

CDAC-Mumbai

Subject: Pattern Challenge in Java

Date: 04-09-2024 Total Marks:10

Time duration: 40 Min. Batch: August-24

Q1. Write a Java program to print a hollow pyramid pattern using asterisks (*) and spaces.

```
import java.util.Scanner;
```

```
public class Code1 {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        int n = sc.nextInt();  
  
        for (int i = 1; i <= n; i++) {  
            // Print leading spaces  
            for (int k=1;k<=n-i;k++) {  
                System.out.print(" ");  
            }  
            // Print stars and spaces in between  
            for (int j = 1; j <= 2 * i - 1; j++) {  
                if (j == 1 || j == 2 * i - 1 || i == n) {  
                    System.out.print("*");  
                } else {  
                    System.out.print(" ");  
                }  
            }  
            System.out.println();  
        }  
    }  
}
```

```

    }
}
}

```

```

J Code1.java > Code1 > main(String[])
1  import java.util.Scanner;
2
3  public class Code1 {
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner sc = new Scanner(System.in);
7          int n = sc.nextInt();
8          for (int i = 1; i <= n; i++) {
9              // Print leading spaces
10             for (int k=1;k<=n-i;k++) {
11                 System.out.print(s:" ");
12             }
13             // Print stars and spaces in between
14             for (int j = 1; j <= 2 * i - 1; j++) {
15                 if (j == 1 || j == 2 * i - 1 || i == n)
16                     System.out.print("*");
17                 else
18                     System.out.print(" ");
19             }
20             System.out.println();
21         }
22     }
23 }

```

PROBLEMS 7 OUTPUT TERMINAL PORTS DEBUG CONSOLE

```

}
5
*
* *
*  *
*   *
*****
PS C:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2> cd "c:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2\" ; if ($?) { javac Code1.java } ; if ($?) { java Code1
}
5
*
* *
*  *
*   *
*****
PS C:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2>

```

Launchpad Run Testcases 0 7 0 Live Share Java: Ready Java Go Live Prettier

Q2. Write a Java program to print a hollow inverted pyramid pattern using asterisks (*) and spaces.

```
import java.util.Scanner;

public class Code1 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int n = sc.nextInt();

        for (int i = n; i >= 1; i--) {

            // Print leading spaces

            for (int k=1;k<=n-i;k++) {

                System.out.print(" ");

            }

            // Print stars and spaces in between

            for (int j = 1; j <= 2 * i - 1; j++) {

                if (j == 1 || j == 2 * i - 1 || i == n) {

                    System.out.print("*");

                } else {

                    System.out.print(" ");

                }

            }

            System.out.println();

        }

    }

}
```

The screenshot shows a Java IDE with a file named `Code1.java` open. The code defines a `main` method that takes an array of strings `args`. It uses a `Scanner` to read an integer `n` from the command line. A `for` loop iterates from `i = n` down to `i = 1`. Inside this loop, there are two nested `for` loops: the first prints leading spaces, and the second prints stars. The stars are printed in a way that forms a pyramid shape. The output of the program is shown in the terminal, displaying a pyramid of stars for `n = 5`.

```
3 public class Code1 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         int n = sc.nextInt();
7
8         for (int i = n; i >= 1; i--) {
9             // Print leading spaces
10            for (int k=1;k<=n-i;k++) {
11                System.out.print(s:" ");
12            }
13            // Print stars and spaces in between
14            for (int j = 1; j <= 2 * i - 1; j++) {
15                if (j == 1 || j == 2 * i - 1 || i == n)
16                    System.out.print(s:"*");
17                } else {
18                    System.out.print(s:" ");
19                }
20            }
21        }
22    }
23 }
```

PS C:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2> cd "c:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2\" ; if (\$?) { javac Code1.java } ; if (\$?) { java Code1 }
5

* *
* *
* *

* *
* *
* *

* *
* *
* *

* *
* *
* *

Q3. Write a Java program to print a pyramid pattern of alphabets, starting from 'A', using spaces.

A

BCD

EFGHI

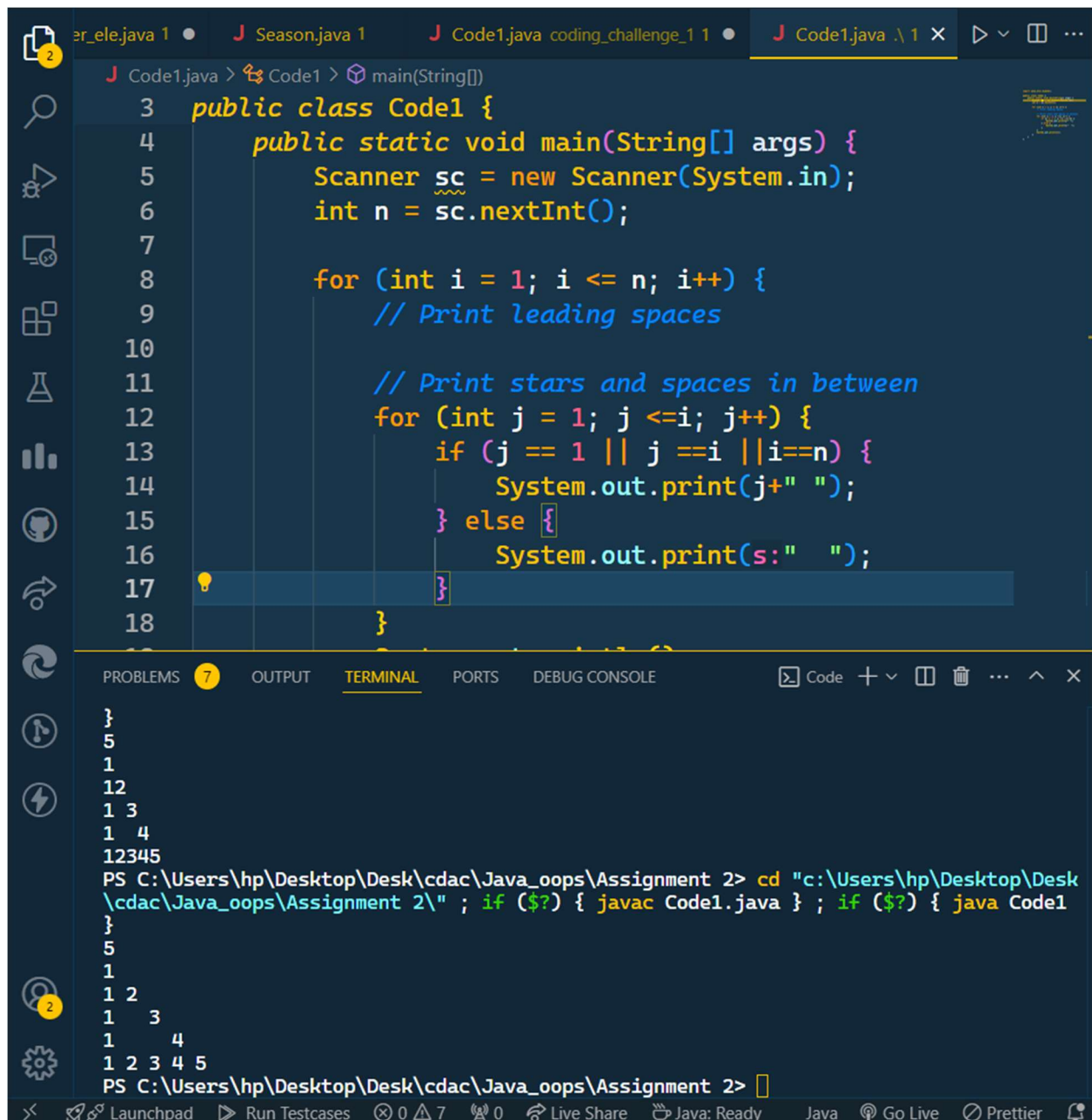
JKLMNOP

QRSTUVWXYZ

Q4. Write a Java program to print a hollow right triangle pattern using numbers and spaces.

```
import java.util.Scanner;
```

```
public class Code1 {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        int n = sc.nextInt();  
  
        for (int i = 1; i <= n; i++) {  
            // Print leading spaces  
  
            // Print stars and spaces in between  
            for (int j = 1; j <= i; j++) {  
                if (j == 1 || j == i || i == n) {  
                    System.out.print(j+" ");  
                } else {  
                    System.out.print(" ");  
                }  
            }  
            System.out.println();  
        }  
    }  
}
```



```
Code1.java > Code1 > main(String[])
3 public class Code1 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         int n = sc.nextInt();
7
8         for (int i = 1; i <= n; i++) {
9             // Print leading spaces
10
11             // Print stars and spaces in between
12             for (int j = 1; j <= i; j++) {
13                 if (j == 1 || j == i || i == n) {
14                     System.out.print(j + " ");
15                 } else {
16                     System.out.print(" ");
17                 }
18             }
19         }
20     }
21 }
```

PROBLEMS 7 OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
}
5
1
12
1 3
1 4
12345
PS C:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2> cd "c:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2\" ; if ($?) { javac Code1.java } ; if ($?) { java Code1
}
5
1
1 2
1 3
1 4
1 2 3 4 5
PS C:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2>
```

Launchpad Run Testcases 0 7 0 Live Share Java: Ready Java Go Live Prettier

Write a Java program to print a Palindrome Number Pyramid

```
import java.util.Scanner;
```

```
public class Code1 {
```

```
    public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);
```

```

int n = sc.nextInt();

for (int i = 1; i <= n; i++) {
    // Print leading spaces
    for (int k=1;k<=n-i;k++) {
        System.out.print(" ");
    }

    // Print stars and spaces in between
    for (int j = 1; j <=i; j++) {

        System.out.print(j);

    }
    for (int j = i - 1; j >= 1; j--) {
        System.out.print(j);
    }
    System.out.println();
}
}
}

```

The screenshot shows an IDE window titled "Assignment 2" with a Java file named "Code1.java". The code defines a class "Code1" with a "main" method. The "main" method contains a loop that prints the numbers 1 through 5. The output of the program is displayed in the "TERMINAL" pane, showing the numbers 1, 120, 12310, 1234210, and 123453210. The terminal also shows the command "cd \"c:\Users\hpl\Desktop\Desk\cdac\Java_oops\Assignment 2\" ; if (\$?) { javac Code1.java } ; if (\$?) { java Code1 }" and the output "1 120 12310 1234210 123453210".

```
3 public class Code1 {
4     public static void main(String[] args) {
18
19
20     }
21     for (int j = i - 1; j >= 1; j--) {
22         System.out.print(j);
23     }
24     System.out.println();
25 }
26 }
27 }
```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
}
5
1
120
12310
1234210
123453210
PS C:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2> cd "c:\Users\hpl\Desktop\Desk\cdac\Java_oops\Assignment 2\" ; if ($?) { javac Code1.java } ; if ($?) { java Code1 }
5
1
121
12321
123321
1234321
PS C:\Users\hp\Desktop\Desk\cdac\Java_oops\Assignment 2>
```

Launchpad Run Testcases 0 0 7 Live Share Java: Ready Java Go Live Prettier

28°C Partly sunny