

Assignment – 01

Module 1- Introduction to OOP in Java

CODE:

1.

```
J assignment1a.java • J assignment1b.java Extension: Ex assignment1a
J assignment1a.java > Patient
1  import java.util.Scanner;
2
3  class Person {
4      String name;
5      int age;
6      String contactInfo;
7
8      Person(String name, int age, String contactInfo) {
9          this.name = name;
10         this.age = age;
11         this.contactInfo = contactInfo;
12     }
13
14     void displayDetails() {
15         System.out.println("Name: " + name);
16         System.out.println("Age: " + age);
17         System.out.println("Contact Info: " + contactInfo);
18     }
19 }
20
21 class Doctor extends Person {
22     String specialization;
23     String availability;
24
25     Doctor(String name, int age, String contactInfo, String specialization, String availability) {
26         super(name, age, contactInfo);
27         this.specialization = specialization;
28         this.availability = availability;
29     }
30
31     @Override
32     void displayDetails() {
33         super.displayDetails();
34         System.out.println("Specialization: " + specialization);
35         System.out.println("Availability: " + availability);
36         System.out.println(x:"-----");
37     }
38 }
39
40 class Patient extends Person {
41     String disease;
42     String admittedDate;
43
44     Patient(String name, int age, String contactInfo, String disease, String admittedDate) {
45         super(name, age, contactInfo);
46         this.disease = disease;
47         this.admittedDate = admittedDate;
48     }
49 }
```

```

49  @Override
50  void displayDetails() {
51      super.displayDetails();
52      System.out.println("Disease: " + disease);
53      System.out.println("Admitted Date: " + admittedDate);
54      System.out.println(x:"-----");
55  }
56  }
57
58  public class assignment1a {
59      Run | Debug
60      public static void main(String[] args) {
61          Scanner sc = new Scanner(System.in);
62
63          System.out.println(x:"Enter Doctor Details:");
64          System.out.print(s:"Name: ");
65          String dName = sc.nextLine();
66          System.out.print(s:"Age: ");
67          int dAge = sc.nextInt();
68          sc.nextLine();
69          System.out.print(s:"Contact Info: ");
70          String dContact = sc.nextLine();
71          System.out.print(s:"Specialization: ");
72          String specialization = sc.nextLine();
73
74          Doctor doctor = new Doctor(dName, dAge, dContact, specialization, availability);
75
76          System.out.println(x:"\nEnter Patient Details:");
77          System.out.print(s:"Name: ");
78          String pName = sc.nextLine();
79          System.out.print(s:"Age: ");
80          int pAge = sc.nextInt();
81          sc.nextLine();
82          System.out.print(s:"Contact Info: ");
83          String pContact = sc.nextLine();
84          System.out.print(s:"Disease: ");
85          String disease = sc.nextLine();
86          System.out.print(s:"Admitted Date: ");
87          String admittedDate = sc.nextLine();
88
89          Patient patient = new Patient(pName, pAge, pContact, disease, admittedDate);
90
91          System.out.println(x:"\n--- Doctor Details ---");
92
93          System.out.println(x:"\n--- Doctor Details ---");
94          doctor.displayDetails();
95
96          System.out.println(x:"--- Patient Details ---");
97          patient.displayDetails();
98
99          sc.close();
100      }

```

OUTPUT:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS assignment1a
PS C:\Users\91788\Downloads\java> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\91788\AppData\Roaming\Code\User\workspaceStorage\f5fa97c5fc5359ad7a9244a856664d07\redhat.java\jdt_ws\java_deb43c15\bin' 'assignment1a'
Enter Doctor Details:
Name: Dr. Akshay Kothari
Age: 37
Contact Info: 9867457869
Specialization: Cardiologist
Availability: 11AM - 2PM

Enter Patient Details:
Name: Manasvi Patil
Age: 21
Contact Info: 7765487939
Disease: Fever
Admitted Date: 16/08/2025

--- Doctor Details ---
Name: Dr. Akshay Kothari
Age: 37
Contact Info: 9867457869
Specialization: Cardiologist
Availability: 11AM - 2PM
-----
--- Patient Details ---
Name: Manasvi Patil
Age: 21
Contact Info: 7765487939
Disease: Fever
```

CODE:

2.

```
J assignment1a.java • J assignment1b.java • Extension: Extension Pack for Java
J assignment1b.java > assignment1b
1 import java.util.Scanner;
2 import java.util.Vector;
3
4 class Item {
5     int itemId;
6     String itemName;
7     int quantity;
8
9     Item(int itemId, String itemName, int quantity) {
10         this.itemId = itemId;
11         this.itemName = itemName;
12         this.quantity = quantity;
13     }
14
15     void displayItem() {
16         System.out.println("Item ID: " + itemId +
17                             ", Name: " + itemName +
18                             ", Quantity: " + quantity);
19     }
20 }
21
22 public class assignment1b {
23     static Scanner sc = new Scanner(System.in);
24
25     public static void manageUsingArray() {
26         Item[] items = new Item[5];
27         int count = 0;
28
29         while (count < 5) {
```

```

29     while (count < 5) {
30         System.out.println("Enter details of Item #" + (count + 1));
31         System.out.print(s:"Item ID: ");
32         int id = sc.nextInt();
33         System.out.print(s:"Item Name: ");
34         sc.nextLine(); // consume newline
35         String name = sc.nextLine();
36         System.out.print(s:"Quantity: ");
37         int qty = sc.nextInt();
38
39         items[count] = new Item(id, name, qty);
40         count++;
41
42         if (count == 5) {
43             System.out.println(x:"Array is full. Cannot add more items.");
44             break;
45         }
46     }
47
48     System.out.println(x:"\nInventory (Array based: fixed size) ");
49     for (int i = 0; i < count; i++) {
50         items[i].displayItem();
51     }
52 }

```

```

53
54 public static void manageUsingVector() {
55     Vector<Item> vector = new Vector<>();
56     char choice = 'y';
57
58     while (choice == 'y' || choice == 'Y') {
59         System.out.println(x:"Enter details of new Item:");
60         System.out.print(s:"Item ID: ");
61         int id = sc.nextInt();
62         System.out.print(s:"Item Name: ");
63         sc.nextLine();
64         String name = sc.nextLine();
65         System.out.print(s:"Quantity: ");
66         int qty = sc.nextInt();
67
68         vector.add(new Item(id, name, qty));
69
70         System.out.print(s:"Add another item? (y/n): ");
71         choice = sc.next().charAt(index:0);
72     }
73
74     System.out.println(x:"\n--- Inventory (Vector based: dynamic size) ---");
75     for (Item it : vector) {
76         it.displayItem();
77     }
78 }

```

```

79
80 public static void main(String[] args) {
81     System.out.println(x:"Inventory using ARRAY (fixed to 5 items):");
82     manageUsingArray();
83
84     System.out.println(x:"\nInventory using VECTOR (dynamic/unlimited items):");
85     manageUsingVector();
86 }
87 }

```

OUTPUT:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Run: assignment1b + - [ ] ... [ ]

PS C:\Users\91788\Downloads\java> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\91788\AppData\Roaming\Code\User\workspaceStorage\f5fa97c5fc5359ad7a9244a856664d07\redhat.java\jdt_ws\java_de
b43c15\bin' 'assignment1b'
Inventory using ARRAY (fixed to 5 items):
Enter details of Item #1
Item ID: 101
Item Name: Sugar
Quantity: 10
Enter details of Item #2
Item ID: 102
Item Name: Rice
Quantity: 5
Enter details of Item #3
Item ID: 103
Item Name: Salt
Quantity: 8
Enter details of Item #4
Item ID: 104
Item Name: Biscuits
Quantity: 15
Enter details of Item #5
Item ID: 105
Item Name: Oil
Quantity: 5
Array is full. Cannot add more items.

Inventory (Array based: fixed size)
Item ID: 101, Name: Sugar, Quantity: 10
Item ID: 102, Name: Rice, Quantity: 5
Item ID: 103, Name: Salt, Quantity: 8
Item ID: 104, Name: Biscuits, Quantity: 15
Item ID: 105, Name: Oil, Quantity: 5
```

```
Inventory using VECTOR (dynamic/unlimited items):
Enter details of new Item:
Item ID: 201
Item Name: Shampoo
Quantity: 10
Add another item? (y/n): y
Enter details of new Item:
Item ID: 202
Item Name: Soap
Quantity: 5
Add another item? (y/n): n

--- Inventory (Vector based: dynamic size) ---
Item ID: 201, Name: Shampoo, Quantity: 10
Item ID: 202, Name: Soap, Quantity: 5
PS C:\Users\91788\Downloads\java>
```