

**JAVA Assignment - 2**

**Name - Prateek Rai**

**Branch - Cse-ai - C**

**Roll no - 2400291520178**

**University roll no : 202401100300179**

```
import java.util.*;

@FunctionalInterface
interface Operation {
    int calculate(int n);
}

public class FunctionalExample {
    public static int square(int n) {
        return n * n;
    }

    public static void main(String[] args) {
        int n = 5;

        Operation anonymous = new Operation() {
            public int calculate(int n) {
                return n * n;
            }
        };

        System.out.println("Square using Anonymous Inner Class: " + anonymous.calculate(n));
    }
}
```

```
Operation lambda = (num) -> num * num;
System.out.println("Square using Lambda Expression: " + lambda.calculate(n));
Operation methodRef = FunctionalExample::square;
System.out.println("Square using Method Reference: " + methodRef.calculate(n));
}
}
```

```
import java.util.*;
import java.util.stream.*;

public class EvenNumbers {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(3, 8, 12, 5, 7, 10, 15);

        System.out.println("Even numbers:");
        numbers.stream()
            .filter(n -> n % 2 == 0)
            .forEach(System.out::println);
    }
}
```

```
import java.util.*;
import java.util.stream.*;

public class SquareAndFilter {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(4, 6, 8, 3, 10);

        System.out.println("Squares greater than 50:");
        numbers.stream()
            .map(n -> n * n)      // square each number
            .filter(n -> n > 50)   // keep only those > 50
            .forEach(System.out::println);
    }
}
```

```
import java.util.*;
import java.util.stream.*;

public class NamesStartingWithA {
    public static void main(String[] args) {
        List<String> names = Arrays.asList("Amit", "Rahul", "Anita", "Sneha", "Arjun");

        System.out.println("Names starting with 'A':");
        names.stream()
            .filter(name -> name.startsWith("A"))
            .forEach(System.out::println);
    }
}
```

```
import java.util.*;
import java.util.stream.*;

public class FindMaximum {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(25, 78, 12, 56, 89, 45);

        int max = numbers.stream()
            .max(Integer::compare)
            .orElseThrow();

        System.out.println("Maximum number: " + max);
    }
}
```

```
import java.util.*;
import java.util.stream.*;

public class UppercaseAndSort {
    public static void main(String[] args) {
        List<String> fruits = Arrays.asList("banana", "apple", "mango", "cherry");

        System.out.println("Uppercase and sorted fruits:");
        fruits.stream()
            .map(String::toUpperCase)
            .sorted()
            .forEach(System.out::println);
    }
}
```

```
import java.util.Scanner;

public class DayName {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter day number (1-7): ");
        int day = sc.nextInt();

        String dayName = switch (day) {
            case 1 -> "Monday";
            case 2 -> "Tuesday";
            case 3 -> "Wednesday";
            case 4 -> "Thursday";
            case 5 -> "Friday";
            case 6 -> "Saturday";
            case 7 -> "Sunday";
            default -> "Invalid day number";
        };

        System.out.println("Day: " + dayName);
        sc.close();
    }
}
```

```
import java.util.Scanner;

public class MonthDays {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter month number (1-12): ");
        int month = sc.nextInt();

        int days = switch (month) {
            case 1, 3, 5, 7, 8, 10, 12 -> 31;
            case 4, 6, 9, 11 -> 30;
            case 2 -> 28; // February (normal year)
            default -> 0;
        };

        if (days == 0)
            System.out.println("Invalid month number!");
        else
            System.out.println("Number of days: " + days);

        sc.close();
    }
}
```

```
import java.util.Scanner;

public class MonthDaysLeap {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter month number (1-12): ");
        int month = sc.nextInt();

        System.out.print("Enter year: ");
        int year = sc.nextInt();

        boolean isLeap = (year % 400 == 0) || (year % 4 == 0 && year % 100 != 0);

        int days = switch (month) {
            case 1, 3, 5, 7, 8, 10, 12 -> 31;
            case 4, 6, 9, 11 -> 30;
            case 2 -> isLeap ? 29 : 28;
            default -> 0;
        };

        if (days == 0)
            System.out.println("Invalid month number!");
        else
            System.out.println("Number of days in month: " + days);

        sc.close();
    }
}
```

```
import java.util.*;  
  
record Book(String title, double price, String author) {}  
  
public class BookTest {  
    public static void main(String[] args) {  
        Book[] books = {  
            new Book("Java Basics", 450, "John Doe"),  
            new Book("Advanced Java", 600, "Jane Smith"),  
            new Book("Data Structures", 750, "Mark Lee")  
        };  
  
        System.out.println("Books with price > 500:");  
        for (Book b : books) {  
            if (b.price() > 500) {  
                System.out.println(b.title() + " by " + b.author() + " (₹" + b.price() + ")");  
            }  
        }  
    }  
}
```

```
import java.util.*;  
  
sealed class Shape permits Circle, Rectangle, Triangle {  
    void area() {  
        System.out.println("Area not defined");  
    }  
}
```

```
final class Circle extends Shape {  
    double radius;  
    Circle(double r) { radius = r; }  
    void area() {  
        double a = Math.PI * radius * radius;  
        System.out.println("Circle area: " + a);  
    }  
}
```

```
final class Rectangle extends Shape {  
    double length, width;  
    Rectangle(double l, double w) { length = l; width = w; }  
    void area() {  
        double a = length * width;  
        System.out.println("Rectangle area: " + a);  
    }  
}
```

```
final class Triangle extends Shape {  
    double base, height;  
    Triangle(double b, double h) { base = b; height = h; }  
    void area() {  
        double a = 0.5 * base * height;  
        System.out.println("Triangle area: " + a);  
    }  
}
```

```
}
```

```
public class ShapeTest {  
    public static void main(String[] args) {  
        Shape s1 = new Circle(5);  
        Shape s2 = new Rectangle(4, 6);  
        Shape s3 = new Triangle(3, 7);  
  
        s1.area();  
        s2.area();  
        s3.area();  
    }  
}
```

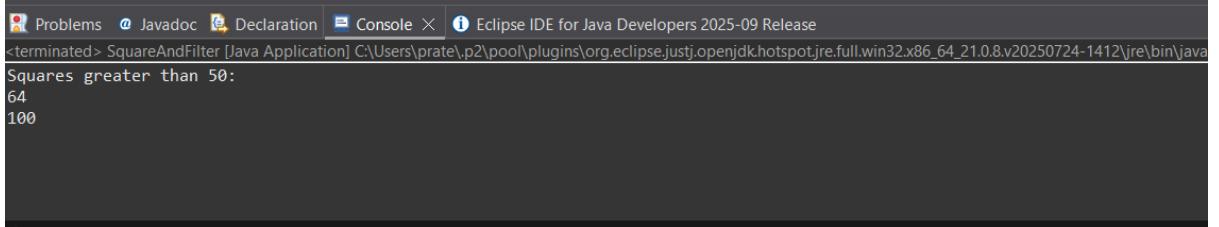
```
1 package first;
2
3 @FunctionalInterface
4 interface Operation {
5     int calculate(int n);
6 }
7
8 public class FunctionalExample {
9     public static int square(int n) {
10         return n * n;
11     }
12
13     public static void main(String[] args) {
14         int n = 5;
15
16         Operation anonymous = new Operation() {
17             public int calculate(int n) {
18                 return n * n;
19             }
20         };
21         System.out.println("Square using Anonymous Inner Class: " + anonymous.calculate(n));
22
23         Operation lambda = (num) -> num * num;
24         System.out.println("Square using Lambda Expression: " + lambda.calculate(n));
25         Operation methodRef = FunctionalExample::square;
26         System.out.println("Square using Method Reference: " + methodRef.calculate(n));
27     }
28 }
29
30
31
```

Problems Javadoc Declaration Console Eclipse IDE for Java Developers 2025-09 Release  
<terminated> FunctionalExample [Java Application] C:\Users\prate\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86\_64\_21.0.8.v20250724-1412\jre\bin\javaw.exe (04-Nov-2025, 10:59:17am - 10:59:18am elapsed: 0:00:01.160) [pid: 10652]  
Square using Anonymous Inner Class: 25  
Square using Lambda Expression: 25  
Square using Method Reference: 25

```
1 package first;
2
3 import java.util.*;
4 import java.util.stream.*;
5
6 public class EvenNumber {
7     public static void main(String[] args) {
8         List<Integer> numbers = Arrays.asList(3, 8, 12, 5, 7, 10, 15);
9
10        System.out.println("Even numbers:");
11        numbers.stream()
12            .filter(n -> n % 2 == 0)
13            .forEach(System.out::println);
14    }
15 }
```

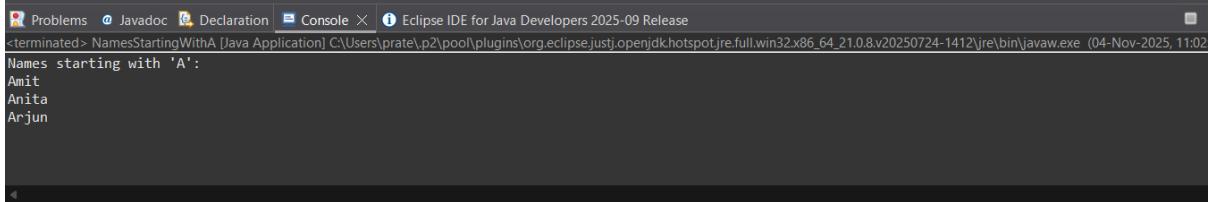
Problems Javadoc Declaration Console Eclipse IDE for Java Developers 2025-09 Release  
<terminated> EvenNumber [Java Application] C:\Users\prate\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86\_64\_21.0.8.v20250724-1412\jre\bin\javaw.exe (04-Nov-2025, 10:59:17am - 10:59:18am elapsed: 0:00:01.160) [pid: 10652]  
Even numbers:  
8  
12  
10

```
1 package first;
2
3 import java.util.*;
4 import java.util.stream.*;
5
6 public class SquareAndFilter {
7     public static void main(String[] args) {
8         List<Integer> numbers = Arrays.asList(4, 6, 8, 3, 10);
9
10        System.out.println("Squares greater than 50:");
11        numbers.stream()
12            .map(n -> n * n)          // square each number
13            .filter(n -> n > 50)      // keep only those > 50
14            .forEach(System.out::println);
15    }
16 }
17
18
19
```



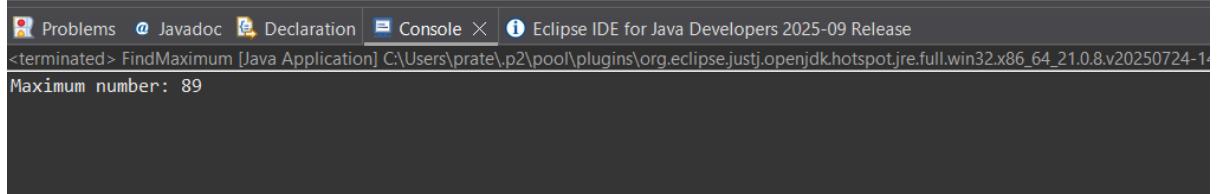
Problems Javadoc Declaration Console × Eclipse IDE for Java Developers 2025-09 Release  
<terminated> SquareAndFilter [Java Application] C:\Users\prate\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86\_64\_21.0.8.v20250724-1412\jre\bin\java  
Squares greater than 50:  
64  
100

```
1 package first;
2
3 import java.util.*;
4 import java.util.stream.*;
5
6 public class NamesStartingWithA {
7     public static void main(String[] args) {
8         List<String> names = Arrays.asList("Amit", "Rahul", "Anita", "Sneha", "Arjun");
9
10        System.out.println("Names starting with 'A':");
11        names.stream()
12            .filter(name -> name.startsWith("A"))
13            .forEach(System.out::println);
14    }
15 }
16
17
```

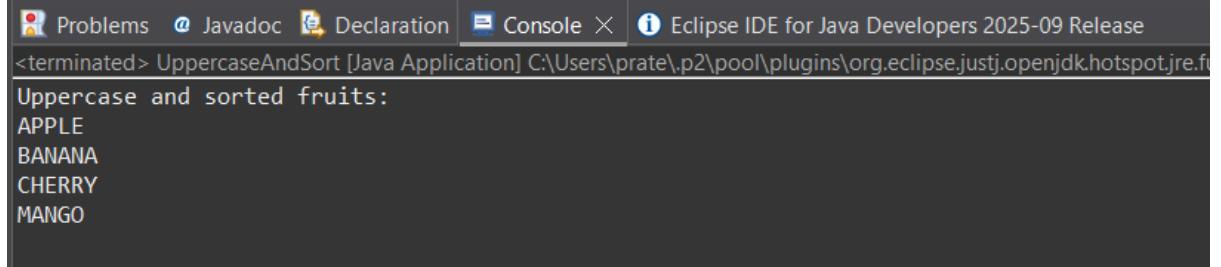


Problems Javadoc Declaration Console × Eclipse IDE for Java Developers 2025-09 Release  
<terminated> NamesStartingWithA [Java Application] C:\Users\prate\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86\_64\_21.0.8.v20250724-1412\jre\bin\javaw.exe (04-Nov-2025, 11:02)  
Names starting with 'A':  
Amit  
Anita  
Arjun

```
1 package first;
2
3 import java.util.*;
4 import java.util.stream.*;
5
6 public class FindMaximum {
7     public static void main(String[] args) {
8         List<Integer> numbers = Arrays.asList(25, 78, 12, 56, 89, 45);
9
10        int max = numbers.stream()
11                .max(Integer::compare)
12                .orElseThrow();
13
14        System.out.println("Maximum number: " + max);
15    }
16 }
17
```



```
1 package first;
2
3 import java.util.*;
4 import java.util.stream.*;
5
6 public class UppercaseAndSort {
7     public static void main(String[] args) {
8         List<String> fruits = Arrays.asList("banana", "apple", "mango", "cherry");
9
10        System.out.println("Uppercase and sorted fruits:");
11        fruits.stream()
12            .map(String::toUpperCase)
13            .sorted()
14            .forEach(System.out::println);
15    }
16 }
17
18
```

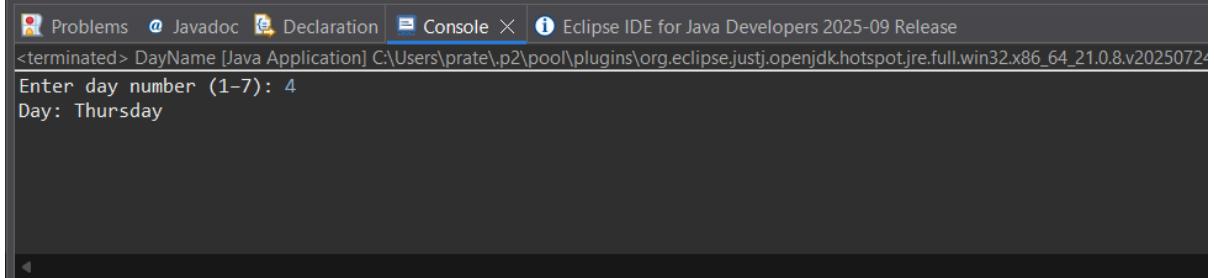


The screenshot shows the Eclipse IDE interface with the following details:

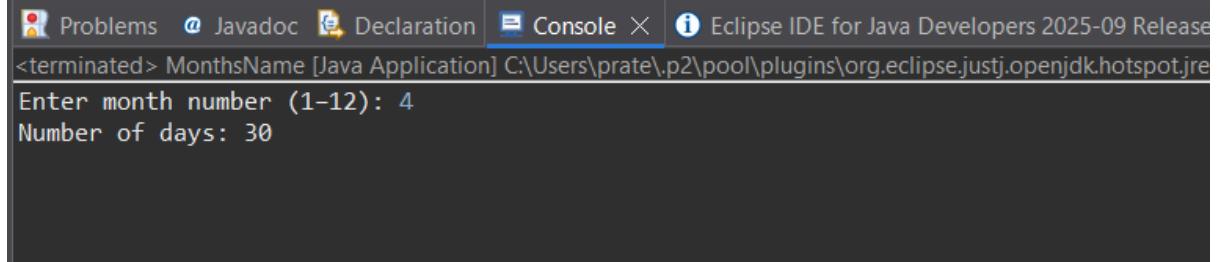
- Top Bar:** Problems, Javadoc, Declaration, Console, Eclipse IDE for Java Developers 2025-09 Release.
- Console Tab:** Shows the output of a Java application named "UppercaseAndSort".
- Output:**

```
<terminated> UppercaseAndSort [Java Application] C:\Users\prate\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.f
Uppercase and sorted fruits:
APPLE
BANANA
CHERRY
MANGO
```

```
1 package first;
2
3 import java.util.Scanner;
4
5 public class DayName {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         System.out.print("Enter day number (1-7): ");
9         int day = sc.nextInt();
10
11     String dayName = switch (day) {
12         case 1 -> "Monday";
13         case 2 -> "Tuesday";
14         case 3 -> "Wednesday";
15         case 4 -> "Thursday";
16         case 5 -> "Friday";
17         case 6 -> "Saturday";
18         case 7 -> "Sunday";
19         default -> "Invalid day number";
20     };
21
22     System.out.println("Day: " + dayName);
23     sc.close();
24 }
25
26
27
28
```



```
1 package first;
2
3 import java.util.Scanner;
4
5 public class MonthsName {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         System.out.print("Enter month number (1-12): ");
9         int month = sc.nextInt();
10
11     int days = switch (month) {
12         case 1, 3, 5, 7, 8, 10, 12 -> 31;
13         case 4, 6, 9, 11 -> 30;
14         case 2 -> 28; // February (normal year)
15         default -> 0;
16     };
17
18     if (days == 0)
19         System.out.println("Invalid month number!");
20     else
21         System.out.println("Number of days: " + days);
22
23     sc.close();
24 }
25
26 }
```



The screenshot shows the Eclipse IDE interface with the following details:

- Top Bar:** Problems, Javadoc, Declaration, Console (selected), Eclipse IDE for Java Developers 2025-09 Release.
- Console Tab:** Shows the application's terminal output.
- Output:**
  - Path: <terminated> MonthsName [Java Application] C:\Users\prate\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre
  - Input: Enter month number (1-12): 4
  - Output: Number of days: 30

```
15     }
16 }
17
18 final class Rectangle extends Shape {
19     double length, width;
20     Rectangle(double l, double w) { length = l; width = w; }
21     void area() {
22         double a = length * width;
23         System.out.println("Rectangle area: " + a);
24     }
25 }
26
27 final class Triangle extends Shape {
28     double base, height;
29     Triangle(double b, double h) { base = b; height = h; }
30     void area() {
31         double a = 0.5 * base * height;
32         System.out.println("Triangle area: " + a);
33     }
34 }
35
36 public class ShapeTest {
37     public static void main(String[] args) {
38         Shape s1 = new Circle(5);
39         Shape s2 = new Rectangle(4, 6);
40         Shape s3 = new Triangle(3, 7);
41
42         s1.area();
43         s2.area();
44         s3.area();
45     }
46 }
47
48
```

Problems Javadoc Declaration Console Eclipse IDE for Java Developers 2025-09 Release  
<terminated> ShapeTest [Java Application] C:\Users\prate\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86\_64\_21.0.8.v20250724-1412\jre\bin\javaw.exe (04-Nov-2025)  
Circle area: 78.53981633974483  
Rectangle area: 24.0  
Triangle area: 10.5

```
3 import java.util.Scanner;
4
5 public class MonthDaysLeap {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         System.out.print("Enter month number (1-12): ");
9         int month = sc.nextInt();
10
11        System.out.print("Enter year: ");
12        int year = sc.nextInt();
13
14        boolean isLeap = (year % 400 == 0) || (year % 4 == 0 && year % 100 != 0);
15
16        int days = switch (month) {
17            case 1, 3, 5, 7, 8, 10, 12 -> 31;
18            case 4, 6, 9, 11 -> 30;
19            case 2 -> isLeap ? 29 : 28;
20            default -> 0;
21        };
22
23        if (days == 0)
24            System.out.println("Invalid month number!");
25        else
26            System.out.println("Number of days in month: " + days);
27
28        sc.close();
29    }
30 }
31
32
33
```

```
Problems Javadoc Declaration Console × Eclipse IDE for Java Developers 2025-09 Release
terminated> MonthDaysLeap [Java Application] C:\Users\prate\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.f
Enter month number (1-12): 4
Enter year: 2025
Number of days in month: 30
```