance = balance + amount;
balance = newBalance;
System.out.println("balance after deposit: Rs" + balance);
}
public void withdraw(double amount)
{
double newBalance = balance - amount;
balance = newBalance;
System.out.println("balance after withdraw: Rs" + balance);
}
public String getaccountNum(){
return accountNum;
}
public double getBalance()
{
return balance;
}
public void AccountInfo(){
System.out.println("Accountnum: " + accountNum);
System.out.println("balance: " + balance);
}
public void ChangeAccountInfo(){
System.out.println("Do you want to change Account info:");
String input=sc.next();
if(input.equals("yes")){
System.out.println("Enter Accountnum");
String acc=sc.next();
accountNum=acc;
System.out.println("Enter balance:");
double bal=sc.nextDouble();
balance=bal;
}
else{
System.out.println("You cant change!");
}
}
}
public class Main {
public static void main(String[] args) {
BankAccount ba = new BankAccount("83743843",232132.323);
System.out.printf("The Account #%s has initial balance of
Rs%.2f%n",ba.getaccountNum(),ba.getBalance());
ba.deposit(2133221.21313);
ba.withdraw(2133.33);
ba.AccountInfo();
ba.ChangeAccountInfo();
ba.AccountInfo();
}
}
```

Output :

```
The Account #83743843 has initial balance of Rs232132.32
balance after deposit: Rs2365353.53613
balance after withdraw: Rs2363220.20613
Accountnum: 83743843
balance: 2363220.20613
Do you want to change Account info:
yes
Enter Accountnum
23213
Enter balance:
21331331
Accountnum: 23213
balance: 2.1331331E7
```

Task#02: Design a Class Calculator, Define All necessary methods you think in the Class. Create the instance of the Class and Use it. Use Math Class.

Source code :

```
package main;
class Calculator {
int add( int x ,int y){
return x+y;
}
int subtract( int x ,int y){
return x-y;
}
int multiply( int x ,int y){
return x*y;
}
double divide( int x ,int y){
return x/y;
}
int modulus( int x ,int y){
```

```
return x%y;
}
int max( int x ,int y){
return Math.max(x,y);
}
int min( int x ,int y){
return Math.min(x,y);
}
int sqrt( int x ){
return (int) Math.sqrt(x);
}
int cbrt( int x ){
return (int) Math.cbrt(x);
}
int pow( int x ,int y){
return (int) Math.pow(x,y);
}
int log( int x ){
return (int) Math.log(x);
}
int log10( int x ){
return (int) Math.log10(x);
}
int exp( int x ){
return (int) Math.exp(x);
}
double sin( int x ){
return Math.sin(x);
}
int cos( int x ){
return (int) Math.cos(x);
}
int tan( int x ){
return (int) Math.tan(x);
}
double asin( int x ){
return Math.toDegrees(Math.asin(x));
}
double acos( int x ){
return Math.toDegrees(Math.acos(x));
}
double atan( int x ){
return Math.toDegrees(Math.atan(x));
}

}
public class Main {
public static void main(String[] args) {
Calculator c1 = new Calculator();
System.out.println("addition: "+c1.add(34,256));
System.out.println("subtrction: "+c1.subtract(45,83));
System.out.println("multiply: "+c1.multiply(34,45));
System.out.println("divide: "+c1.divide(25,4));
System.out.println("modulus: "+c1.modulus(34,16));
System.out.println("max value: "+c1.max(23,29));
System.out.println("Square: "+c1.sqrt(64));
System.out.println("log: "+c1.log(10));
System.out.println("exp: "+c1.exp(76));
System.out.println("sin: "+c1.sin(90));
System.out.println("tan: "+c1.tan(45));
System.out.println("cos: "+c1.cos(0));
System.out.println("asin: "+c1.asin(1));
System.out.println("acos: "+c1.acos(0));
System.out.println("atan: "+c1.atan(1));
}
}
```

Output :

```
addition: 290
subtrction: -38
multiply: 1530
divide: 6.0
modulus: 2
max value: 29
Square: 8
log: 2
exp: 2147483647
sin: 0.8939966636005579
tan: 1
cos: 1
asin: 90.0
acos: 90.0
atan: 45.0
```

Task#03: Design a Class Car, Define All necessary methods you think in the Class. Create the instance of the Class and Use it.

Source code :

```
package main;
class Car {
String name(){
return "Corolla";
}
String colour(){
return "Black";
}
int model(){
return 2020;
}
int mpg(){
return 14000;
}
public String autoOrManual(){
return "manual";
}
}
public class Main {
public static void main(String[] args) {
```

```
Car c = new Car();
System.out.println("car name: " + c.name());
System.out.println("car colour: " + c.colour());
System.out.println("car model year: " + c.model());
System.out.println("car mileage: " + c.mpg());
System.out.println("Is the car is auto or manual? :" + c.autoOrManual());
}
}
```

Output :

```
car name: Corolla
car colour: Black
car model year: 2020
car mileage: 14000
Is the car is auto or manual? :manual
```

Task#04: Create a class Point, with two properties x and y. Write all the methods to manipulate the values of Point. Write a method that can check two Objects of Point class for Equality. Create an Array of 10 Point class Objects by random values and check Equality between them (i.e. if two objects have same x and y the program show there indexes).

Source code :

```
package main;
class point {
int x,y;
point(int x,int y){
this.x=x;
this.y=y;
}
boolean equalto(point p){
return (x==p.x && y ==p.y);
}
String tostring(){
return "x = " + x + "," + " y = " + y;
}
}
public class Main{
public static void main(String[] args) {
point p = new point(3,6);
point p1=new point(6,7);
System.out.println(p.equalto(p1));
point arr[] = new point[10];
for(int i =0;i<arr.length;i++){
point a = new point((int) (1+(4-1)*Math.random()),(int) (1+(4-1)*Math.random()));
arr[i] = a;
System.out.println(arr[i].tostring());
}
for(int i=0;i<arr.length-1;i++){
if(arr[i].equalto(arr[i+1])){
System.out.println("at index " + i  + " and " + (i+1) + " values of x and y are same" );
}
}
}
```

Output :

```
false
x = 1, y = 1
x = 1, y = 1
x = 1, y = 3
x = 2, y = 2
x = 3, y = 3
x = 2, y = 3
x = 2, y = 2
x = 2, y = 3
x = 1, y = 2
x = 1, y = 3
at index 0 and 1 values of x and y are same
```

Task#05: Create a Class Line. Line Objects will be created by Point Objects. Write a Constructor that can create a Line with 10 randomly created Point Objects. The Point Objects will be created in the main class.

Source code :

```
package main;
class point {
int x,y;
point(int x,int y){
this.x=x;
this.y=y;
}
}
class line {
line(point p,point s,String a){
System.out.println( a +" is created from point("+ p.x + ","+ p.y + ") to point ("+ s.x + ","+ s.y +")") ;
}
}
public class Main{
public static void main(String[] args) {
point arr[] = new point[10];
for(int i =0;i<arr.length;i++){
point a = new point((int) (1+(9-1)*Math.random()),(int) (1+(9-1)*Math.random()));
arr[i] = a;
}
line l1 = new line(arr[0],arr[1],"line 1");
line l2 = new line(arr[2],arr[3],"line 2");
line l3 = new line(arr[4],arr[5],"line 3");
line l4 = new line(arr[6],arr[7],"line 4");
line l5 = new line(arr[8],arr[9],"line 5");
}
}
```

Output :

```
line 1 is created from point(2,1) to point (5,6)
line 2 is created from point(5,6) to point (7,3)
line 3 is created from point(8,4) to point (8,6)
line 4 is created from point(8,1) to point (3,8)
line 5 is created from point(6,8) to point (8,2)
```

Task#06: Create a Class „Class‟. Create another class Student with necessary properties. Write all methods for Student class. Create an Object of Class „Class‟ with the help of 5 Student Objects. Use Random method where required. Write a method in Class „Class‟ that can search any Student by Roll No in the Class Object.

Source code:

```
package main;
import java.util.Scanner;
class Class{
Scanner sc=new Scanner(System.in);
Class(Student obj1,Student obj2,Student obj3,
Student obj4,Student obj5,String name){

if(name.equals(obj1.username)){
System.out.println("Roll no is:"+obj1.rollno);
}
if(name.equals(obj2.username)){
System.out.println("Roll no is:"+obj2.rollno);
}
if(name.equals(obj3.username)){
System.out.println("Roll no is:"+obj3.rollno);
}
if(name.equals(obj4.username)){
System.out.println("Roll no is:"+obj4.rollno);
}
if(name.equals(obj5.username)){
System.out.println("Roll no is:"+obj5.rollno);
}
}
}
class Student {
String username;
int rollno;
Student(String username,int rollno){
this.username=username;
this.rollno=rollno;
}
public String name(String name){
return name;
}
public int Id(int ID){
return ID;
}
}
public class Main{
static String name;
public static void main(String[] args) {
Scanner sc=new Scanner(System.in);
Student std1=new Student("Ahmad",124);
Student std2=new Student("Ali",132);
Student std3=new Student("Aoun",104);
Student std4=new Student("Faizan",143);
Student std5=new Student("Zemmam",106);
for(int i=1;i<=6;i++){
```

```
System.out.println("Enter name of Stduent do u want to check:");
name=sc.next();
Class cls=new Class(std1,std2,std3,std4,std5,name);
System.out.println("Do you want to continue checking? Yes/No:");
String ans=sc.next();
if(ans.equals("Yes")){
continue;
}
else{
System.out.println("As u Wish!");
break;
}
}
}
}
```

Output:

```
Enter name of Stduent do u want to check:
Ahmad
Roll no is:124
Do you want to continue checking? Yes/No:
Yes
Enter name of Stduent do u want to check:
Ali
Roll no is:132
Do you want to continue checking? Yes/No:
Yes
Enter name of Stduent do u want to check:
Aoun
Roll no is:104
Do you want to continue checking? Yes/No:
Yes
Enter name of Stduent do u want to check:
Faizan
Roll no is:143
Do you want to continue checking? Yes/No:
Yes
Enter name of Stduent do u want to check:
Zemmam
Roll no is:106
Do you want to continue checking? Yes/No:
No
As u Wish!
```

Task#07: Create four classes; Shape, Circle, Square and Rectangle. Create a method Draw in each class. Create an Object of Shape class that can call the method Draw and it will call automatically the Draw method of each class. The text of Draw method e.g. Circle is "This is a Circle". Hint: The Object of Shape contains the Objects of all three classes.

Source code :

```
package main;
class shape {
void draw(){
shape s1 ,s2,s3;
s1=new circle();
s1.draw();
s2 = new rectangle();
s2.draw();
s3 = new square();
s3.draw();
}
}
class circle extends shape{
void draw(){
System.out.println("This is circle!");
}
}
class rectangle extends shape{
void draw(){
System.out.println("This is rectangle!");
}
}
class square extends shape {
void draw(){
System.out.println("This is square!");
}
}
public class Main{
public static void main(String[] args) {
shape s = new shape();
s.draw();
}
}
```

Output :

```
This is circle!
This is rectangle!
This is square!
```

Task#08:Create an Interface Shape with three child classes Circle, Square and Rectangle. Create a method Draw in the interface implemented in each class. E.g. in Circle "This is a Circle". Create an array variable of type Shape of 10 elements in main class. Design a method in class main that can fills the Shape array with the child classes Objects randomly. Create another method in main class that can call the Draw method of each object filled in the array.

Source code :

```
package main;
interface Shape {
String draw();
}
class circle implements Shape {
public  String draw(){
return ("this is circle");
}
}
class square implements Shape{
public String draw(){
return ("this is square");
}
}
class rectangle implements Shape {
```

```
public String draw(){
return ("this is rectangle");
}
}
public class Main{
Shape arr[] = new Shape[10];
void fillArray(){
circle c = new circle();
square s = new square();
rectangle r = new rectangle();
for(int i=0;i<arr.length;i++){
int rn = (int)(2*Math.random());
if(rn==0){
arr[i]=c;
}
else if(rn==1){
arr[i]=s;
}
else if(rn==2){
arr[i]=r;
}
}
}
void display(){
for(int i=0;i<arr.length;i++){
System.out.println(arr[i].draw());
}
}
public static void main(String[] args) {
Main ma = new Main();
ma.fillArray();
ma.display();
}
}
```

Output :

```
this is square
this is square
this is square
this is circle
this is square
this is circle
this is square
this is circle
this is circle
this is circle
```

Task#09: Develop a registration system for a University. It should consist of three classes namely Student, Teacher, and Course. For our example, a student needs to have a name, roll number, gender, and GPA to be eligible for registration. Therefore choose appropriate data types for encapsulating these properties in a Student object/s. Similarly a teacher needs to have name, address, telephone number, and a degree (or a list of degrees) he has received. Finally courses must have a name, a student (or a list of students) registered for the course, and a teacher assigned to conduct the course. This implies that an object of the Course class should be composed of one or more Students and a Teacher, along with any other type of objects or primitives you think appropriate. Write a Test class to create three

different courses, each having students and a teacher assigned to it. Store all three objects in an array of type Course and pick a reference randomly from it. A call to a method, say printDetails(), of the selected course reference should print name of the course, details of the teacher assigned to that course, and names and roll numbers of the students enrolled with the course. Also, there should be some arrangement in the course class so that the number of courses in existence at a given point can be known. That is, there should be a method in the Course class that keeps track of the number of objects created and return the count as int.

Source code :

```
package main;
class student{
student(){
}
student(String s , String add , int rn , double gp){
setname(s);
setaddress(add);
setrollno(rn);
setgpa(gp);
}
void printing(){
System.out.println("name : " + getname());
System.out.println("address : " + getaddress());
System.out.println("rollno : " + getrollno());
System.out.println("gpa : "+ getgpa());
}
private String name;
private String address;
private int rollno;
private double gpa;
void setname(String s){
name=s;
}
String getname(){
return name;
}
void setaddress(String s){
address=s;
}
String getaddress(){
return address;
}
void setrollno(int s){
rollno=s;
}
int getrollno(){
return rollno;
}
void setgpa(double s){
gpa=s;
}
double getgpa(){
return gpa;
}
String tostring(){
```

```
return ("name : "+getname()+" " + " roll no : " + (getrollno()) + " address : " + (getaddress()) + " gpa :" +
getgpa());
}
}
class teacher  {
teacher(){
}
String name;
String address;
long tp;
String degree;
teacher(String n,String ad,long t,String d){
name=n;
address=ad;
tp=t;
degree=d;
}
void printing(){
System.out.println("teacher name : " + name);
System.out.println("address : " + address);
System.out.println("telephone no : " + tp);
System.out.println("degree : " + degree);
}}
class course extends Main {
int count;
student s[];
teacher t;
course(String a,student s[],teacher t){
course = a;
this.s=s;
this.t=t;
}
String course;
void printing(){
System.out.println("course name : " + course);
t.printing();
System.out.println("students registered in this course are :");
for(int i = 0;i<s.length;i++){
System.out.println(s[i].tostring());
}
}
course a[] = new course[3];
int created(){
int i=0;
for(;i<a.length;i++){
}
System.out.print("no of courses available in existence are : ");
return i;
}
}
public class Main{
public static void main(String[] args) {
student c1[] = new student[3];
```

```
teacher t1 = new teacher("Ahmad","C-384",3434,"CE");
c1[0] = new student("aoun","b-103",104,3.2);
c1[1] = new student("ali","b-104",104,3.6);
c1[2] = new student("faizan","b-105",143,3.8);
teacher t = new teacher("Kaleemullah","b-20",23,"bscs");
student c2[] = new student[3];
teacher t2 = new teacher("Sir din","c-20",045665,"bscs");
c2[0] = new student("Kakul","c-103",104,2.6);
c2[1] = new student("Arshad","c-102",104,2.8);
c2[2] = new student("Zemmam","c-106",104,2.9);
student c3[] = new student[3];
teacher t3 = new teacher("Huzaifa","f-20",372834,"bscs");
c3[0] = new student("husnain","f-114",149,3.13);
c3[1] = new student("Ahmed kapadia","f-176",104,3.83);
c3[2] = new student("Akhil","f-103",104,3.93);
course c = new course("Phyiscs" ,c1,t1);
course a[] = new course[3];
a[0] = c = new course("C#" , c1,t1);
a[1]=c=new course ("Java",c2,t2);
a[2]=c=new course("CF" ,c3,t3);
a[(int)(0+(2-0)*Math.random())].printing();
System.out.println(c.created());
}
}
```

Output :

```
course name : Java
teacher name : Sir din
address : c-20
telephone no : 19381
degree : bscs
students registered in this course are :
name : Kakul  roll no : 104 address : c-103 gpa :2.6
name : Arshad  roll no : 104 address : c-102 gpa :2.8
name : Zemmam  roll no : 104 address : c-106 gpa :2.9
no of courses available in existence are : 3
```

Task#10: Write a program that can show the precedence level of Constructor, Static Block and Non Static Block. Use atleast three classes in a hierarchy in this program.

Source code:

```
package main;
class a extends b {
a(){
System.out.println("constructor");
}
{
System.out.println("non static block");
}
}
class b  {
static{
System.out.println("static Block");
}
}
public class Main{
public static void main(String[] args) {
a obj= new a();
}
}
```

Output :

```
static Block
non static block
constructor
```

Task#11:Create a String array. Fill it and Display the contents of array in Ascending Order.

Source code :

```
package main;
public class Main {
public static void main(String[] args) {
String arr1[] = {"Haider", "Aoun", "Ali", "Ahmad"};
String temp;
System.out.println("Orriginal Array!");
for (int i = 0; i < arr1.length; i++) {
System.out.print(arr1[i] + " ");
}
System.out.println();
for (int i = 0; i < arr1.length; i++) {
int flag = 0;
for (int j = 0; j < arr1.length - 1 - i; j++) {
if (arr1[j].compareTo(arr1[j + 1]) > 0) {
temp = arr1[j];
arr1[j] = arr1[j + 1];
arr1[j + 1] = temp;
flag = 1;
}
}
if (flag == 0) {
break;
}
}
System.out.println("Sorted Array!");
for (int i = 0; i < arr1.length; i++) {
System.out.print(arr1[i] + " ");
}
}
}
```

Output :

```
Orriginal Array!

Haider Aoun Ali Ahmad

Sorted Array!

Ahmad Ali Aoun Haider
```

Task#12:Create an int aray of size 10 and initialize it with Random even numbers from 22 to 40. Display the contents of array

Source code :

```
package main;
public class Main {
public static void main(String[] args) {
int arr []=new int[10];
int even;
for(int i=0;i<arr.length;i++){
even=(int)(((Math.random()*19)+21));
if(even%2==0){
arr[i]=even;
}
else if(even%2!=0){
even+=1;
arr[i]=even;
}
}
for(int j=0;j<10;j++){
System.out.print(arr[j]+" ");
}
}
}
```

Output:

```
38 22 40 24 30 28 26 36 38 28
```

Task#13: Declare and initialize an array of type char and Randomly initialize it with alphabets from „a‟ to 'z'. Print character with its integral equivalent.

Source code :

```
package main;
public class Main {
public static void main(String[] args) {
int ar[] = new int[86];
char arr[] = new char[26];
for(int i =0;i<ar.length;i++){
ar[i]=(int)(97+(122-97)*Math.random());
}
int p =0;
for(int i =0;i<ar.length;i++){
if(ar[i]!=ar[i+1]){
arr[p]=(char)ar[i];
p++;
if(p==26){
break;
}
}
}
for(int i=0;i<arr.length;i++){
System.out.println(arr[i] + "=" + (int)arr[i] +" ,"  );
}
}
}
```

Output:

```
c=99 ,k=107 ,l=108 ,r=114 ,q=113 ,j=106 ,x=120 ,v=118 ,i=105 ,y=121 ,q=113 ,h=104 ,m=109 ,n=110
,o=111 ,g=103 ,d=100 ,g=103 ,m=109 ,d=100 ,j=106 ,h=104 ,y=121 ,a=97 ,b=98 ,n=110
```

Task#14:Create an array of String variables and initialize it with the names of the months from January to December. Create an array containing 12 random decimal values between 0.0 and 100.0. Display the names of each month along with the corresponding value. Calculate and display the average of the 12 decimal values.

Source code:

```
package main;
public class Main {
public static void main(String[] args) {
double correspond=0;
String month[]={"jan","feb","march","april","may","june","july","august","sept","oct","novem","decem"};
double arr[] = new double[12];
for(int i = 0;i<arr.length;i++){
arr[i]=100*Math.random();
correspond=correspond+arr[i];
}
for(int j=0;j<month.length;j++){
System.out.println(month[j] + "=" + arr[j]);
}
System.out.println("average of 12 decimal values is : " + correspond);
}
}
```

Output :

```
jan=78.4829544404039
feb=74.56906080136854
march=22.851644553999495
april=19.469975197758004
may=28.342017469895787
june=39.53248637179072
july=15.896424775455786
august=91.01638062781107
sept=26.882657982005654
oct=98.66540686787671
novem=0.15081587875560576
decem=83.1780385702204
average of 12 decimal values is : 579.0378635373417
```

Task#15:Generate 100 Random characters and checks every character for a vowel. If it is a vowel print it.

Source code :

```
package main;
public class Main {
public static void main(String[] args) {
for(int i =0;i<100;i++){
int s = (int)(65+(122-65)*Math.random());
char ch = (char)s;
if(ch=='a' || ch == 'e' || ch =='i' || ch=='o'|| ch=='u' || ch=='A' || ch == 'E' || ch =='I' || ch=='0'|| ch=='U'){
System.out.println(ch + " is a vowel");
}
}
}
}
```

Output :

```
a is a vowel
i is a vowel
i is a vowel
u is a vowel
u is a vowel
o is a vowel
i is a vowel
e is a vowel
o is a vowel
A is a vowel
u is a vowel
o is a vowel
I is a vowel
U is a vowel
U is a vowel
```

Task#16:Write a Program that can fill and array of int with prime numbers less then 100.

Source code:

```
package main;
public class Main {
public static void main(String[] args) {
int a =2,s=0,temp=0;
int prime[]= new int[26];
for(int i =1;i<100;i++){
for(int j =2;j<i-1;j++){
if(i%j==0){
temp=temp+1;
}
}
if(temp==0){
prime[s]=i;
s++;
}
else{
temp=0;
}
}
for(int i=0;i<prime.length;i++){
System.out.print(prime[i]+" ");
}
}
}
```

Output:

```
1 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
```

Task#17:Create a String variable with a Paragraph. Create a String Array equal to the numbers of word in the Paragraph. Fill the Array with the words and sort them in Ascending Order. The program must be dynamic i.e. change in the Paragraph will not effect the Program.

Source code :

```
package main;
import java.util.*;
public class Main{
public static void main(String[] args) {
String s = "Life is like a box of chocolates you never know what you're gonna get.";
int count=1;
for(int i =0;i<s.length()-1;i++){
if((s.charAt(i)==' ') && (s.charAt(i+1)!=' ')){
count+=1;
}
}
int k =0;
System.out.println("There are " + count + " words in paragraph");
String arr[] = s.split(" ");
Arrays.sort(arr);
for(int j = 0 ; j<arr.length;j++){
System.out.print(arr[j] + " ,");
}
}
}
```

Output :

```
There are 14 words in paragraph

Life ,a ,box ,chocolates ,get. ,gonna ,is ,know ,like ,never ,of ,what ,you ,you're
```

Task#18: Write a program that extracts username and the domain information from an E-mail address. For example, if the email address is "user@mydomain.com", your program will print User name = user Domain = mydomain Extension = com

Source code :

```
package main;
import java.util.*;
public class Main {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Write your e mail:");
String s = sc.next();
System.out.println("username : " + username(s));
System.out.println("domain : " + domain(s));
System.out.println("extension : " + extension(s));
}
static String username(String id){
int i =0;
int h=id.indexOf('@');
return id.substring(i,h);
}
static String domain(String id){
int i = id.indexOf('@')+1;
int h=id.indexOf('.')+1;
return id.substring(i,h);
}
static  String extension(String id){
int i = id.lastIndexOf('.')+1;
int h=id.length();
return id.substring(i,h);
}
}
```

Output :

```
Write your e mail:
ahmedbaseer871@gmail.com
username : ahmedbaseer871
domain : gmail.
extension : com
```

Task#19:A Company wants to develop a Company Information System that can show all the information related to the Company and its Employees. The system can also calculate the Employee weekly and monthly salary. The company has two types of Employees, Hourly and Monthly. Monthly Employee are Workers and Managers. Hourly Employees are only Workers. Working hours are 2 to 8 daily for Hourly Employee at a rate of 20rs per Hour. Monthly pay is 3000rs for Workers and 5000rs for Managers. Design a System that can do the following tasks. · Show all the information of Company i.e. Name, Address, Phone Numbers and No of Employees. · Show all the information of Employee i.e. Employee ID, Type of Employee, Employee Designation, Name, Address, Phone number and Salary. Calculate the Salary of Hourly Employee using No of Working Hours. · Calculate the amount paid to All Hourly Employees. · Calculate the amount paid to All Monthly Employees. · Calculate the entire amount paid by the Company to All Employees. Use Composition and Inheritance where necessary. Create common Attributes and Methods in Parent class. Do not allow the user to create the object of Parent class. Create a Test class. Create 6 Employees objects in the Test class; 2 Hourly, 2 Managers and 2 Monthly Workers. Use Arrays to store the objects. Call methods from the Test class that verifies the above define tasks.

Source code :

```
package main;
class parent {
String name;
int id;
String address;
int pn;
int hour_salary = 20;
int month_salary = 3000;
int manager_salary = 5000;
int total = 2*(6*7*hour_salary) + 2*month_salary + 2*manager_salary;
void company(){
System.out.println("Name : BANK Al HABIB");
System.out.println("Address : Main Road Sanghar");
System.out.println("Phone No : 2324323242");
System.out.println("No of employees 6");
System.out.println("total amount paid to hourly workers : " + 2*(6*7*hour_salary)+"rs");
System.out.println("total amount paid to monthly workers : " + 2*month_salary+"rs");
System.out.println("total amount paid to all the employees : " + total );
}
void display(){
System.out.println("name : " + name);
System.out.println("id : "+  id);
System.out.println("address : " + address);
System.out.println("weekly salary : " + (6*7*hour_salary)+"rs");
System.out.println("phone no :  " + pn);
}
void display1(){
```

```
System.out.println("name : " + name);
System.out.println("id : "+  id);
System.out.println("address : " + address);
System.out.println("monthly salary  : " + manager_salary+"rs");
System.out.println("phone no :  " + pn);
}
void display2(){
System.out.println("name : " + name);
System.out.println("id : "+  id);
System.out.println("address : " + address);
System.out.println("monthly salary  : " + month_salary+"rs");
System.out.println("phone no :  " + pn);
}
}
class employee extends parent {
employee(){
}
String name;
int id;
employee(String s , int i,String address,int phonen){
super.address=address;
super.name =s;
super.id=i;
super.pn=phonen;
}
void company(){
super.company();
}
void display(){
super.display();
}
void display1(){
super.display1();
}
void display2(){
super.display2();
}
}
public class Main {
public static void main(String[] args) {
employee arr[] = new employee[6];
employee e = new employee();
arr[0] = new employee("Ahmad",124,"C-384",3424);
arr[1]  = new employee("Aoun",104,"Buns Road",4873);
arr[2] = new employee("Kaleemullah",135,"d123",345453);
arr[3] = new employee("Arshad",229,"f234",9558);
arr[4] = new employee("Ali",132,"f909",3323);
arr[5] = new employee("Mstafa",110,"d9309",3278);
e.company();
for(int i =0;i<arr.length;i++){
if(i==0){
System.out.println("Hourly Workers");
}
if(i<2){
System.out.print(i+1+")");
System.out.println();
arr[i].display();
System.out.println();
}
if(i==2){
System.out.println("Managers");
}
if(i>1 && i<4){
System.out.println(i-1+")");
arr[i].display1();
System.out.println();
}
if(i==3){
System.out.println("Monthly Workers");
}
if(i>3 && i<6){
System.out.println(i-3+")");
arr[i].display2();
System.out.println();
}
}
}
}
```

Output :

```
Name : BANK Al HABIB
Address : Main Road Sanghar
Phone No : 2324323242
No of employees 6
total amount paid to hourly workers : 1680rs
total amount paid to monthly workers : 6000rs
total amount paid to all the employees : 17680
Hourly Workers
1)
name : Ahmad
id : 124
address : C-384
weekly salary : 840rs
phone no :  3424
2)
name : Aoun
id : 104
address : Buns Road
weekly salary : 840rs
phone no :  4873
Managers
1)
name : Kaleemullah
id : 135
address : d123
monthly salary  : 5000rs
phone no :  345453
```

```
2)
name : Arshad
id : 229
address : f234
monthly salary  : 5000rs
phone no :  9558
Monthly Workers
1)
name : Ali
id : 132
address : f909
monthly salary  : 3000rs
phone no :  3323
2)
name : Mstafa
id : 110
address : d9309
monthly salary  : 3000rs
phone no :  3278
```

Task#20:Create a class with a name Calculate. Add two basic arithmetic functions to it, such as add() and subtract() to perform mathematical calculations. Now overload these methods so that they can take three types of values, an int, a double or a char. Note that if characters are passed to a method, it should return char, if double is sent to a method it should give its answer in double, and so on. Use this class from a test class stored in a separate file.

Source code :

```
package main;
class calculate {
void add(int a,int b){
System.out.println("addition of " + a + " and " + b + " is " + a+b);
}
void subtract(int a,int b){
System.out.println("subtraction of " + a + " and " + b + " is " + (a-b));
}
char add(char c , char d){
System.out.println("char 1 " + c + " " + " char 2 " + d);
int  a = c+d;
char ch = (char)a;
System.out.print("sum of these characters are : " + " ");
return ch;
}
char subtract(char c , char d){
System.out.println("char 1 " + c + " " + " char 2 " + d);
int  a = c-d;
char ch = (char)a;
System.out.print("difference of these characters are : " + " ");
return ch;
}
Double add(double d , double s){
System.out.println("addition of " + d + " and " + s + " is " + " ");
return d+s;
}
Double subtract(double d , double s){
```

```
System.out.print("subtraction of " + d + " and " + s + " is " + " ");
return d-s;
}
}
public class Main {
public static void main(String[] args) {
char a,b;
calculate c = new calculate();
System.out.println(c.add('d','f'));
System.out.println();
System.out.println(c.add(7.3, 3.6));
System.out.println();
c.add(2, 5);
System.out.println();
System.out.println(c.subtract('a','h'));
System.out.println();
System.out.println(c.subtract(9.3, 2.6));
System.out.println();
c.subtract(10, 7);
}
}
```

Output :

```
char 1 d  char 2 f
sum of these characters are :  Ê

addition of 4.5 and 2.2 is
6.7

addition of 23 and 45 is 2345

char 1 a  char 2 h
difference of these characters are :  □

subtraction of 1.2 and 9.8 is  -8.600000000000001

subtraction of 21 and 34 is -13
```

Task#21:Create a class named Distance that can store measurement of distances in two possible units, kilometers and meters. The distances in kilometers are to be stored as double values whereas any distance in meters is to be stored as int. Override the equals() method of the Object class so that any two objects of Distance class can be compared. Also, override the toString() of the Object class to display the content of Distance object with its proper units.

Source code:

```
package main;
import java.util.Scanner;
class Distance {
double km;
int m;
Distance(double cm){
km=cm/100000;
m=(int)(cm/100);
}
@Override
public boolean equals(Object obj){
Distance ob=(Distance)obj;
if(this.km!=ob.km){
if(this.m!=ob.m){
return false;}}
else{
return true;
}
return true;
}
@Override
public String toString(){
return "Distance in km :"+km+"\nDistance in meter :"+m;
}
}
public class Main {
public static void main(String[] args) {
Scanner sc=new Scanner(System.in);
System.out.println("Enter the distance travelled in cm");
double cm=sc.nextInt();
System.out.println("Enter the Second distance travelled in cm");
double cm1=sc.nextInt();
Distance obj1=new Distance(cm);
Distance obj2=new Distance(cm1);
if(obj1.equals(obj2)){
System.out.println("They are equal");
}
else{
System.out.println("They are not equal");
}
System.out.println(obj1.toString());
System.out.println(obj2.toString());
}
}
```

Output:

```
Enter the distance travelled in cm
10000
Enter the Second distance travelled in cm
1000
They are not equal
Distance in km :0.1
Distance in meter :100
Distance in km :0.01
Distance in meter :10
```

Task#22: Create an inheritance hierarchy of Vehicle: Car, Bus and Truck, with the SchoolBus further extending the hierarchy through the Bus. Each of the derived class objects should be packaging information about category and passenger carrying capacity of that vehicle, the category being the variable of the abstract base class. Provide a method that is common to all derived classes, such as getPassengerCapacity (), with class Truck having an additional method called getLoadingCapacity () that should be returning that truck's loading capacity in tons.

Create objects of all the derived classes in a test class and store the references in an array of type Vehicle, the top-level class of this hierarchy. Provide logic so that at every run of the program any one

of the four references stored in the array is randomly selected and the methods of that object are called polymorphically. Every random access must result in display of category and passenger carrying capacity of that particular vehicle, except in the case of a Truck object, where the call must result in the display of category, passenger carrying capacity and the cargo loading capacity.

Source code:

```
package main;
class vehicle{
void category()
{System.out.println("This Is all Vehicle");}
void getpassengercapacity()
{System.out.println("Total Capacity is null");}
void getloadingcapacity()
{System.out.println("5 tons");}
void vehicle()
{System.out.println("All Vehicles");
}}
class car extends vehicle{
void category()
{System.out.println("This Is Car");}
void getpassengercapacity()
{System.out.println("Total Capacity is 4");}
}
class truck extends vehicle{
void category()
{System.out.println("This Is truck");}
void getpassengercapacity()
{System.out.println("Total Capacity is 2");}
void getloadingcapacity()
{System.out.println("5 tons");}
}
class bus extends vehicle{
void category()
{System.out.println("This Is Bus");}
void getpassengercapacity()
{System.out.println("Total Capacity is 30");}
}
class schoolbus extends bus{
void category()
{System.out.println("This Is Schoolbus Extending Bus");}
void getpassengercapacity()
{System.out.println("Total Capacity is 32");}
}
public class Main {
public static void main(String[] args) {
vehicle v[]=new vehicle[4];
v[0]=new bus();
v[1]=new car();
v[2]=new schoolbus();
v[3]=new truck();
for(int a=0;a<v.length;a++)
{
v[a].category();
v[a].getpassengercapacity();
if(a==3)
{v[a].getloadingcapacity();}
}
}
}
```

Output:

```
This Is Bus
Total Capacity is 30
This Is Car
Total Capacity is 4
This Is Schoolbus Extending Bus
Total Capacity is 32
This Is truck
Total Capacity is 2
5 tons
```

Task#23: Create a String variable with a Paragraph. Check the number of characters in the String and count the occurrence of each character.

Source code :

```
package main;
public class Main{
public static void main(String[] args) {
String para = "Life is like riding a bicycle. To keep your balance, you must keep moving";
int count =0;
for(int i = 0;i<para.length();i++){
if(para.charAt(i)!=' '){
count=count+1;
}
}
System.out.println(" there are " + count + " words in paragraph");
int count1 = para.length();
int check = para.replace("i","").length();
int occurence = count1-check;
System.out.println("no of occurences of i is " + occurence);
}
}
```

Output :

```
There are 60 words in paragraph
no of occurences of i is 7
```

Task#24: Write a Program that can sort an Array of 10 Randomly Generated Integers by using Bubble sort.

Source code :

```
package main;
public class Main{
public static void main(String[] args) {
int arr []=new int[10];
for(int i=0;i<10;i++){
arr[i]=(int)(2+(20-2)*Math.random());
}
System.out.println("Original Array!");
for(int i=0;i<10;i++){
System.out.print(arr[i]+" ");
}
System.out.println();
System.out.println("Sorted Array!");
int temp;
for(int i =0;i<arr.length;i++){
for(int j = 0;j<arr.length-1;j++){
if(arr[j]>arr[j+1]){
temp = arr[j];
arr[j]=arr[j+1];
arr[j+1]=temp;
}
}
}
for(int i =0;i<arr.length;i++){
System.out.print(arr[i]+" ");
}
}
}
```

Output :

```
Original Array!
12 16 14 7 12 5 11 18 4 4
Sorted Array!
4 4 5 7 11 12 12 14 16 18
```

Task#25: Find Minimum and Maximum Values in a Randomly Generated Integer Array of 10 Elements with sorting it.

Source code :

```
package main;
public class Main {
public static void main(String[] args) {
int arr []=new int[10];
System.out.println("Generated Array!");
for(int i=0;i<10;i++){
arr[i]=(int)(2+(20-2)*Math.random());
}
for(int i=0;i<10;i++){
System.out.print(arr[i]+" ");
}
System.out.println();
int maximum = arr[0];
int minimum = arr[0];
for(int i =0;i<arr.length;i++){
if(arr[i]>maximum)
maximum=arr[i];
if(arr[i]<minimum){
minimum = arr[i];
}
}
System.out.println("Maximum no of array is " + maximum);
System.out.println("minimum no of array is " + minimum);
}
}
```

Output :

```
Generated Array!
2 5 12 15 3 18 11 8 12 8
Maximum no of array is 18
```

minimum no of array is 2

Task#26:Write a program that can Store and Remove Data of 10 Integers in an Array in Stack pattern. (Stack functionality is "Last in First Out" LIFO).

Source code:

```
package main;
class Stack{
private int array[];
private int top;
private int capacity;
Stack (int capacity){
this.array=new int[capacity];
this.capacity= capacity;
this.top=-1;
}
public void push(int item){
if(isfull()){
throw new RuntimeException("Stack is full:");
}
array[++top]=item;
}
public int pop(){
if(isempty()){
throw new RuntimeException("Stack is Empty:");
}
return array[top--];
}
public int peek(){
if(isempty()){
throw new RuntimeException("Stack is Empty:");
}
return array[top];
}
public boolean isfull(){
return top==capacity-1;
}
public boolean isempty(){
return top==-1;
}
}
public class Main {
public static void main(String[] args) {
Stack obj=new Stack(10);
for (int i=0;i<10;i++){
obj.push((int)(Math.random()*100)+1);
}
System.out.println("POP:");
for (int i=0;i<9;i++){
System.out.println(obj.pop());
}
System.out.println("This element is on peek:"+obj.peek());
}
}
```

Output:

```
POP:
82
81
22
56
1
40
92
81
33
This element is on peek:33
```

Task#27:Write a program that can Store and Remove Data of 10 Integers in an Array in Queue pattern. (Queue functionality is “First in First Out” FIFO).

Source code:

```
package main;
class Queue{
int queue[]=new int[10];
int size;
int front;
int rear;
public void enQueue(int data){
queue[rear]=data;
rear++;
size++;
}
public int deQueue(){
int data=queue[front];
front++;
size--;
return data;
}
public void show(){
System.out.println("Elements:");
for(int i=0;i<10;i++){
try{
System.out.print(queue[front+i]+" ,");
}
catch(ArrayIndexOutOfBoundsException e){
System.out.println();
System.out.println("Array size reduces to 5 that's why cant run more:");
break;
}
}
}
}
public class Main {
public static void main(String[] args) {
Queue q=new Queue();
for(int i=0;i<10;i++){
q.enQueue((int)(Math.random()*100)+1);
}
q.show();
for(int i=0;i<5;i++){
q.deQueue();
}
System.out.println();
System.out.println("Remaing Elements after removing 5 elements:");
q.show();
}
}
```

Output:

```
Elements:
67 ,61 ,42 ,54 ,17 ,28 ,9 ,48 ,69 ,90 ,
Remaing Elements after removing 5 elements:
Elements:
28 ,9 ,48 ,69 ,90 ,
Array size reduces to 5 that's why cant run more:
```