Lesson Plan

Java Pattern-2

PW SKILLS

Pre-requisites:

- Java Syntax
- Loops

List of patterns involved:

1. Rhombus
2. Star Pyramid
3. Number Pyramid Palindrome
4. Star Diamond
5. Star Bridge
6. Number Bridge
7. Number Spiral

Q1) Print the given Pattern:

Rhombus

```
public class Main {
    public static void main(String[] args) {
        int rows = 4; // You can change this value to adjust
the number of rows

        for (int i = 1; i <= rows; i++) {
            // Print spaces
            for (int j = i; j < rows; j++) {
                System.out.print(" ");
```

```
            }

            // Print asterisks
            for (int k = 1; k <= rows; k++) {
                System.out.print("*");
            }

            System.out.println();
        }
    }
}
```

Q2) Print the given Pattern:

Star Pyramid

```
public class Main {
    public static void main(String[] args) {
        int rows = 4; // You can change this value to adjust
the number of rows

        for (int i = 1; i <= rows; i++) {
            // Print spaces
            for (int j = i; j < rows; j++) {
                System.out.print(" ");
            }

            // Print asterisks
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }
```

```
            System.out.println();
        }
    }
}
```

Q3) Print the given Pattern:

Number Pyramid Palindrome


```
public class Main {
    public static void main(String[] args) {
        int rows = 4; // You can change this value to adjust
the number of rows

        for (int i = 1; i <= rows; i++) {
            // Print spaces
            for (int j = i; j < rows; j++) {
                System.out.print("  ");
            }

            // Print increasing numbers
            for (int k = 1; k <= i; k++) {
                System.out.print(k + " ");
            }

            // Print decreasing numbers
            for (int l = i - 1; l >= 1; l--) {
                System.out.print(l + " ");
            }
```

```
            System.out.println();
        }
    }
}
```

Q4) Print the given Pattern:

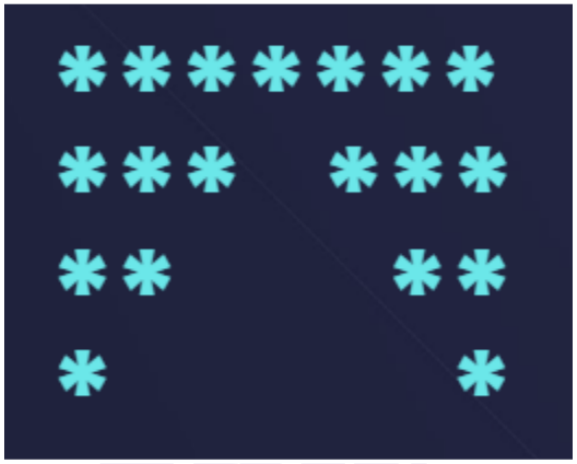
Star Diamond

```
public class Main {
    public static void main(String[] args) {
        int rows = 4; // You can change this value to adjust
the number of rows

        for (int i = 1; i <= rows; i++) {
            for (int j = i; j < rows; j++) {
                System.out.print(" ");
            }

            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }

            System.out.println();
        }

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

```
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }

            System.out.println();
        }
    }
}
```

Q5) Print the given Pattern:

Star Bridge

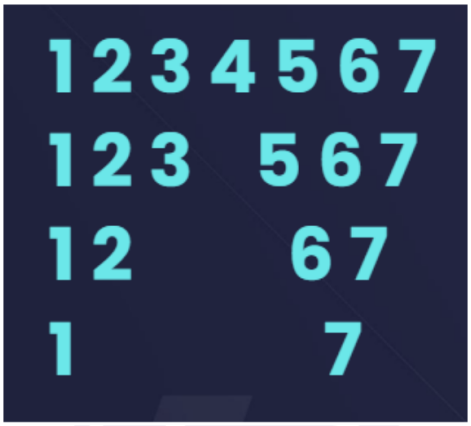


```
public class Main {
    public static void main(String[] args) {
        int rows = 4; // You can change this value to adjust
the number of rows
        int initialAsterisks = 7;

        // Upper part of the reversed pattern
        for (int i = rows; i >= 0; i--) {
            // Print stars on the left side
            for (int j = 0; j < i; j++) {
                System.out.print("*");
            }

            // Print spaces in the middle
            for (int k = 0; k < 2 * (rows - i) ; k++) {
                System.out.print(" ");
            }
```

```
            // Print stars on the right side
            for (int l = 0; l < i; l++) {
                System.out.print("*");
            }

            System.out.println();
        }
    }
}
```

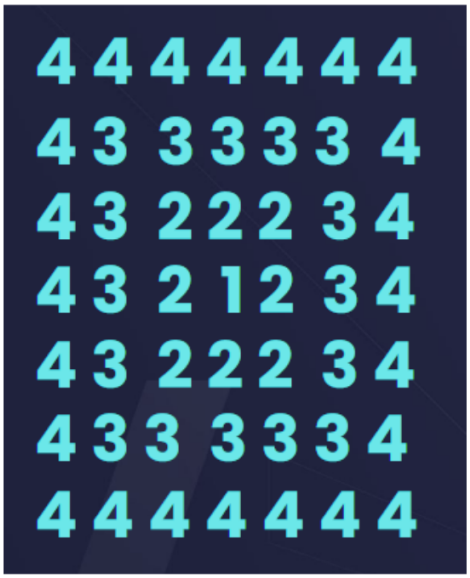
Q6) Print the given Pattern:

Number Bridge



Q7) Print the given Pattern:

Number Spiral



```
public class Main {
    public static void main(String[] args) {
        int n = 4;

        for (int i = n; i > 1; i--) {
            for (int j = n; j ⩾ 1; j--) {
                // prints top left of the pattern
                if (j > i)
                    System.out.print(j + " ");
                else
                    System.out.print(i + " ");
            }
            for (int j = 2; j ⩽ n; j++) {
                // prints top right of the pattern
                if (j > i)
                    System.out.print(j + " ");
                else
                    System.out.print(i + " ");
            }
            System.out.println();
        }

        for (int i = 1; i ⩽ n; i++) {
            for (int j = n; j ⩾ 1; j--) {
                // prints left bottom of the pattern
                if (j > i)
                    System.out.print(j + " ");
                else
                    System.out.print(i + " ");
            }
            for (int j = 2; j ⩽ n; j++) {
                // prints right bottom of the pattern
                if (j > i)
                    System.out.print(j + " ");
                else
                    System.out.print(i + " ");
            }
            System.out.println();
        }
    }
}
```

Upcoming Lecture:

Functions