Name	**Hatim Yusuf Sawai**
UID no.	**2021300108**
Experiment No.	**7**

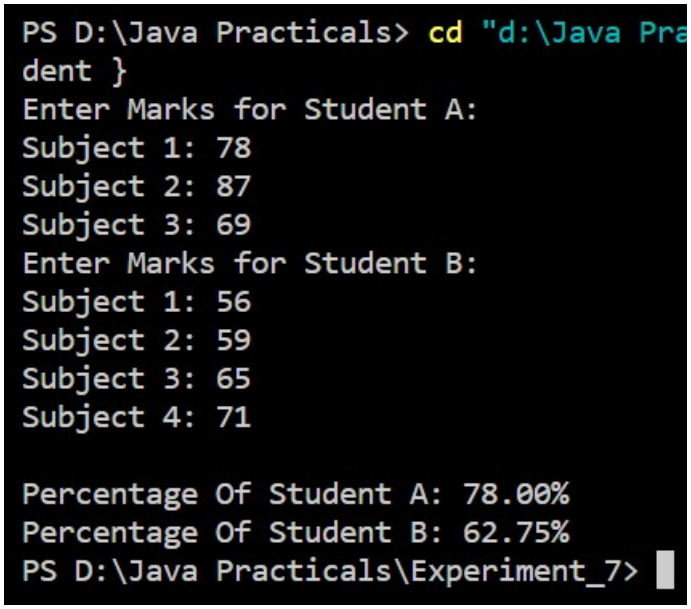
AIM:	**Programs on Abstraction: Abstract Classes & methods with multilevel inheritence**
Program 1	
PROBLEM STATEMENT:	Write a program to calculate the percentage of marks obtained in three subjects (each out of 100) by student A and in four subjects (each out of 100) by student B. Create an abstract class 'Marks' with an abstract method 'getPercentage'. It is inherited by two other classes 'A' and 'B' each having a method with the same name which returns the percentage of the students. The constructor of student A takes the marks in three subjects as its parameters and the marks in four subjects as its parameters for student B.

PROGRAM:	import *java.util.*;*
	abstract class Marks {
	int s1,s2,s3,s4;
	double perc;
	abstract void getPercentage();
	}
	class A extends Marks {
	A(int s1, int s2,int s3) {
	this.*s1* = s1;
	this.*s2* = s2;
	this.*s3* = s3;
	}
	void getPercentage() {
	perc = s1+s2+s3;

```
        perc = (perc/3);
        System.out.printf("\nPercentage Of Student A: %.2f%%",perc);
    }
}
class B extends Marks {
    B(int s1, int s2,int s3,int s4) {
        this.s1 = s1;
        this.s2 = s2;
        this.s3 = s3;
        this.s4 = s4;
    }
    void getPercentage() {
        perc = s1+s2+s3+s4;
        perc = (perc/4);
        System.out.printf("\nPercentage Of Student B: %.2f%%", perc);
    }
}
public class Student {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int[] marks = new int[4];
        System.out.println("Enter Marks for Student A:");
        for(int i=0;i<3;i++) {
            System.out.printf("Subject %d: ",i+1);
            marks[i] = sc.nextInt();
        }
        A g1 = new A(marks[0], marks[1], marks[2]);
        System.out.println("Enter Marks for Student B:");
        for (int i=0;i<4;i++) {
            System.out.printf("Subject %d: ", i + 1);
            marks[i] = sc.nextInt();
        }
        B g2 = new B(marks[0], marks[1], marks[2], marks[3]);
        g1.getPercentage();;
        g2.getPercentage();
        sc.close();
```

```
    }
}
```

RESULT:



Program 2

PROBLEM STATEMENT:

Define a class Westside that have sales in cloths and accessories. Let us say 10 cloths and 10 accessories each cost is 5000. Cloths and accessories are limited and updated as soon as purchase is done.
Define a class named Payment(abstract class) that contains an instance variable of type double that stores the amount of the payment. Amount is initialized 25,000 and updated with each purchase. Also create a method named (abstract) paymentDetails that updates the amount of the payment.
Next, define a class named CashPayment that is derived from Payment. This class should redefine the paymentDetails method to indicate that the payment is in cash. Include appropriate constructor(s)/methods.
Define a class named CreditCardPayment that is derived from Payment. This class should contain instance variables for the name on the card, expiration date, and credit card number. Include appropriate constructor(s)/methods. Finally, redefine the paymentDetails method to include all credit card information in the printout.

	Define a class Person that contains person_name and P_id. Create a main method that creates at least five persons who will be given random chance for buying using any way of payment method CashPayment /CreditCardPayment. Once person buys cloths/ accessories amount get debited.

PROGRAM:	import *java.util.**;
	class Westside {
	int clothes = 10;
	int accessories = 10;
	void reduce_clothes(int n) {
	clothes = clothes - n;
	}
	void reduce_accessories(int m) {
	accessories = accessories - m;
	}
	}
	abstract class Payment {
	abstract void payment_details(int a, int b);
	}
	class cashpayment *extends* Payment {
	void payment_details(int a, int b) {
	System.*out*.println("Amount " + a + " Paid in cash, balance : " + b);
	}
	}
	class CreditCardPayment *extends* Payment {
	String cardname;
	int cardnumber;
	int carddate, cardmonth, cardyear;
	int e;
	static int balance;
	CreditCardPayment() {
	balance = 25000;
	}
	void get_details() {
	Scanner sc = new Scanner(System.*in*);

```
        System.out.println("Enter Card name:");
        cardname = sc.next();
        System.out.println("Enter card number:");
        cardnumber = sc.nextInt();
        System.out.println("Enter expiry date in dd/mm/yyyy format:");
        carddate = sc.nextInt();
        cardmonth = sc.nextInt();
        cardyear = sc.nextInt();
        sc.close();
    }
    int reduce_card(int n) {
        balance = balance - n;
        return balance;
    }
    void payment_details(int a, int r) {
        System.out.println("Amount $" + a + " paid by credit card , Card
Name: " + this.cardname
                + " Credit Card number: " + this.cardnumber + ", balance
remaining :" + r + " date of expiry :"
                + this.carddate + "/" + this.cardmonth + "/" + cardyear);
    }
    int card_balance() {
        return balance;
    }
}
public class Person {
    String name;
    int id;
    void setPerson() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter your name:");
        name = sc.nextLine();
        System.out.println("Enter your id no.:");
        id = sc.nextInt();
        sc.close();
    }
```

```
void getperson_details() {
    System.out.println("Name :" + this.name + " , ID :" + this.id);
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    Westside w = new Westside();
    Person[] p = new Person[5];
    CreditCardPayment[] card = new CreditCardPayment[5];
    cashpayment[] cash = new cashpayment[5];
    int x;
    int[] check = new int[5];
    int r = 0, z;
    int n,m;
    int index;
    while (w.clothes != 0 || w.accessories != 0) {
        index = (int) (Math.random() * 5);
        z = 0;
        for (int i = 0; i < r; i++) {
            if (check[i] == index) {
                z++;
            }
        }
        if (r == 0) {
            check[0] = index;
            r++;
        }
        if (z == 0) {
            check[r] = index;
            if (r != 4) {
                r++;
            }
        }
        System.out.println("clothes remaining =" + w.clothes);
        System.out.println("accessories remaining =" + w.accessories);
        System.out.println("Person : " + (index + 1));
```

```
            if (z == 0) {
                card[index] = new CreditCardPayment();
                cash[index] = new cashpayment();
                p[index] = new Person();
                p[index].setPerson();
            }
            System.out.println("Enter no. of clothes u wish to buy");
            n = sc.nextInt();
            System.out.println("Enter no. of accessories u wish to buy");
            m = sc.nextInt();
            int amount = ((n + m) * (5000));
            if ((card[index].card_balance() >= amount) && (n <= w.clothes
&& m <= w.accessories)) {
                w.reduce_accessories(m);
                w.reduce_clothes(n);
                System.out.println("Total amount to be paid: $" + amount);
                System.out.println("Press 2 to pay in cash or press 1 to pay
using credit card");
                int option = sc.nextInt();
                if (option == 1) {
                    System.out.println("Enter Card Details:");
                    card[index].get_details();
                    x = card[index].reduce_card(amount);
                    p[index].getperson_details();
                    card[index].payment_details(amount, x);
                }
                if (option == 2) {
                    x = card[index].reduce_card(amount);
                    p[index].getperson_details();
                    cash[index].payment_details(amount, x);
                }
            } else
                System.out.println("Transaction failed!!");
        }
    }
}
```

RESULT:

```
clothes remaining =10
accessories remaining =10
Person : 2
Enter your name:
aryan
Enter your id no.:
2
Enter no. of clothes u wish to buy
1
Enter no. of accessories u wish to buy
2
Total amount to be paid: $15000
Press 2 to pay in cash or press 1 to pay using credit card
2
Name :aryan , ID :2
Amount 15000 Paid in cash, balance : 10000
clothes remaining =9
accessories remaining =8
Person : 5
Enter your name:
werner
Enter your id no.:
12
```

```
2
Enter no. of accessories u wish to buy
1
Total amount to be paid: $15000
Press 2 to pay in cash or press 1 to pay using credit card
2
Name :werner , ID :12
Amount 15000 Paid in cash, balance : 10000
clothes remaining =7
accessories remaining =7
Person : 4
Enter your name:
rahul
Enter your id no.:
32
Enter no. of clothes u wish to buy
1
Enter no. of accessories u wish to buy
1
Total amount to be paid: $10000
Press 2 to pay in cash or press 1 to pay using credit card
2
Name :rahul , ID :32
Amount 10000 Paid in cash, balance : 15000
```

```
Enter Card Details:
Enter Card name:
axis
Enter card number:
2343
Enter expiry date in dd/mm/yyyy format:
23 4 2032
Name :aryan , ID :2
Amount $5000 paid by credit card , Card Name: axis Credit Card number: 2343, balance remaining :5000 date of expiry :23/4/2032
```

Program 3

PROBLEM STATEMENT:	There is an abstract class called 'AmusementPark'. It has a methods getCost() - Suppose a group comes together so this will return the

	total cost of the group. There are 2 classes 'Esselworld' and 'Imagica' that inherit 'AmusementPark'. In EsselWorld, those above 21 age have an entry fee of Rs. 1050 and for ages below 21 or above 60 it's Rs. 660. Similarly in Imagica, those above 21 age have an entry fee of Rs. 1500 and for ages below 21 or above 60 it's Rs. 1100. Given:- • Esselworld has a total of 15 games • Imagica has a total of 20 games • Both have some games which are not included in the pass and have some additional cost. (Note: You can assume games are numbered like Game 1 to 15 out of which 7 first 7 games(Game no 1 to 7) are included in the pass and the rest of the games (game no 8 to 15) have an extra charge of Rs. 50) • When the person is going back home, the total number of games that were played and which were not played must be shown. Also, show the count of the games played. • On weekends Pass cost is Rs 300 higher compared to the actual cost. • Also, display the total cost of the person when he is leaving.(extra game cost+ticket cost)

PROGRAM:	import *java.util.*;*
	
	abstract class Amusement {
	  abstract int getCost(int a, int b, int c);
	}
	
	class Esselworld *extends* Amusement {
	  int getday(String s) {
	  if (s.compareTo("sunday") == 0 || s.compareTo("saturday") == 0)
	  return 300;
	  else
	  return 0;
	  }
	
	  int getage(int a) {
	  if (a > 21 && a < 60)
	  return 1050;
	  else
	  return 600;

```
    }

    int getextra(int n) {
        if (n > 7) {
            return (n - 7) * 50;
        } else
            return 0;
    }

    int getCost(int a, int b, int c) {
        return (a + b + c);
    }
}

class Imagica extends Amusement {
    int getday(String s) {
        if (s.compareTo("sunday") == 0 || s.compareTo("saturday") == 0)

            return 300;
        else
            return 0;
    }

    int getage(int a) {
        if (a > 21 && a < 60)
            return 1500;
        else
            return 1100;
    }

    int getextra(int n) {
        if (n > 7) {
            return (n - 7) * 50;
        } else
            return 0;
    }

    int getCost(int a, int b, int c) {
        return (a + b + c);
    }
}

public class Main{
    public static void main(String[] args) {
        Imagica imagica = new Imagica();
        Esselworld essel = new Esselworld();
        Scanner sc = new Scanner(System.in);
        System.out.println("press 1 to book tickets for imagica OR press 2 to book tickets for
Esselworld");
```

```
      int c = sc.nextInt();
      if(c==1) {
         System.out.println("Hi ! Welcome to Imagica!!\n");
         System.out.println("Enter the day of your visit");
         String day = sc.next();
         System.out.println("Enter the no. of tickets to be booked");
         int num = sc.nextInt();
         int tot=0;
         int[][] times = new int[num][15];
         int[] played = new int[num];
         int[] games = new int[num];
         int[] age = new int[num];
         int[][] check = new int[num][15];
         for(int i =0;i<num;i++){
            System.out.println("Enter age of person :"+(i+1));
            age[i]= sc.nextInt();
            sc.nextLine();
            int tell;
            int stop=1;
            while(stop!=0) {
                  System.out.println("Which game do u wish to play ");
                  tell = sc.nextInt();
                     times[i][tell-1] =times[i][tell-1]+1 ;
                     played[i] = played[i] + 1;
                     if(check[i][tell-1]==0) {
                        games[i] = games[i] + 1;
                        check[i][tell-1]=check[i][tell-1]+1;
                     }
                     sc.nextLine();
               System.out.println("To play more press 1 or press 0 to quit");
               stop= sc.nextInt();
            }
         }
         for(int i =0;i<num;i++){
            System.out.println("\nTicket cost of person "+(i+1)+" =
$"+imagica.getCost(imagica.getage(age[i]),imagica.getday(day),0));
            System.out.println("\nTotal available Games played by person "+(i+1)+" = "+games[i]);
            System.out.println("Total available Games not played by person "+(i+1)+" = "+(20-
games[i]));
            System.out.println("\n");
            for(int j =0;j<15;j++){
               System.out.println("Game no. "+(j+1)+" played "+times[i][j]+" times.");
            }

tot=tot+imagica.getCost(imagica.getage(age[i]),imagica.getday(day),imagica.getextra(played[i]));
            System.out.println("Ticket cost of person + cost of extra games played =
$"+imagica.getCost(imagica.getage(age[i]),imagica.getday(day),imagica.getextra(played[i])));
         }
         System.out.println("\nTotal Ticket cost of Imagica for "+num+" people = $"+tot);
```

```
    }
    if(c==2) {
        System.out.println("Hi ! Welcome to Esselworld!!\n");
        System.out.println("Enter the day of your visit:");
        String day = sc.next();
        System.out.println("Enter the no. of tickets to be booked:");
        int num = sc.nextInt();
        int tot=0;
        int[][] times = new int[num][15];
        int[] played = new int[num];
        int[] games = new int[num];
        int[] age = new int[num];
        int[][] check = new int[num][15];
        for(int i =0;i<num;i++){
            System.out.println("Enter age of person :"+(i+1));
            age[i]= sc.nextInt();
            sc.nextLine();
            int tell;
            int stop=1;
            while(stop!=0) {
                System.out.println("Which game do u wish to play ");
                tell = sc.nextInt();
                times[i][tell-1] =times[i][tell-1]+1 ;
                played[i] = played[i] + 1;
                if(check[i][tell-1]==0) {
                    games[i] = games[i] + 1;
                    check[i][tell-1]=check[i][tell-1]+1;
                }
                sc.nextLine();
                System.out.println("To play more press 1 or press 0 to quit");
                stop= sc.nextInt();
            }
        }
        for(int i =0;i<num;i++){
            System.out.println("\nTicket cost of person "+(i+1)+" =
$"+essel.getCost(essel.getage(age[i]),essel.getday(day),0));
            System.out.println("\nTotal available Games played by person "+(i+1)+" = "+games[i]);
            System.out.println("Total available Games not played by person "+(i+1)+" = "+(15-
games[i]));
            System.out.println("\n");
            for(int j =0;j<15;j++){
                System.out.println("Game no. "+(j+1)+" played "+times[i][j]+" times.");
            }
            tot= tot+essel.getCost(essel.getage(age[i]),essel.getday(day),essel.getextra(played[i]));
            System.out.println("Ticket cost of person + cost of extra games played =
$"+essel.getCost(essel.getage(age[i]),essel.getday(day),essel.getextra(played[i])));
        }
        System.out.println("\nTotal Ticket cost of Esselworld for "+num+" people = $"+tot);
    }
```

```
  }
}
```

RESULT:

```
Ticket cost of person 1 = $1400

Total available Games played by person 1 = 8
Total available Games not played by person 1 = 12


Game no. 1 played 1 times.
Game no. 2 played 0 times.
Game no. 3 played 1 times.
Game no. 4 played 0 times.
Game no. 5 played 1 times.
Game no. 6 played 0 times.
Game no. 7 played 1 times.
Game no. 8 played 0 times.
Game no. 9 played 1 times.
Game no. 10 played 0 times.
Game no. 11 played 1 times.
Game no. 12 played 0 times.
Game no. 13 played 1 times.
Game no. 14 played 0 times.
Game no. 15 played 2 times.
Ticket cost of person + cost of extra games played = $1500

Ticket cost of person 2 = $1800
```

CONCLUSION: In this experiment, we learned how to declare an abstract class and abstract methods and use them in multilevel inheritance problems.