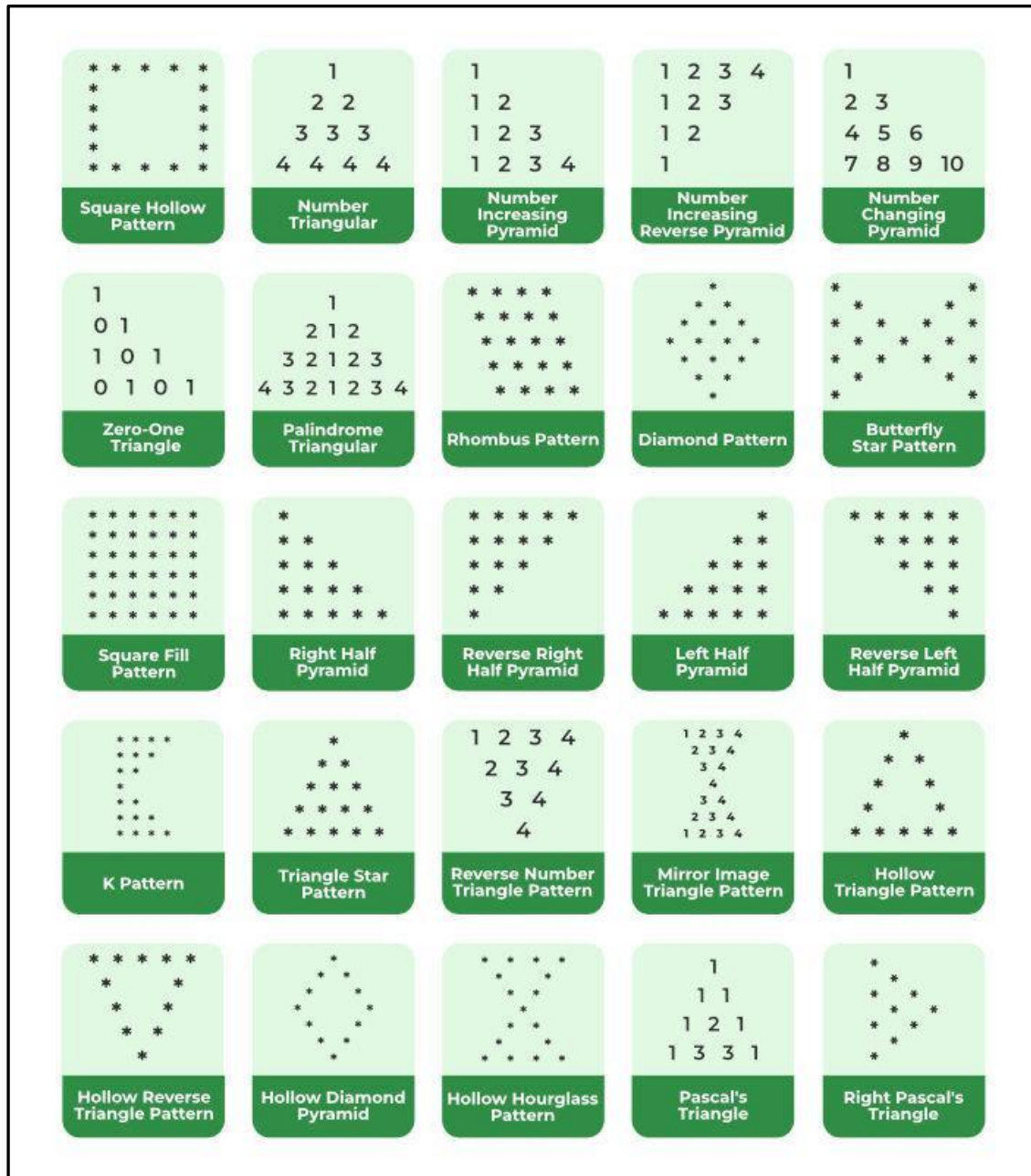


Java Pattern Programs



1. Square Hollow Pattern

```
public class SquareHollowPattern {
    public static void main(String[] args) {
        int n = 5; // Number of rows and columns
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= n; j++) {
                if (i == 1 || i == n || j == 1 || j == n)
                    System.out.print("* ");
                else
                    System.out.print(" ");
            }
        }
    }
}
```

```

        System.out.println();
    }
}
}

```

2. Number Triangular Pattern:

```

public class NumberTriangularPattern {
    public static void main(String[] args) {
        int n = 4;
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print(i + " ");
            }
            System.out.println();
        }
    }
}

```

3. Number Increasing Pyramid

```

public class NumberIncreasingPyramid {
    public static void main(String[] args) {
        int n = 4;
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print(j + " ");
            }
            System.out.println();
        }
    }
}

```

4. Number Increasing Reverse Pyramid

```

public class NumberIncreasingReversePyramid {
    public static void main(String[] args) {
        int n = 4;
        for (int i = n; i >= 1; i--) {
            for (int j = 1; j <= i; j++) {
                System.out.print(j + " ");
            }
        }
    }
}

```

```

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

```

5. Number Changing Pyramid

```

public class NumberChangingPyramid {
    public static void main(String[] args) {
        int n = 10;
        int number = 1;
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print(number + " ");
                number++;
            }
            System.out.println();
        }
    }
}

```

6. Zero-One Triangle

```

public class ZeroOneTriangle {
    public static void main(String[] args) {
        int n = 4;
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= i; j++) {
                if ((i + j) % 2 == 0)
                    System.out.print("1 ");
                else
                    System.out.print("0 ");
            }
            System.out.println();
        }
    }
}

```

7. Palindrome Triangular Pattern

```

public class PalindromeTriangularPattern {
    public static void main(String[] args) {
        int n = 4;
        for (int i = 1; i <= n; i++) {
            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();
        }
    }
}

```

8. Rhombus Pattern

```

public class RhombusPattern {
    public static void main(String[] args) {
        int n = 5;
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= n - i; j++) {
                System.out.print(" ");
            }
            for (int j = 1; j <= n; j++) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}

```

9. Diamond Pattern

```

public class DiamondPattern {
    public static void main(String[] args) {
        int n = 4;
        for (int i = 1; i <= n; i++) {
            for (int j = i; j < n; j++) {
                System.out.print(" ");
            }

```

```

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

```

10.Butterfly Star Pattern

```

public class ButterflyStarPattern {
    public static void main(String[] args) {
        int n = 4;
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print("* ");
            }
            for (int j = 1; j <= 2 * (n - i); j++) {
                System.out.print(" ");
            }
            for (int j = 1; j <= i; j++) {
                System.out.print("* ");
            }
            System.out.println();
        }
        for (int i = n; i >= 1; i--) {
            for (int j = 1; j <= i; j++) {
                System.out.print("* ");
            }
            for (int j = 1; j <= 2 * (n - i); j++) {

```

```

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

```

11. Square Fill Pattern

```

public class SquareFillPattern {
    public static void main(String[] args) {
        int n = 5; // Number of rows and columns
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= n; j++) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}

```

12. Right Half Pyramid

```

public class RightHalfPyramid {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}

```

13. Reverse Right Half Pyramid

```

public class ReverseRightHalfPyramid {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = n; i >= 1; i--) {
            for (int j = 1; j <= i; j++) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}

```

14. Left Half Pyramid

```

public class LeftHalfPyramid {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= n - i; j++) {
                System.out.print(" ");
            }
            for (int j = 1; j <= i; j++) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}

```

15. Reverse Left Half Pyramid

```

public class ReverseLeftHalfPyramid {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j < i; j++) {
                System.out.print(" ");
            }
            for (int j = i; j <= n; j++) {
                System.out.print("* ");
            }
        }
    }
}

```

```

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

```

16. K Pattern

```

public class KPattern {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = 0; i < n; i++) {
            System.out.print("*");
            for (int j = 0; j < n; j++) {
                if (i == j || i + j == n - 1) {
                    System.out.print("*");
                } else {
                    System.out.print(" ");
                }
            }
            System.out.println();
        }
    }
}

```

17. Triangle Star Pattern

```

public class TriangleStarPattern {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = 1; i <= n; i++) {
            for (int j = i; j < n; j++) {
                System.out.print(" ");
            }
            for (int j = 1; j <= (2 * i - 1); j++) {
                System.out.print("*");
            }
            System.out.println();
        }
    }
}

```

```
}
```

18. Reverse Number Triangle Pattern

```
public class ReverseNumberTrianglePattern {  
    public static void main(String[] args) {  
        int n = 4;  
        for (int i = n; i >= 1; i--) {  
            for (int j = 1; j <= i; j++) {  
                System.out.print(j + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```

19. Mirror Image Triangle Pattern

```
public class MirrorImageTrianglePattern {  
    public static void main(String[] args) {  
        int n = 4;  
        for (int i = 1; i <= n; i++) {  
            for (int j = 1; j <= n - i; j++) {  
                System.out.print(" ");  
            }  
            for (int j = 1; j <= i; j++) {  
                System.out.print(j + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```

20. Hollow Triangle Pattern

```
public class HollowTrianglePattern {  
    public static void main(String[] args) {  
        int n = 5; // Number of rows  
        for (int i = 1; i <= n; i++) {  
            for (int j = 1; j <= i; j++) {
```

```

        if (j == 1 || j == i || i == n) {
            System.out.print("* ");
        } else {
            System.out.print(" ");
        }
    }
    System.out.println();
}
}
}

```

21. Hollow Reverse Triangle Pattern

```

public class HollowReverseTrianglePattern {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = n; i >= 1; i--) {
            for (int j = 1; j <= i; j++) {
                if (i == n || j == 1 || j == i) {
                    System.out.print("* ");
                } else {
                    System.out.print(" ");
                }
            }
            System.out.println();
        }
    }
}

```

22. Hollow Diamond Pyramid

```

public class HollowDiamondPyramid {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = 1; i <= n; i++) {
            for (int j = i; j < n; j++) {
                System.out.print(" ");
            }
            for (int j = 1; j <= (2 * i - 1); j++) {
                if (j == 1 || j == (2 * i - 1)) {

```

```

        System.out.print("*");
    } else {
        System.out.print(" ");
    }
}
System.out.println();
}
for (int i = n - 1; i >= 1; i--) {
    for (int j = n; j > i; j--) {
        System.out.print(" ");
    }
    for (int j = 1; j <= (2 * i - 1); j++) {
        if (j == 1 || j == (2 * i - 1)) {
            System.out.print("*");
        } else {
            System.out.print(" ");
        }
    }
    System.out.println();
}
}
}

```

23. Hollow Hourglass Pattern

```

public class HollowHourglassPattern {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = n; i >= 1; i--) {
            for (int j = i; j < n; j++) {
                System.out.print(" ");
            }
            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();
        }
    }
}

```

```

    }
    for (int i = 2; i <= n; i++) {
        for (int j = i; j < n; j++) {
            System.out.print(" ");
        }
        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();
    }
}

```

24. Pascal's Triangle

```

public class PascalsTriangle {
    public static void main(String[] args) {
        int n = 5; // Number of rows
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n - i; j++) {
                System.out.print(" ");
            }
            int number = 1;
            for (int j = 0; j <= i; j++) {
                System.out.print(number + " ");
                number = number * (i - j) / (j + 1);
            }
            System.out.println();
        }
    }
}

```

25. Right Pascal's Triangle

```

public class RightPascalsTriangle {
    public static void main(String[] args) {

```

```
int n = 5; // Number of rows
for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= i; j++) {
        System.out.print("* ");
    }
    System.out.println();
}
for (int i = n - 1; i >= 1; i--) {
    for (int j = 1; j <= i; j++) {
        System.out.print("* ");
    }
    System.out.println();
}
}
```
