K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Computer Engineering

Batch: D2 **Roll No.: 16010122323**

Experiment / assignment / tutorial No.06

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

Signature of the Staff In-charge with date

Group Number: 3

Group Members:

Roll Nos.: 16010122323 16010122308 & 16010122309

Title for Case Study:

Hospital Management System

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:

CO1: Understand the features of object oriented programming compared with procedural approach with C++ and Java.

CO2: Explore arrays, vectors, classes and objects in C++ and Java.

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

CO4: Explore the interface, exceptions, multithreading, packages

---

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:

Staff: Contains the details of staff members

Doctor: Contains the details of doctors

Patient: Contains the details of patients

Medical: Contains details of medicines

Facility: Contains details about facilities available

Lab: Contains details about Lab

Identify Attributes for each class:

Staff: sid, sname, desg, sex, salary

Doctor: did, dname, specilist, appoint, doc_qual, droom

Patient: pid, pname, disease, sex, admit_status, age

Medical: med_name, med_comp, exp_date, med_cost, count

Facility: fac_name

Lab: facility, lab_cost

Identify List of Methods in each classes:

Staff: new_staff(), staff_info()

Doctor: new_doctor(), doctor_info

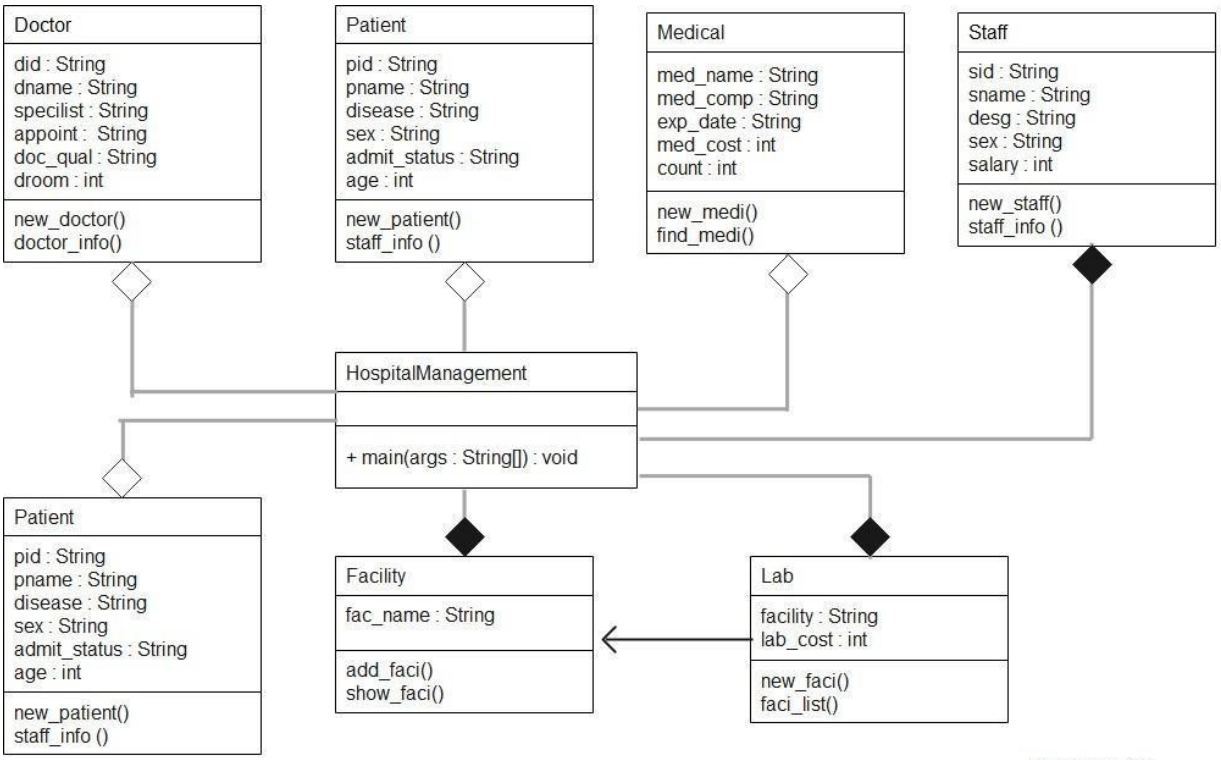
Patient: new_patient(), patient_info()

Medical: new_medi()

Facility: add_faci(), show_faci()

Lab: new_faci(), faci_list()

Class Diagram:



By Group 5 C3
Aayush Durgule 16010120203
Saksham Raina 16010120204

Algorithm:

1. Start
2. import java.util.* and java.util.calendar
3. Declare a class Staff
4. Declare attributes of class Staff sid, sname, desg, sex, salary.
5. Declare a method new_staff in Staff class to set the attributes for new members ofthe class.
6. Declare a method staff_info to print these details.
7. Declare a class Doctor
8. Declare attributes of class Doctor did, dname, specilist, appoint, doc_qual, droom
9. Declare method new_doctor and doctor_jnfo which has same purpose as class Staff methods had for Staff.
10. In the same way as done in previous steps declare classes Patient, Medical, Facility, Lab
11. Declare following attributes for the respective classes

    Patient: pid, pname, disease, sex, admit_status, age

    Medical: med_name, med_comp, exp_date, med_cost, count

    Facility: fac_name

    Lab: facility, lab_cost.
12. Now, declare following methods for the classes and these methods serve the same purpose as described for class Staff and Doctor i.e. to set the attributes for new members of the class and to print the details.

    Patient: new_patient(), patient_info()
    Medical: new_medi()
    Facility: add_faci(), show_faci()
    Lab: new_faci(), faci_list()
13. In the main method define a String containing months names
14. Create a object calendar for calendar.
15. Declare variables count1, count2, count3, count4, count5,count6.
16. Print current date and time.

17. Create Array of objects for the respective classes.
18. Initialize the for loops for the classes which are used to add new objects to the class.
19. Add some objects to this array list.
20. Get choice from the user.
21. If 1 is selected then user gets two options 1. Add New Entry 2. Existing Doctorslist
22. After to go back enter 1 and 0 for main menu.
23. According to choose entered by the user add the objects to the class or display the details using the methods defined earlier.
24. Repeat step 22 and 23 for all other options.
25. If user enters 1.
26. Exit

Implementation details: (Class Diagram and Code)

```
import java.util.*;
import java.util.Calendar;

class Staff {
    String sid, sname, desg, sex;
    int salary;
    void new_staff() {
        Scanner input = new Scanner(System.in);
        System.out.print("ID:-");
        sid = input.nextLine();
        System.out.print("Name:-");
        sname = input.nextLine();
        System.out.print("Designation:-");
        desg = input.nextLine();
        System.out.print("Sex:-");
        sex = input.nextLine();
        System.out.print("Salary:-");
        salary = input.nextInt();
    }
    void staff_info() {
        System.out.println(sid + "\t" + sname + "\t" + sex + "\t" + salary);
    }
}
class Doctor {
    String did, dname, specilist, appoint, doc_qual;
    int droom;
    void new_doctor() {
        Scanner input = new Scanner(System.in);
```

```
        System.out.print("ID:-");
        did = input.nextLine();
        System.out.print("Name:-");
        dname = input.nextLine();
        System.out.print("Specialization:-");
        specilist = input.nextLine();
        System.out.print("Work Time:-");
        appoint = input.nextLine();
        System.out.print("Qualification:-");
        doc_qual = input.nextLine();
        System.out.print("Room No..:-");
        droom = input.nextInt();
    }
    void doctor_info() {
        System.out.println(did + "\t" + dname + "  \t" + specilist + "      \t"
+ appoint + "     \t" + doc_qual + "        \t" + droom);
    }
}
class Patient {
    String pid, pname, disease, sex, admit_status;
    int age;
    void new_patient() {
        Scanner input = new Scanner(System.in);
        System.out.print("Id:-");
        pid = input.nextLine();
        System.out.print("Name:-");
        pname = input.nextLine();
        System.out.print("Disease:-");
        disease = input.nextLine();
        System.out.print("Sex:-");
        sex = input.nextLine();
        System.out.print("Admit Status:-");
        admit_status = input.nextLine();
        System.out.print("Age:-");
        age = input.nextInt();
    }
    void patient_info() {
        System.out.println(pid + "\t" + pname + " \t" + disease + "     \t" +
sex + "      \t" + admit_status + "\t" + age);
    }
}
class Medical {
    String med_name, med_comp, exp_date;
    int med_cost, count;
    void new_medi() {
        Scanner input = new Scanner(System.in);
        System.out.print("Name:-");
        med_name = input.nextLine();
        System.out.print("Comp:-");
        med_comp = input.nextLine();
        System.out.print("Exp. Date:-");
        exp_date = input.nextLine();
```

```
        System.out.print("Cost:-");
        med_cost = input.nextInt();
        System.out.print("No of Units:-");
        count = input.nextInt();
    }
    void find_medi() {
        System.out.println(med_name + "  \t" + med_comp + "     \t" + exp_date
+ "      \t" + med_cost);
    }
}

class Facility{
    String fac_name;
    void add_faci() {
        Scanner input = new Scanner(System.in);
        System.out.print("Facility:-");
        fac_name = input.nextLine();
    }
    void show_faci() {
        System.out.println(fac_name);
    }
}

class Lab extends Facility{
    String facility;
    int lab_cost;
    void new_faci() {
        Scanner input = new Scanner(System.in);
        System.out.print("Facility:-");
        facility = input.nextLine();
        System.out.print("Cost:-");
        lab_cost = input.nextInt();
    }
    void faci_list() {
        System.out.println(facility + "\t\t" + lab_cost);
    }
}

public class HospitalManagement {
    public static void main(String[] args) {
        String[] months = {"Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul",
"Aug", "Sep", "Oct", "Nov", "Dec"};
        Calendar calendar = Calendar.getInstance();
        int count1 = 4, count2 = 4, count3 = 4, count4 = 4, count5 = 4, count6
= 4;
        System.out.println("\t\tWelcome to Hospital Management System");
        System.out.print("Date: " + months[calendar.get(Calendar.MONTH)] + " "
+ calendar.get(Calendar.DATE) + " " + calendar.get(Calendar.YEAR));
        System.out.println("\t\t\t\t\t\tTime: " + calendar.get(Calendar.HOUR)
+ ":" + calendar.get(Calendar.MINUTE) + ":" + calendar.get(Calendar.SECOND));
        Doctor[] d = new Doctor[25];
        Patient[] p = new Patient[100];
```

```
        Lab[] l = new Lab[20];
        Facility[] f = new Facility[20];
        Medical[] m = new Medical[100];
        Staff[] s = new Staff[100];
        int i;
        for (i = 0; i < 25; i++)
            d[i] = new Doctor();
        for (i = 0; i < 100; i++)
            p[i] = new Patient();
        for (i = 0; i < 20; i++)
            l[i] = new Lab();
        for (i = 0; i < 20; i++)
            f[i] = new Facility();
        for (i = 0; i < 100; i++)
            m[i] = new Medical();
        for (i = 0; i < 100; i++)
            s[i] = new Staff();

        d[0].did = "11";d[0].dname = "Dr.A";d[0].specilist =
"ENT";d[0].appoint = "5-11AM";d[0].doc_qual = "MBBS,MD";d[0].droom = 17;
        d[1].did = "15";d[1].dname = "Dr.B";d[1].specilist =
"Physician";d[1].appoint = "10-3AM";d[1].doc_qual = "MBBS,MD";d[1].droom = 25;
        d[2].did = "17";d[2].dname = "Dr.C";d[2].specilist =
"Surgeon";d[2].appoint = "8-2AM";d[2].doc_qual = "BDM";d[2].droom = 5;
        d[3].did = "33";d[3].dname = "Dr.D";d[3].specilist =
"Artho";d[3].appoint = "10-4PM";d[3].doc_qual = "MBBS,MS";d[3].droom = 40;

        p[0].pid = "12";p[0].pname = "P";p[0].disease = "Cancer";p[0].sex =
"Male";p[0].admit_status = "Y";p[0].age = 30;
        p[1].pid = "13";p[1].pname = "Q";p[1].disease = "Cold";p[1].sex =
"Male";p[1].admit_status = "Y";p[1].age = 23;
        p[2].pid = "14";p[2].pname = "R";p[2].disease = "Joint Pain";p[2].sex
= "Male";p[2].admit_status = "Y";p[2].age = 45;
        p[3].pid = "15";p[3].pname = "S";p[3].disease = "Diabetes";p[3].sex =
"Male";p[3].admit_status = "Y";p[3].age = 25;

        m[0].med_name = "Corex";m[0].med_comp = "Cino pvt";m[0].exp_date = "9-
5-25";m[0].med_cost = 55;m[0].count = 8;
        m[1].med_name = "Nytra";m[1].med_comp = "Ace pvt";m[1].exp_date = "4-
4-26";m[1].med_cost = 500;m[1].count = 5;
        m[2].med_name = "Brufa";m[2].med_comp = "Reckitt";m[2].exp_date = "12-
7-27";m[2].med_cost = 50;m[2].count = 56;
        m[3].med_name = "Pride";m[3].med_comp = "DDF pvt";m[3].exp_date = "12-
4-22";m[3].med_cost = 1100;m[3].count = 100;

        l[0].facility = "X-ray     ";l[0].lab_cost = 800;
        l[1].facility = "CT Scan   ";l[1].lab_cost = 1200;
        l[2].facility = "OR Scan   ";l[2].lab_cost = 500;
        l[3].facility = "Blood Bank";l[3].lab_cost = 50;

        f[0].fac_name = "Ambulance";
        f[1].fac_name = "Admit Facility ";
```

```
        f[2].fac_name = "Canteen";
        f[3].fac_name = "Emergency";

        s[0].sid = "22";s[0].sname = "W";s[0].desg = "Worker";s[0].sex =
"Male";s[0].salary = 5000;
        s[1].sid = "23";s[1].sname = "X";s[1].desg = "Nurse";s[1].sex =
"Female";s[1].salary = 10000;
        s[2].sid = "24";s[2].sname = "Y";s[2].desg = "Worker";s[2].sex =
"Male";s[2].salary = 5000;
        s[3].sid = "25";s[3].sname = "Z";s[3].desg = "Nurse";s[3].sex =
"Female";s[3].salary = 10000;

        Scanner input = new Scanner(System.in);
        int choice, j, c1, status = 1, s1 = 1, s2 = 1, s3 = 1, s4 = 1, s5 = 1,
s6 = 1;
        while (status == 1) {
            System.out.println("\nMAIN MENU");
            System.out.println("1.Doctos  2. Patients  3.Medicines
4.Laboratories  5. Facilities  6. Staff ");
            choice = input.nextInt();
            switch (choice) {
                case 1: {
                    System.out.println("DOCTOR SECTION");
                    s1 = 1;
                    while (s1 == 1) {
                        System.out.println("1.Add New Entry\n2.Existing
Doctors List");
                        c1 = input.nextInt();
                        switch (c1) {
                            case 1: {
                                d[count1].new_doctor();count1++;
                                break;
                            }
                            case 2: {
                                System.out.println("id \t Name\t Specilist \t
Timing \t Qualification \t Room No.");
                                for (j = 0; j < count1; j++) {
                                    d[j].doctor_info();
                                }
                                break;
                            }
                        }
                        System.out.println("\nReturn to Back Press 1 and for
Main Menu Press 0");
                        s1 = input.nextInt();
                    }
                    break;
                }
                case 2: {
                    System.out.println("PATIENT SECTION");
                    s2 = 1;
                    while (s2 == 1) {
```

```
                        System.out.println("1.Add New Entry\n2.Existing
Patients List");
                        c1 = input.nextInt();
                        switch (c1) {
                            case 1: {
                                p[count2].new_patient();count2++;
                                break;
                            }
                            case 2: {
                                System.out.println("id \t Name \t Disease \t
Gender \t Admit Status \t Age");
                                for (j = 0; j < count2; j++) {
                                    p[j].patient_info();
                                }
                                break;
                            }
                        }
                        System.out.println("\nReturn to Back Press 1 and for
Main Menu Press 0");
                        s2 = input.nextInt();
                    }
                    break;
                }
                case 3: {
                    s3 = 1;
                    System.out.println("MEDICINE SECTION");
                    while (s3 == 1) {
                        System.out.println("1.Add New Entry\n2.Existing
Medicines List");
                        c1 = input.nextInt();
                        switch (c1) {
                            case 1: {
                                m[count3].new_medi();count3++;
                                break;
                            }
                            case 2: {
                                System.out.println("Name \t Company \t Expiry
Date \t Cost");
                                for (j = 0; j < count3; j++) {
                                    m[j].find_medi();
                                }
                                break;
                            }
                        }
                        System.out.println("\nReturn to Back Press 1 and for
Main Menu Press 0");
                        s3 = input.nextInt();
                    }
                    break;
                }
                case 4: {
                    s4 = 1;
```

```
                    System.out.println("LABORATORY SECTION");
                    while (s4 == 1) {
                        System.out.println("1.Add New Entry \n2.Existing
Laboratories List");
                        c1 = input.nextInt();
                        switch (c1) {
                            case 1: {
                                l[count4].new_faci();count4++;
                                break;
                            }
                            case 2: {
                                System.out.println("Facilities\t\t Cost");
                                for (j = 0; j < count4; j++) {
                                    l[j].faci_list();
                                }
                                break;
                            }
                        }
                        System.out.println("\nReturn to Back Press 1 and for
Main Menu Press 0");
                        s4 = input.nextInt();
                    }
                    break;
                }
                case 5: {
                    s5 = 1;
                    System.out.println("HOSPITAL FACILITY SECTION");
                    while (s5 == 1) {
                        System.out.println("1.Add New Facility\n2.Existing
Facilities List");
                        c1 = input.nextInt();
                        switch (c1) {
                            case 1: {
                                f[count5].add_faci();count5++;
                                break;
                            }
                            case 2: {
                                System.out.println("Hospital  Facility are:");
                                for (j = 0; j < count5; j++) {
                                    f[j].show_faci();
                                }
                                break;
                            }
                        }
                        System.out.println("\nReturn to Back Press 1 and for
Main Menu Press 0");
                        s5 = input.nextInt();
                    }
                    break;
                }
                case 6: {
                    s6 = 1;
```

```
                    System.out.println("STAFF SECTION");
                    while (s6 == 1) {
                        String a = "Nurse", b = "Worker", c = "Security";
                        System.out.println("1.Add New Entry \n2.Existing
Nurses List\n3.Existing Workers List \n4.Existing Security List");
                        c1 = input.nextInt();
                        switch (c1) {
                            case 1: {
                                s[count6].new_staff();count6++;
                                break;
                            }
                            case 2: {
                                System.out.println("id \t Name \t Gender \t
Salary");
                                for (j = 0; j < count6; j++) {
                                    if (a.equals(s[j].desg))
                                        s[j].staff_info();
                                }
                                break;
                            }
                            case 3: {
                                System.out.println("id \t Name \t Gender \t
Salary");
                                for (j = 0; j < count6; j++) {
                                    if (b.equals(s[j].desg))
                                        s[j].staff_info();
                                }
                                break;
                            }
                            case 4: {
                                System.out.println("id \t Name \t Gender \t
Salary");
                                for (j = 0; j < count6; j++) {
                                    if (c.equals(s[j].desg))
                                        s[j].staff_info();
                                }
                                break;
                            }
                        }
                        System.out.println("\nReturn to Back Press 1 and for
Main Menu Press 0");
                        s6 = input.nextInt();
                    }
                    break;
                }
                default: {
                    System.out.println("Input Not Valid!");
                }
            }
            System.out.println("\nReturn to MAIN MENU Press 1");
            status = input.nextInt();
        }
```

```
    }
}
```

Output:

```
        Welcome to Hospital Management System
Date: Nov 14 2021                       Time: 10:44:52

MAIN MENU
1.Doctos  2. Patients  3.Medicines  4.Laboratories  5. Facilities  6. Staff
1
DOCTOR SECTION
1.Add New Entry
2.Existing Doctors List
2
id   Name    Specilist   Timing      Qualification   Room No.
11  Dr.A    ENT         5-11AM      MBBS,MD         17
15  Dr.B    Physician       10-3AM      MBBS,MD         25
17  Dr.C    Surgeon         8-2AM       BDM         5
33  Dr.D    Artho       10-4PM      MBBS,MS         40

Return to Back Press 1 and for Main Menu Press 0
1
1.Add New Entry
2.Existing Doctors List
1
ID:-45
Name:-E
Specialization:-Heart
Work Time:-9-3PM
Qualification:-MBBS
Room No..:-23
```

```
Return to Back Press 1 and for Main Menu Press 0
1
1.Add New Entry
2.Existing Doctors List
2
id   Name    Specilist   Timing      Qualification   Room No.
11  Dr.A    ENT         5-11AM      MBBS,MD         17
15  Dr.B    Physician       10-3AM      MBBS,MD         25
17  Dr.C    Surgeon         8-2AM       BDM         5
33  Dr.D    Artho       10-4PM      MBBS,MS         40
45  E   Heart       9-3PM       MBBS        23
```

```
Return to MAIN MENU Press 1
1

MAIN MENU
1.Doctos  2. Patients  3.Medicines  4.Laboratories  5. Facilities  6. Staff
4
LABORATORY SECTION
1.Add New Entry
2.Existing Laboratories List
2
Facilities       Cost
X-ray           800
CT Scan         1200
OR Scan         500
Blood Bank      50

Return to Back Press 1 and for Main Menu Press 0
0

Return to MAIN MENU Press 1
0

Process finished with exit code 0
```

Conclusion

The aim of the experiment is verified.

Date:_____ **Signature of faculty in-charge**

Post Lab Descriptive Questions

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?

A static variable is common to all the instances (or objects) of the class because it is a class level variable. In other words you can say that only a single copy of static variable is created and shared among all the instances of the class. Memory allocation for such variables only happens once when the class is loaded in the memory.

Few Important Points:

Static variables are also known as Class Variables.

Unlike non-static variables, such variables can be accessed directly in static and non-static methods.

Here a is static variable.

b). What are the instance variables?

Instance variable in Java is used by Objects to store their states. Variables which are defined without the STATIC keyword and are Outside any method declaration are Object-specific and are known as instance variables. They are called so because their values are instance specific and are not shared among instances.

Here b,c are instance variables.

c.) What are local variables?

A local variable in Java is a variable that's declared within the body of a method. Then you can use the variable only within that method. Other methods in the class aren't even aware that the variable exists. You don't specify static on a declaration for a local variable.

Here x is a local variable.

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
```