Batch: A3 Roll No.: 1911034

Experiment / assignment / tutorial No.06

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of the Staff In-charge with date

TITLE :Case Study (for Class Diagram)

AIM : Draw class Diagram for the chosen Case Study . Clearly show

- Attributes
- Multiplicities between classes
- Aggregations/compositions/Association between classes
- Generalization between classes in the class diagram.

And show the implementation of aggregation, association, composition and generalization between the classes.

Expected OUTCOME of Experiment:

CO3: Implement scenarios using object oriented concepts (Drawing class diagram, relationship between classes, sequence diagram)

Books/ Journals/ Websites referred:

1.Ralph Bravaco , Shai Simoson , "Java Programing From the Group Up" Tata McGraw-Hill.

2.Grady Booch, Object Oriented Analysis and Design .

Pre Lab/ Prior Concepts:

Define Class, Methods, Object.

Understanding of Aggregation, Association, Composition and Generalization between classes

List Of Classes:

In our project , we have included the following List of classes:
In the main.java file :

1. Class Main
2. Class Common
3. Class Student
4. Class Instructor
5. Class Admin
6. Class Attendance

In the QuesandTest.java file

1. Class Questions
2. Class JavaQues
3. Class ScieQues

In the StudyMaterial.java file;

1. Abstract class StudyMaterial
2. Class StudyNotes

We have also created a **user-defined Exception class** , class **TestNotFoundException.**

Identify Attributes for each class:

For the class Common, the attributes are:

1. user_id: int
2. user_name : String
3. age : int
4. id : String
5. pass : String

(note : the class Common is inherited by the class Student , Admin and Instructor)

For the class <u>Student,</u> the attributes are

1. div: char
2. Vector<Attendance> s1,s2;
3. S1_m, s2_m : int

For the class Instructor , the attributes are (apart from the ones already inherited from class Common)

1. sub :String
2. iname : String
3. ipass : String
4. s_id: String
5. p: String
6. id: int
7. i_age: int

For the class Admin , the attributes are :

1. id1: String
2. pass1: String
3. name: String
4. u1: String
5. age : int
6. c: char

For the class Atendance, the attributes are :

1. c : char
2. d: int

For the class Questions , the attributes are :

1. String ques
2. String ans

For the classes JavaQues and ScieQues , the attributes are:

1. i: int

For the class StudyNotes, the attributes are:

1. c : int
2. ch : int

Identify List of Methods in each classes:

Methods in class common:

```
1. public void Login(String us_id,String password)
2. public int CheckLogin(Common e,String s1)
```

Methods in class Student

```
1. public ArrayList<Student> navigate(ArrayList<Student>a1,
   ArrayList<Student>a2,int k,char c)
2. public Student Initialize(int i )
3. public int StudentLogin(ArrayList <Student> ar)
4. public ArrayList<Student>
   CheckAtt(ArrayList<Student>a1,ArrayList<Student>a2,int k,char c)
5. public Vector<Questions> TestQues(int k, Vector <Questions>vec,
   Student obj)
6. public void CheckMarks(int k,ArrayList<Student>a)
```

Methods in Class Instructor

```
1. public Instructor Initialize(String s)
2. public int InstructorLogin(Instructor ob)
3. public ArrayList<Student> navigate(ArrayList<Student>a1,
   ArrayList<Student> a2,int k,char cl)
4. public Vector<Questions> TestQues(int k,
   Vector<Questions>ques,Student obj)
5. public void CheckMarks(int k,ArrayList<Student>a)
6.  public ArrayList<Student>
   CheckAtt(ArrayList<Student>a1,ArrayList<Student>a2,int k,char c)
7. public ArrayList<Student> navigate(ArrayList<Student>a1,
   ArrayList<Student>a2,int k,char c)
```

Methods in Class Admin:

```
1. public int adminLogin()
2. public void navigate1(ArrayList <Student> a1, ArrayList
   <Student> a2)
3. public void navigate1(ArrayList <Student> a1, ArrayList
   <Student> a2)
4. public ArrayList <Student> Create(ArrayList <Student> ar, int
   k)
5. public Instructor Create(Instructor obj)
```

Methods in Class JavaQues

1. public static Vector<Questions> JavaQues1()

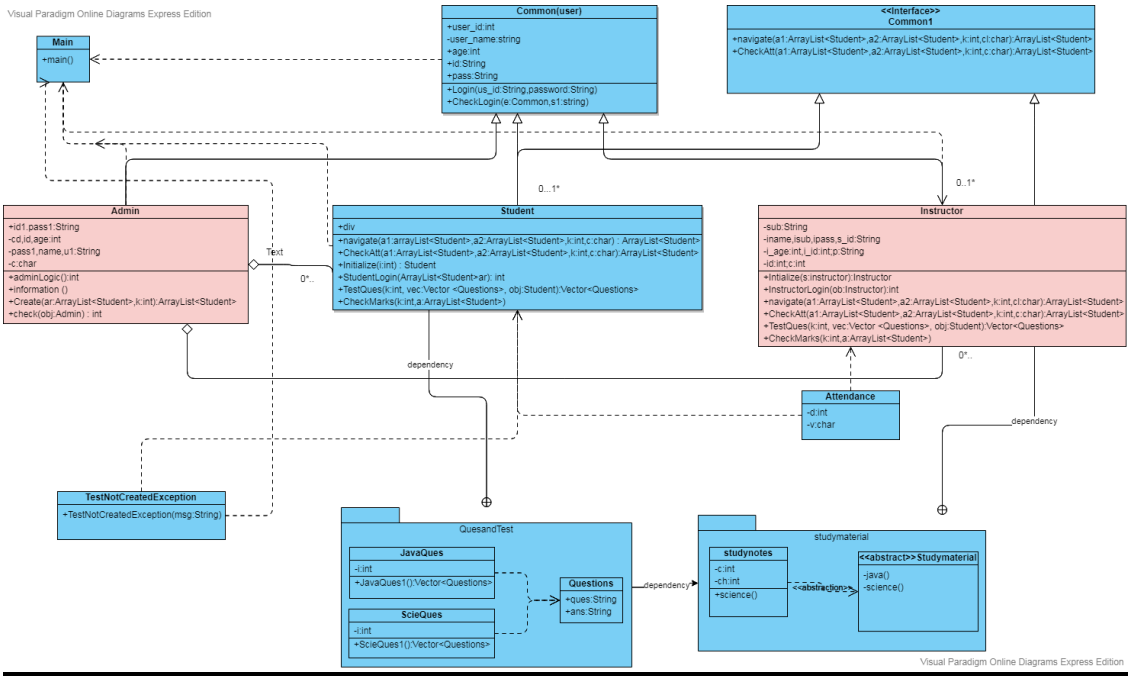
Methods in Class ScieQues

1. public static Vector<Questions>SciQues()

Methods in class StudyMaterial

1. public abstract void science()
2. public abstract void java()

Class Diagram:



Implementation details: (Class Diagram and Code)

Code :

```
import java.util.*;
import QuesandTest.*;
import studymaterial.*;
public class Main
{

public static void main(String args[])
{

System.out.println("Welcome to Course Management System");
int i,ns1,ns2,ns3,ns4,ni,ch,ch1,k,id_n,flag,ich,ich1,tch,s1,k1;

ich1=0;
```

```
char ch2,d,chi;
String p1,p2,sub;
Scanner sc = new Scanner(System.in);

ArrayList <Student> a1 = new ArrayList <Student>();
ArrayList <Student> a2 = new ArrayList <Student>();
Vector<Questions>quesc = new Vector <Questions>();
Vector<Questions>quesj = new Vector <Questions>();


int sch,sch1=0,al=0,au=0;
Admin obj = new Admin();//Admin Class will be used to initially create record of the students , so using Admin object to access the class
do{
  ch = obj.adminLogin();
  if(ch==1)// ch will be used to check , if the authorized admin has access , else it will deny the access to continue.
  break;

}while(ch==2);
System.out.println("CREATING STUDENT RECORDS");
a1= obj.Create(a1,1);

a2 = obj.Create(a2,2);
System.out.println("CREATING INSTRUCTOR RECORDS");
Instructor objs = new Instructor();
Instructor objm = new Instructor();
Instructor objg = new Instructor();
objs = obj.Create(objs);
System.out.println("----------------------");
objm = obj.Create(objm);

do{
System.out.println("Enter any of the following choices");
System.out.println("1 to access a Student Account");
System.out.println("2 to access an Instructor Account");
System.out.println("3 to access Admin Control");
System.out.println("-1 to exit");
ch1 = sc.nextInt();
switch(ch1)
{
  case 1:
  {
System.out.println("Enter the division to which you belong");
```

```
 d = sc.next().charAt(0);

 Student obj2 = new Student();
 switch(d)
 {

   case 'A':
   {
     do{/*this do-while loop is used to prevent the program from ending prematurely if the
password entered is incorrect*/
    sch= obj2.StudentLogin(a1);
    if(sch>=0)
    {sch1 = 2;
    Vector<Questions> que = new Vector <Questions> ();
    System.out.println("Enter 1 to give a test or 2 to choose other options");
s1 = sc.nextInt();
if(s1==1)
{Student ob1 = new Student();
ob1 = a1.get(sch);

 System.out.println("Enter the subject that you want to give a test for, enter 1 for Science or
2 for Java");
ch = sc.nextInt();
if(ch==1)
{
    obj2.TestQues(1,quesc,ob1);

}
else if(ch==2)
{
  obj2.TestQues(2,quesj,ob1);

}


}
    else if(s1==2)
    {
 obj2.navigate(a1,a2,sch,'A');
    }


    }
     else
```

```
     {
       System.out.println("Enter 1 to try again or 2 to continue");
       sch1 = sc.nextInt();

     }}while(sch1!=2);
   }
   break;
   case 'B':
   {

     do{
    sch= obj2.StudentLogin(a2);
    if(sch>=0)
    {
      sch1 = 2;
    Vector<Questions> que = new Vector <Questions> ();
    System.out.println("Enter 1 to give a test or 2 to choose other options");
s1 = sc.nextInt();
if(s1==1)
{Student ob1 = new Student();
ob1 = a2.get(sch);

 System.out.println("Enter the subject that you want to give a test for, enter 1 for Science or
2 for Java");
ch = sc.nextInt();
if(ch==1)
{
    obj2.TestQues(1,quesc,ob1);

}
else if(ch==2)
{
  obj2.TestQues(2,quesj,ob1);

}



}
    else if(s1==2)
    {
 obj2.navigate(a1,a2,sch,'B');
    }
    }
     else
```

```
     {
       System.out.println("Enter 1 to try again or 2 to continue");
       sch1 = sc.nextInt();

     }}while(sch1!=2);
   }

 }
 }
  break;
  case 2:
  {
    Student ob = new Student();
do{
  System.out.println("Please enter your subject");
sub = sc.next();
if(sub.equals(objs.sub)==true)
{
  ich = objs.InstructorLogin(objs);
  if(ich==1)
  {
 System.out.println("Wrong username/password. Press 0 to try again or 2 to return to the main
menu");
  ich1= sc.nextInt();
  }
  else if(ich==0)
  {
    ich1=2;
    System.out.println("enter 1 to check attendance/marks status or enter 2 to create a test");
    tch = sc.nextInt();
    if(tch==1)
    {


    System.out.println("Enter the division for attendance/marks");
    chi = sc.next().charAt(0);
    if(chi=='A')
    {
        objs.navigate(a1,a2,1,'A');
    }
  else if(chi=='B')
  {
      objs.navigate(a1,a2,1,'B');
  }
```

```
    }

    else if(tch==2)
    {
Vector <Questions> ques = new Vector <Questions>();
System.out.println("The list of questions is as follows");

Vector <Questions> ques1 = ScieQues.SciQues1();
quesc = objs.TestQues(1,ques1,ob);/*accepting the questions from the teacher for the test*/
 /*the student will have to attempt the questions chosen by the teacher*/

    }
  }

}

else if (sub.equals(objm.sub)==true)
{
  ich = objm.InstructorLogin(objm);
  if(ich==1)
  {
 System.out.println("Wrong username/password. Press 0 to try again or 2 to return to the main
menu");
  ich1= sc.nextInt();
  }
  else if(ich==0)
  {
    ich1=2;
    System.out.println("enter 1 to check attendance/marks status or enter 2 to create a test");
    tch = sc.nextInt();
    if(tch==1)
    {


    System.out.println("Enter the division for attendance/marks");
    chi = sc.next().charAt(0);
    if(chi=='A')
    {
       objs.navigate(a1,a2,2,'A');
    }
  else if(chi=='B')
  {
      objs.navigate(a1,a2,2,'B');
  }
```

```
    }

    else if(tch==2)
    {
Vector <Questions> ques = new Vector <Questions>();
System.out.println("The list of questions is as follows");

Vector <Questions> ques1 = JavaQues.JavaQues1();
quesj = objm.TestQues(2,ques1,ob);/*accepting the questions from the teacher for the test*/
 /*the student will have to attempt the questions chosen by the teacher*/

    }
  }

}

}while(ich1!=2);
  }
  break;
  case 3:
  {
    Admin obj1 = new Admin();
    do{
    System.out.println("Enter the username");
    p1 = sc.next();
    if(obj1.id.equals(p1)==true)
    {

au = obj1.check(obj1);
if(au==0)
{
System.out.println("Wrong password. Enter 1 to try to again or 2 to continue");


al = sc.nextInt();
}
else if(au==1)
{al=2;
  obj1.navigate1(a1,a2);
}
    }
   else if(obj1.id.equals(p1)==false)
    {
System.out.println("Wrong user-name.Enter 1 to try again or 2 to continue");
```

```
al = sc.nextInt();


    }
    }while(al!=2);

  }
}}while(sch1!=-1);
}
}

 class Common /* this class will provide functions commmon to each of the other three classes so
that they can be directly used after Inheritance.*/

{
public int user_id;
public String user_name;
public int age;
public String id, pass;
public Common(int id, String name, int age)// constructor of Common class used to initialize
some common values for objects belonging to each of the three classes
{
this.user_id = id;
this.user_name= name;
this.age = age;

}

public void Login(String us_id,String password)// this method will be used to give only the
authorized user the access in each of the three classes.
{

this.id = us_id;
this.pass = password;
}
public int CheckLogin(Common e,String s1)/*used to authenticate login of any object*/
{
if(e.pass.equals(s1)==true)
{
  return 1;
}
else
{
  return 0;
```

```
}
}

}
class Attendance{
  int d;
  char v;/*represents whether the student is present or absent*/
}
interface Common1 /*this will provide those methods that can be implemented in almost all
classes , but will be overridden in each class*/
{
public ArrayList<Student> navigate(ArrayList<Student>a1,ArrayList<Student>a2,int k,char
c);/*navigate function gives a call to some functions that changes the student arraylist
therefore it has a return type of arraylist student*/
public ArrayList<Student> CheckAtt(ArrayList<Student>a1,ArrayList<Student>a2,int k,char c);

public Vector<Questions> TestQues(int k,Vector<Questions> ques,Student obj);

public void CheckMarks(int k, ArrayList<Student>a);
}

 class Student extends Common implements Common1
{
int s1_m,s2_m;
public String div;
Vector<Attendance> s1 = new Vector<Attendance>();
Vector<Attendance> s2 = new Vector<Attendance>();
Student()//default constructor for creating the objects of ArrayList of Students
{
  super(0," ",0);
  this.div = " ";
}

 Student(int  s_id, String s_name , int s_age, String s_div,String u1, String pass1)//overloaded
constructor for initializing the variables.
{

super(s_id, s_name, s_age);
super.Login(u1,pass1);/*to create username and password for every student account*/
this.div= s_div;

}
```

```
public ArrayList<Student> navigate(ArrayList<Student>a1, ArrayList<Student>a2,int k,char c)
/*function from the interface*/
{
  int ch,ch1;
  ArrayList<Student> ar = new ArrayList<Student>();
  System.out.println("Login Successful!");
  System.out.println("Enter any of the following choices");
  System.out.println("Enter 1 to check your attendance for any subject");
  System.out.println("Enter 2 to check your marks");
  System.out.println("Enter 3 for study material");
Scanner sc = new Scanner(System.in);
ch = sc.nextInt();
 Student obj = new Student();
switch(ch)
{

  case 1:
  {

      obj.CheckAtt(a1,a2,k,c);
  }
  break;
  case 2:
  {
    if(c=='A')
    {
obj.CheckMarks(k, a1);
    }

    else if(c=='B')
    {
obj.CheckMarks(k, a2);
    }
  }
  case 3:
  {
    while(true){
      System.out.println("1.Study material for subject java");
      System.out.println("2.study material for subject science");
      System.out.println("3.exit");
      System.out.println("enter choice");
      int chi = sc.nextInt();
      switch(chi){
        case 1:{
```

```
          studynotes.java();
          break;
        }
        case 2:
        {
          studynotes.science();
          break;
        }
        case 3:
        {
          return ar;
          //break;
        }
      }


  }
  }

}
return ar;

}

public Student Initialize(int i )
{
char d;
  int age,id;
  String name,pass1,u1;

  System.out.println("Enter the id of student");
 Scanner sc1 = new Scanner(System.in);
id = sc1.nextInt();
System.out.println("Enter the name of student");
name = sc1.next();
System.out.println("Enter the age");
age = sc1.nextInt();
System.out.println("Enter the password for this student account:");
pass1 = sc1.next();/*every student account will have a unique password to access their account
and the user-id is the id no. of the student*/
u1 = " " +id;
Student obj1 = new Student();
System.out.println("Account Created!");
System.out.println("-----------------");
```

```
switch(i)
{
  case 1:
  {
 Student obj2= new Student(id, name,age,"A",u1,pass1);// calling parametrized constructor to
initialize the values
obj1 = obj2;
  }
  break;
  case 2:
  {
Student obj2= new Student(id, name,age,"B",u1,pass1);
obj1=obj2;
  }
  break;

}
 return obj1;
}

public int StudentLogin(ArrayList <Student> ar)
{/*method used to authenticate a student's login*/
  int i, flag, id_n;
  String p1;
  int c;
  flag=0;
   Scanner sc = new Scanner(System.in);
  System.out.println("Enter your id-no");
id_n= sc.nextInt();
for(i=0;i<ar.size();i++)
{/*loop to check whether user id belongs to the id's of the students in the class*/
  if(id_n==ar.get(i).user_id)
  {
    System.out.println("ID FOUND");
    flag = 1;
    break;
  }
}
  if(flag==0)
 {
   System.out.println("User-ID not found.Please Try Again");
   return -1;
 }
 else
```

```
 {
   Student obj2 = new Student();
   obj2 = ar.get(i);
   System.out.println("Enter the password");
p1 = sc.next();
c = super.CheckLogin(obj2,p1);
if(c==1)
{
  return i;
}

else{
  System.out.println("Wrong Password.Please Try Again");
  return -1;
}
 }


}

public Vector<Questions> TestQues(int k, Vector <Questions>vec, Student obj)
{

  int ch,i;
  String answer;
  Scanner sc = new Scanner(System.in);
String stuans = new String();
System.out.println("GENERAL INSTRUCTIONS:");
System.out.println("This is a fill-in the blank test");
System.out.println("Enter a SINGLE WORD as the answer");
for(i=0;i<3;i++)
{
System.out.println("Question: "+vec.elementAt(i).ques);
System.out.print("Answer:");
stuans = sc.next();
if(stuans.compareToIgnoreCase(vec.elementAt(i).ans)==0)/*comparing student's answer with right
answer ignoring case */
{
  if(k==1)
  {
    obj.s1_m++;
  }
  else if(k==2)
  {
```

```
    obj.s2_m++;
  }
}
}


return vec;
}

public void CheckMarks(int k,ArrayList<Student>a)
{
  Student ob = new Student();
  ob = a.get(k);
Scanner sc = new Scanner(System.in);
  int ch;
  System.out.println("enter 1 to find your marks in science or 2 in java");
  ch = sc.nextInt();
  if(ch==1)
  {
    System.out.println("Your marks in Science are:"+ob.s1_m);
  }
 if(ch==2)
  {
    System.out.println("Your marks in Java are:"+ob.s2_m);
  }

}



public ArrayList<Student> CheckAtt(ArrayList<Student>a1,ArrayList<Student>a2,int k,char c)
{
int i;
Student obj =new Student();
int ch;
Scanner sc = new Scanner(System.in);
if(c=='A')
{
obj = a1.get(k);
}
else if (c=='B')
{
  obj = a2.get(k);
}
```

```
System.out.println("Enter 1 to check attendance for subject 1 or 2 to check attendance for
subject 2");
ch = sc.nextInt();
System.out.println("your attendance for this subject is");
System.out.println("Day      Present/Absent");
if(ch==1)
{
  for(i=0;i<obj.s1.size();i++)
{
  System.out.println(" "+obj.s1.elementAt(i).d+"      "+obj.s1.elementAt(i).v);
}
}
else if(ch==2)
{
   for(i=0;i<obj.s2.size();i++)
{
  System.out.println(" "+obj.s2.elementAt(i).d+"      "+obj.s2.elementAt(i).v);
}
}

return a2;
}

}

class Admin extends Common

{

Admin()
{
  super(0," ",0);
  super.Login("admin9660","answer");// invoking super class method for only  giving authorized
admin access.
}
public int adminLogin()
{Scanner sc3 = new Scanner(System.in);
  String id1 , pas1;
  System.out.println("ADMIN LOGIN");
  System.out.print("Username: ");
  id1 = sc3.next();
  System.out.println(" ");
  System.out.print("Password: ");
  pas1 = sc3.next();
```

```
if((id1.equals(id)==true)&&(pas1.equals(pass)==true))
{
  System.out.println("Access Granted!");
  return 1;
}
else
{
  System.out.println("Admin Access Denied. Please Try Again");
  return 2;
}
}

public void navigate1(ArrayList <Student> a1, ArrayList <Student> a2)
{
  int ch,id,age;
  String pass1,name,u1;
  char c;
  System.out.println("Login Successful!");
  System.out.println("Enter any of the following options");
  System.out.println("1 to add a new student record");
  System.out.println("2 to delete any student record");
  System.out.println("3 to view the attendance status for any day");
  Scanner sc = new Scanner(System.in);
  ch = sc.nextInt();
  switch(ch)
  {
    case 1:
    {
System.out.println("Enter the division of the student");
c = sc.next().charAt(0);
 System.out.println("Enter the id of student");

id = sc.nextInt();
System.out.println("Enter the name of student");
name = sc.next();
System.out.println("Enter the age");
age = sc.nextInt();
System.out.println("Enter the password for this student account:");
pass1 = sc.next();
Main obj = new Main();
u1 = id + " ";
if(c=='A')
{
Student obj2= new Student(id, name,age,"A",u1,pass1);
```

```
a1.add(obj2);

}
else if(c=='B')
{
  Student obj2= new Student(id, name,age,"B",u1,pass1);
a2.add(obj2);
}
    }
    break;
    case 2:
    {
System.out.println("Enter the division of the student");

    }
    break;
    case 3:
    {

    }
  }
}


public ArrayList <Student> Create(ArrayList <Student> ar, int k)
{int i , ns, ndiv;
  Scanner sc2 = new Scanner(System.in);

  System.out.print("Enter the number of students in the Division");
  switch(k)
  {
    case 1:
    {
      System.out.println("A");
      ns=sc2.nextInt();

for(i=0;i<ns;i++)
{
Student obj1 = new Student();
obj1 = obj1.Initialize(1);/*calling Initialize method of student class to Initialize the student
objects*/
ar.add(i,obj1);

}
```

```
     }
    break;
    case 2:
    {
System.out.println("B");

      ns=sc2.nextInt();

for(i=0;i<ns;i++)
{
Student obj1 = new Student();
obj1 = obj1.Initialize(2);
ar.add(i,obj1);
}

    }

  }
 return ar;

  }
public Instructor Create(Instructor obj)/*overloaded method for creating Instructor objects */
{
  String s;
  System.out.println("Enter the subject of the instructor");
  Scanner ob1 = new Scanner(System.in);
  s = ob1.next();
Instructor obj1 = new Instructor();
obj1 = obj1.Initialize(s);
  return obj1;
}

public int check(Admin obj)
{String p2;
Scanner sc = new Scanner(System.in);
int ch;
  System.out.println("Enter the password");
p2= sc.next();
ch = super.CheckLogin(obj, p2);
  return ch;
}

}
```

```
class Instructor extends Common implements Common1{
  public String sub;
Instructor()
{
  super(0," ",0);
this.sub = " ";
}
Instructor (int  i_id, String i_name , int i_age, String subj,String u1, String pass1)
{
super(i_id, i_name, i_age);
super.Login(u1,pass1);
this.sub = subj;
}
public Instructor Initialize(String s)
{
  String iname,isub,ipass,s_id;
  int i_age,i_id;
  Scanner sc = new Scanner(System.in);
  System.out.println("Enter the ID no of this Instructor");
  i_id = sc.nextInt();
  System.out.println("Enter the name of the Instructor");

iname = sc.next();
System.out.println("Enter the age of the instructor");
i_age = sc.nextInt();

System.out.println("Enter the password for the instructor account");
ipass = sc.next();
s_id = String.valueOf(i_id);


Instructor obj = new Instructor(i_id,iname,i_age,s,s_id,ipass);
return obj;
}
public int InstructorLogin(Instructor ob)
{
String p; int id,c;
Scanner sc= new Scanner(System.in);

  System.out.println("Enter your username please");
  id = sc.nextInt();
  if(id!=ob.user_id)
  { return 1;}
  else if (id==ob.user_id)
```

```
  {
    System.out.println("Enter the password");
    p = sc.next();
    c = super.CheckLogin(ob,p);
    if(c==1)
    {

      return 0;
    }
    else
    {
      return 1;
    }
  }
  else return 0;

}

 public ArrayList<Student> navigate(ArrayList<Student>a1, ArrayList<Student> a2,int k,char cl)
 {
   ArrayList <Student> ar = new ArrayList<Student>();
System.out.println("Welcome!");
System.out.println("Enter any of the following choices");
 char c;
System.out.println("Enter 1 to mark/check attendance");
System.out.println("Enter 2 to check the marks of all students in a class");

Scanner sc = new Scanner(System.in);
int ch;
Instructor obj = new Instructor();
ch = sc.nextInt();
switch(ch)
{
  case 1:
{


if(cl=='A')
{
  a1 = obj.CheckAtt(a1, a2, k,cl);

  ar.addAll(a1);
}
else if(cl=='B')
```

```
{
  a2 = obj.CheckAtt(a1,a2,k,cl);

 ar.addAll(a2);
}


}

  break;
  case 2:
  {
if(cl=='A')
{
obj.CheckMarks(k,a1);
}
else if(cl=='B')
{
  obj.CheckMarks(k,a2);
}
  }

 }

return ar;
 }

 public Vector<Questions> TestQues(int k, Vector<Questions>ques,Student obj)
 {
   int i,ch;
   Scanner sc = new Scanner(System.in);
   Vector <Questions> vec = new Vector <Questions>(3);


  System.out.println("Choose any three questions for your test by entering their index
numbers");
for(i=0;i<3;i++)
{
ch = sc.nextInt();
Questions ob = new Questions();
ob = ques.elementAt(ch-1);

vec.add(ob);
System.out.println("Question added successfully ");
```

```
}

return vec;

 }
 public void CheckMarks(int k,ArrayList<Student>a)
 {
   int i;
   Student obj = new Student();
if(k==1)
{
  System.out.println("The marks of the students in Science are:");
  System.out.println("Roll no              Name                Marks");
  for(i=0;i<a.size();i++)
  {
obj = a.get(i);
System.out.println(""+obj.user_id+"      "+obj.user_name+"     "+obj.s1_m);

  }
}

else if(k==2)
{
   System.out.println("The marks of the students in Java are:");
  System.out.println("Roll no              Name                Marks");
  for(i=0;i<a.size();i++)
  {
obj = a.get(i);
System.out.println(""+obj.user_id+"      "+obj.user_name+"     "+obj.s2_m);

  }
}
 }

 public ArrayList<Student> CheckAtt(ArrayList<Student>a1,ArrayList<Student>a2,int k,char c)
 {

int i;
char ch;
int d,s=0;
ch = c;
int r,day;
int flag =0;
```

```
ArrayList<Student> ar = new ArrayList<Student> ();
System.out.print("Rollno      Name     ");
System.out.println("");
if(ch=='A')
{
  for(i=0;i<a1.size();i++)
{
System.out.println(" "+a1.get(i).user_id+ "  "+"      "+a1.get(i).user_name);
}
}
else if (ch=='B')
{
  for(i=0;i<a2.size();i++)
{
System.out.println(" "+a2.get(i).user_id+ "  "+"      "+a2.get(i).user_name);
}
}
Scanner sc = new Scanner(System.in);
System.out.println("Enter 1 to see the attendance list for class  " +ch+ " or enter 2 to mark
the attendance for the students of the class ");
r = sc.nextInt();
if(r==1)
{
  System.out.println("Enter the day number to check the attendance");
  day = sc.nextInt();
  if(ch=='A')
{
ar = a1;
}

else if(ch=='B')
{
  ar = a2;
}
Student oc = new Student();
Student od = new Student();
oc = ar.get(0);
  System.out.println("Roll no:       Name:      P/A");

  if(k==1)
  {
   for(i=0;i<oc.s1.size();i++)
   {
     if(oc.s1.elementAt(i).d==day)
```

```
      {
       s=i;
flag = 1;
break;
      }
   }
    for(i=0;i<ar.size();i++)
   {
     od = ar.get(i);
     System.out.println(" "+od.user_id+"              "+od.user_name+"
"+od.s1.get(s).v);
   }
  }
else if(k==2)
{
   for(i=0;i<oc.s2.size();i++)
   {
     if(oc.s2.elementAt(i).d==day)
     {
      s=i;
flag = 1;
break;
      }
   }
    for(i=0;i<ar.size();i++)
   {
     od = ar.get(i);
     System.out.println(" "+od.user_id+"               "+od.user_name+"
"+od.s1.get(s).v);
   }
}




}
else {
System.out.println("Enter the day number to mark the attendance");
d = sc.nextInt();
System.out.println("Press P to mark Present or A to mark Absent for each student roll-no wise");

if(ch=='A')
{
  if(k==1)
```

```
  {
    for(i=0;i<a1.size();i++)
{
  Student obj1 = new Student();
  obj1 = a1.get(i);
  Attendance obj = new Attendance();
  obj.d =d;
  c = sc.next().charAt(0);
  obj.v=c;
 obj1.s1.add(obj);
 a1.set(i,obj1);

}
  }

   else if(k==2)
  {
    for(i=0;i<a1.size();i++)
{
  Student obj1 = new Student();
  obj1 = a1.get(i);
  Attendance obj = new Attendance();
  obj.d =d;
  c = sc.next().charAt(0);
  obj.v=c;
 obj1.s2.add(obj);
 a1.set(i,obj1);

}
  }


ar.addAll(a1);
}
else if(ch=='B')
{
    if(k==1)
  {
    for(i=0;i<a2.size();i++)
{
  Student obj1 = new Student();
  obj1 = a2.get(i);
  Attendance obj = new Attendance();
  obj.d =d;
```

```
  c = sc.next().charAt(0);
  obj.v=c;
 obj1.s1.add(obj);
 a2.set(i,obj1);

}

  }

   else if(k==2)
  {
    for(i=0;i<a2.size();i++)
{
  Student obj1 = new Student();
  obj1 = a2.get(i);
  Attendance obj = new Attendance();
  obj.d =d;
  c = sc.next().charAt(0);
  obj.v=c;
 obj1.s2.add(obj);
 a2.set(i,obj1);

}

  }


ar.addAll(a2);
}
return ar;
 }
 return ar;
}

 }
```

Conclusion: In this experiment , we were able to implement the case study of our mini project and analyse and create a class diagram for the same. A lso we were able to understand and identify the attributes the attributes and methods for each of the classes and the various Generalization/Association relationships between them , through the class diagram

Date: _____ Signature of faculty in-charge

Post Lab Descriptive Questions (Add questions from examination point view)

1. Consider the following class:

```
public class TypeOfVariable{
        public static int a;
        int b,c;
        public void printValue(){
                int x = 10;
        }
        public static void main(String args[]){
                TypeOfVariable object=new TypeOfVariable();
                object.printValue();
        }
}
```

a). What are the class/static variables?

The class variable is ‘a’, as it has been declared with the keyword static one copy of it will be accessible to all members of the class

b). What are the instance variables?

int b , c (since every object of the class will have it’s own copy of the instance variables)

c.)What are local variables?

int x

2.What is the output from the following code:

```
public class Test
{
    static int x = 11;
    private int y = 33;
    public void method1(int x)
    {
        Test t = new Test();
        this.x = 22;
        y = 44;

        System.out.println("Test.x: " + Test.x);
        System.out.println("t.x: " + t.x);
        System.out.println("t.y: " + t.y);
        System.out.println("y: " + y);
    }

    public static void main(String args[])
    {
        Test t = new Test();
        t.method1(5);
    }
}
```

Output:

Test.x: 22

t.x: 22

t.y: 33

y: 44