

Compilation of Object-Oriented Programming Activities



In Partial fulfillment of the requirements in
IT 5/L: IT Elective 2

Submitted by:
CAMPILAN, STEPHEN JOHN T.

Submitted to:
JOHN RAVEN MANULAT, MIT

December 2025

Table of Contents

Table of Contents	1
Array Java Code.....	2
Array Java Code Sample Output	3
Car Modification	4
Car Modification Sample Output	5
Car Performance	6
Car Performance Sample Output	7
Encapsulation	8
Encapsulation Sample Output	9
Enums Program	10
Enums Program Sample Output	11
Abstract Program	12
Abstract Program Sample Output	13

```
import java.util.*;
public class array {
    Run | Debug
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);

        int arr[] = new int[10];
        for (int i= 0; i <= 9;i++){
            System.out.print(s:"Enter a number :");
            arr[i] = scan.nextInt();
        }
        System.out.println(x:"The numbers that you inputted are:" );
        for (int i = 0; i<= 9; i++){
            System.out.print(arr[i] + " ");
        }
    }
}
```

Figure 1. Java Array Code

```
Enter a number :1
Enter a number :2
Enter a number :3
Enter a number :4
Enter a number :5
Enter a number :6
Enter a number :7
Enter a number :8
Enter a number :9
Enter a number :10
The numbers that you inputted are:
1 2 3 4 5 6 7 8 9 10 Press any key to continue . . .
```

Figure 1.1. Java Array Code Sample Output

```

import java.util.*;
public class car {
    String brand = "Lamborghini";
    String type = "Sedan";
    String color = "Navy Blue";
    Run | Debug
    public static void main(String[] args) {
        car myObj = new car();
        Scanner scan = new Scanner(System.in);
        System.out.println(x:"Default Value ");
        System.out.println(x:"-----");
        System.out.println("Brand: " + myObj.brand);
        System.out.println("Type: " + myObj.type);
        System.out.println("Color: " + myObj.color);
        System.out.println(x:"-----");

        System.out.print(s:"Enter New Brand: ");
        myObj.brand = scan.nextLine();
        System.out.print(s:"Enter New Type: ");
        myObj.type = scan.nextLine();
        System.out.print(s:"Enter New Color: ");
        myObj.color = scan.nextLine();
        System.out.println(x:"-----");
        System.out.println(x:"New Car Attributes ");
        System.out.println(x:"-----");
        System.out.println("Brand: " + myObj.brand);
        System.out.println("Type: " + myObj.type);
        System.out.println("Color: " + myObj.color);
    }
}

```

Figure 2. Car Modification

Default Value

Brand: Lamborghini

Type: Sedan

Color: Navy Blue

Enter New Brand: Toyota

Enter New Type: SUV

Enter New Color: Red

New Car Attributes

Brand: Toyota

Type: SUV

Color: Red

Figure 2.1. Car Modification Sample Output

```

import java.util.*;
public class carperformance {
    Scanner scan = new Scanner(System.in);

    public void fullThrottle() {
        String model = "";
        double distance;
        double time;
        System.out.println(x:"Enter new Model");
        model = scan.nextLine();
        System.out.println(x:"Enter Distance traveled");
        distance = scan.nextInt();
        System.out.println(x:"Enter time traveled");
        time = scan.nextInt();

        double KPH = distance/time ;
        double kmph = KPH * 60;
        System.out.println(x:"-----");
        System.out.println("Model:\t\t\t\t" + model);
        System.out.println("Distance Traveled:\t\t" + distance);
        System.out.println("Time Traveled:\t\t\t" + time);
        System.out.println("Speed:\t\t\t\t" + kmph + "kph");
        System.out.println(x:"");
        if (kmph <= 60) {
            System.out.println(x:"Your Car Is Too Slow");
        } else if ((kmph >= 61 )&&(kmph<=120)){
            System.out.println(x:"Your Car Is Awesome");
        }else{
            System.out.println(x:"Your Car Is Too Fast");
        }
    }
}
Run | Debug
public static void main(String[] args) {
    carperformance myCar = new carperformance();
    // Create a myCar object
    String model = "Honda Civic";
    String dis = "120km";
    String trav = "45 minutes";
    System.out.println(x:"Sample User Input: ");
    System.out.println(x:"-----");
    System.out.println("Car Model: "+ model);
    System.out.println("track Distance : "+ dis);
    System.out.println("time Traveled: "+ trav);
    System.out.println(x:"-----");
    myCar.fullThrottle();
}

```

Figure 3. Car Performance

```
Sample User Input:
-----
Car Model: Honda Civic
track Distance : 120km
time Traveled: 45 minutes
-----
Enter new Model
Lamborghini
Enter Distance traveled
160
Enter time traveled
60
-----
Model:                Lamborghini
Distance Traveled:    160.0
Time Traveled:        60.0
Speed:                160.0kph

Your Car Is Too Fast
```

Figure 3.1. Car Performance Sample Output


```

public class encapas {
    private String name;
    private int age;
    private int grade;

    public String getName() {
        return name;
    }
    public int getAge() {
        return age;
    }
    public int getGrades() {
        return grade;
    }
    public void setName(String newName) {
        this.name = newName;
    }
    public void setAge(int newAge) {
        this.age = newAge;
    }
    public void setGrade(int newGrades) {
        this.grade = newGrades;
    }
}

Run | Debug

public static void main(String[] args) {
    encapass students = new encapass();

    students.setName(newName:"Stephen");
    students.setAge(newAge:19);
    students.setGrade(newGrades:85);

    System.out.println("Name: " + students.getName());
    System.out.println("Age: " + students.getAge());
    System.out.println("Grade: " + students.getGrades());
}
}

```

Figure 4. Encapsulation

```
Name: Stephen
Age: 19
Grade: 85
Press any key to continue . . .
```

Figure 4.1 Encapsulation Sample Output

```
enum Level {  
    RED,  
    YELLOW,  
    GREEN  
}  
  
public class menu {  
    Run | Debug  
    public static void main(String[] args) {  
        Level myVar = Level.GREEN;  
  
        switch (myVar) {  
            case RED:  
                System.out.println(x:"The light is red stop!");  
                break;  
            case YELLOW:  
                System.out.println(x:"The light is yellow prepare to stop.");  
                break;  
            case GREEN:  
                System.out.println(x:"The light is green go!");  
                break;  
        }  
    }  
}
```

Figure 5. Enums

```
PS C:\Users\User\Documents\JAVA OOP> & 'C:\Program Files\Eclipse
es' '-cp' 'C:\Users\User\AppData\Roaming\Code\User\workspaceStora
```

The light is green go!

```
PS C:\Users\User\Documents\JAVA OOP> |
```

Figure 5.1. Enums Sample Output

```

import java.util.Scanner;
abstract class Vehicle {
    public abstract void carSound();
    public abstract String getCarModel();
}
class Toyota extends Vehicle {
    public void carSound() {
        System.out.println("Vroom! Vroom!");
    }
    public String getCarModel() {
        return "Toyota";
    }
}
class Honda extends Vehicle {
    public void carSound() {
        System.out.println("Hooooonnddaaaaaa!");
    }
    public String getCarModel() {
        return "Honda";
    }
}
class Suzuki extends Vehicle {
    public void carSound() {
        System.out.println("Suuuuzzzuuuuki!!!!!!");
    }
    public String getCarModel() {
        return "Suzuki";
    }
}

```

```

class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Select Car:");
        System.out.println("- Toyota");
        System.out.println("- Honda");
        System.out.println("- Suzuki");
        System.out.print("Enter Car Model:");
        String carChoice = scanner.nextLine();
        Vehicle myCar = null;
        if (carChoice.equalsIgnoreCase("toyota")) {
            myCar = new Toyota();
        } else if (carChoice.equalsIgnoreCase("honda")) {
            myCar = new Honda();
        } else if (carChoice.equalsIgnoreCase("suzuki")) {
            myCar = new Suzuki();
        }

        if (myCar != null) {
            System.out.println("Car model: " + myCar.getCarModel());
            System.out.print("Car sounds:");
            myCar.carSound();
        }
    }
}

```

Figure 6. Abstract Program

```
Select Car:  
  
- Toyota  
- Honda  
- Suzuki  
  
Enter Car Model:Toyota  
  
Car model: Toyota  
  
Car sounds:Vroom! Vroom!  
  
PS C:\Users\User\Documents\JAVA OOP> |
```

Figure 6.1. Abstract Program Sample Output