Basavaraj Aili

KodNest CSR June 2023

(29-07-2023)

Assignment: Pattern printing in JAVA

1) INCREASING TRAINGLE PATTERN

```
package patternProgramsPractice;

public class Pattern1 {
   //INCREASING TRIANGLE PATTERN
   public static void main(String[] args) {
        int n=5;
        for(int i=1;i<=n;i++)
        {
             for(int j=1;j<=i;j++)
             {
                  System.out.print("* ");
             }
             System.out.println();
        }
   }
}
```

Output:

```
*
* *
* * *
* * * *
* * * * *
```

2) DECREASING TRAINGLE PATTERN

```
package patternProgramsPractice;

public class Pattern2 {
	//DECREASING TRIANGLE PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=i;j<=n;j++)
			{
				System.out.print("* ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
* * * * *
* * * *
* * *
* *
*
```

3) RIGHT SIDED TRAINGLE (DECREASING SPACES)

```
package patternProgramsPractice;

public class Pattern3 {
     //RIGHT SIDED TRIANGLE (DECREASING SPACES)
     public static void main(String[] args) {
          int n=5;
          for(int i=1;i<=n;i++)
          {
               for(int j=i;j<=n;j++)
               {
                    System.out.print("  ");
               }
               for(int j=1;j<=i;j++)
               {
                    System.out.print("* ");
               }
               System.out.println();
          }
     }
}
```

Output:

```
            *
          * *
        * * *
      * * * *
    * * * * *
```

4) LEFT SIDED TRAINGLE (DECREASING SPACES)

```
package patternProgramsPractice;

public class Pattern4 {
	//LEFT SIDED TRIANGLE (DECREASING SPACES)
	public static void main(String[] args) {
		int n=5;
		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();
		}
	}
}
```

Output:

```
*
* *
* * *
* * * *
* * * * *
```

5) HILL PATTERN

```
package patternProgramsPractice;

public class Pattern5 {
	//HILL PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=i;j<=n;j++)
			{
				System.out.print("  ");
			}
			for(int j=1;j<i;j++)
			{
				System.out.print("* ");
			}
			for(int j=1;j<=i;j++)
			{
				System.out.print("* ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
        *
      * * *
    * * * * *
  * * * * * * *
* * * * * * * * *
```

6) REVERSE HILL PATTERN

```
package patternProgramsPractice;

public class Pattern6 {
     //REVERSE HILL PATTERN
     public static void main(String[] args) {
          int n=5;
          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("* ");
               }
               for(int j=i;j<=n;j++)
               {
                    System.out.print("* ");
               }
               System.out.println();
          }
     }
}
```

Output:

```
* * * * * * * * *
  * * * * * * *
    * * * * *
      * * *
        *
```

7) RIGHT SIDED TRAINGLE (INCREASING SPACES)

```
package patternProgramsPractice;

public class Pattern7 {
     //RIGHT SIDED TRIANGLE (INCREASING SPACES)
     public static void main(String[] args) {
          int n=5;
          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();
          }
     }
}
```

Output:

```
    * * * * *
      * * * *
        * * *
          * *
            *
```

8) DIAMOND PATTERN

```
package patternProgramsPractice;

public class Pattern8 {
	//DAIMOND PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<n;i++)
		{
			for(int j=i;j<=n;j++)
			{
				System.out.print("  ");
			}
			for(int j=1;j<i;j++)
			{
				System.out.print("* ");
			}
			for(int j=1;j<=i;j++)
			{
				System.out.print("* ");
			}
			System.out.println();
		}
		//REVERSE HILL
		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("* ");
			}
			for(int j=i;j<=n;j++)
			{
				System.out.print("* ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
            *
          * * *
        * * * * *
      * * * * * * *
    * * * * * * * * *
      * * * * * * *
        * * * * *
          * * *
            *
```

9) SQUARE PARALLEL BAR PATTERN

```
package patternProgramsPractice;

public class Pattern9 {
	//SQUARE PARALLEL BAR PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=n;j++)
			{
				if(j==1 || j==n)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
*         *
*         *
*         *
*         *
*         *
```

10) SQUARE PLUS PATTERN

```
package patternProgramsPractice;

public class Pattern10 {
	//SQUARE PLUS PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=n;j++)
			{
				if(i==n/2+1 || j==n/2+1)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
      *
      *
* * * * *
      *
      *
```

11) CROSS PATTERN

```
package patternProgramsPractice;

public class Pattern11 {
	//CROSS PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=n;j++)
			{
				if(i==j || i+j==n+1)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
*         *
  *     *
    *
  *     *
*         *
```

12) HOLLOW SQUARE PATTERN

```
package patternProgramsPractice;

public class Pattern12 {
	//HOLLOW SQUARE PATTERN
	public static void main(String[] args) {
		int n=5;
		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();
		}
	}
}
```

Output:

```
* * * * *
*       *
*       *
*       *
* * * * *
```

13) HOLLOW INCREASING TRAINGLE (LEFT SIDED)

```
package patternProgramsPractice;

public class Pattern13 {
	//HOLLOW INCREASING TRAINGLE (LEFT SIDED)
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=n;j++)
			{
				if(i==n || j==1 || i==j)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
*
* *
*   *
*     *
* * * * *
```

14) HOLLOW DECREASING TRAINGLE (LEFT SIDED)

```
package patternProgramsPractice;

public class Pattern14 {
	//HOLLOW DECREASING TRAINGLE (LEFT SIDED)
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=n;j++)
			{
				if(i==1 || j==1 || i+j==6)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
* * * * *
*       *
*     *
* *
*
```

15) HOLLOW INCREASING TRAINGLE (RIGHT SIDED)

```
package patternProgramsPractice;

public class Pattern15 {
	//HOLLOW INCREASING TRAINGLE (RIGHT SIDED)
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=n;j++)
			{
				if(i==n || j==n || i+j==6)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
        *
      * *
    *   *
  *     *
* * * * *
```

16) HOLLOW HILL PATTERN

```
package patternProgramsPractice;

public class Pattern16 {
	//HOLLOW HILL PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=i;j<=n;j++)
			{
				System.out.print("  ");
			}
			for(int j=1;j<i;j++)
			{
				if(i==n || j==1)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			for(int j=1;j<=i;j++)
			{
				if(i==n || i==j)
				System.out.print("* ");
				else
					System.out.print("  ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
          *
        *   *
      *       *
    *           *
  * * * * * * * * *
```

17) HOLLOW DIAMOND PATTERN

```
package patternProgramsPractice;

public class Pattern17 {
	//HOLLOW DIAMOND PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<n;i++)
		{
			for(int j=i;j<=n;j++)
			{
				System.out.print("  ");
			}
			for(int j=1;j<i;j++)
			{
				if(j==1)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			for(int j=1;j<=i;j++)
			{
				if(j==i)
					System.out.print("* ");
				else
					System.out.print("  ");
			}
			System.out.println();
		}
		//REVERSE HILL
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=i;j++)
			{
				System.out.print("  ");
			}
			for(int j=i;j<n;j++)
			{
				if(j==i)
					System.out.print("* ");
				else
```

```
                    System.out.print("  ");
                }
                for(int j=i;j<=n;j++)
                {
                    if(j==n)
                        System.out.print("* ");
                    else
                        System.out.print("  ");
                }
                System.out.println();
            }
        }
}
```

Output:

```
            *
          *   *
        *       *
      *           *
    *               *
      *           *
        *       *
          *   *
            *
```

18) NUMBER TRAINGLE PATTERN

```
package patternProgramsPractice;

public class Pattern18 {
	//NUMBER TRIANGLE PATTERN
	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<=i;j++)
			{
				System.out.print(i+" ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
    1
   2 2
  3 3 3
 4 4 4 4
5 5 5 5 5
```

19) NUMBER INCREASING PYRAMID PATTERN

```
package patternProgramsPractice;

public class Pattern19 {
	//NUMBER INCREASING PYRAMID PATTERN
	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=i;j++)
			{
				System.out.print(j+" ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

20) NUMBER CHANGING PYRAMID PATTERN

```
package patternProgramsPractice;

public class Pattern20 {
	//NUMBER CHANGING PYRAMID PATTERN
	public static void main(String[] args) {
		int n=5,num=1;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=i;j++)
			{
				System.out.print(num+" ");
				num++;
			}
			System.out.println();
		}
	}
}
```

Output:

```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

21) 1 OR 0 PATTERN

```
package patternProgramsPractice;

public class Pattern21 {
	//1 or 0 TRIANGLE PATTERN
	public static void main(String[] args) {
		int n=5;
		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();
		}
	}
}
```

Output:

```
1
0 1
1 0 1
0 1 0 1
1 0 1 0 1
```

22) BUTTERFLY PATTERN

```
package patternProgramsPractice;

public class Pattern22 {
     //BUTTERFLY PATTERN
     public static void main(String[] args) {
          int n=5,i,j;
          // outer loop to handle upper part
          for (i=1;i<=n;i++) {
               // inner loop to print stars
               for (j=1;j<=i;j++) {
                    System.out.print("* ");
               }

               // inner loop to print spaces
               int spaces = 2*(n-i);
               for (j=1;j<=spaces;j++) {
                    System.out.print("  ");
               }

               // inner loop to print stars
               for (j=1;j<=i;j++) {
                    System.out.print("* ");
               }

               System.out.println();
          }

          // outer loop to handle lower part
          for (i=n;i>=1;i--) {
               // inner loop to print stars
               for (j=1;j<=i;j++) {
                    System.out.print("* ");
               }

               // inner loop to print spaces
               int spaces = 2*(n - i);
               for (j=1;j<=spaces;j++) {
                    System.out.print("  ");
               }

               // inner loop to print stars
               for (j=1;j<=i;j++) {
```

```
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}
```

Output:

```
*                   *
* *                 * *
* * *               * * *
* * * *           * * * *
* * * * * * * * * * * * *
* * * * * * * * * * * * *
* * * *           * * * *
* * *               * * *
* *                 * *
*                   *
```

23) K PATTERN

```
package patternProgramsPractice;

public class Pattern23 {
	//K PATTERN
	public static void main(String[] args) {
		int n=5,i,j;
		// outer loop to handle rows
		for(i=n;i>=1;i--) {

			// inner loop to handle columns
			for(j=1;j<=i;j++) {
				System.out.print("* ");
			}
			// printing new line for each row
			System.out.println();
		}

		// outer loop to handle rows
		for(i=2;i<=n;i++) {

			// inner loop to handle columns
			for(j=1;j<=i;j++) {
				System.out.print("* ");
			}
			// printing new line for each row
			System.out.println();
		}
	}
}
```

Output:

```
* * * * *
* * * *
* * *
* *
*
* *
* * *
* * * *
* * * * *
```

24) RIGHT ANGLE PASCAL'S PATTERN

```
package patternProgramsPractice;

public class Pattern24 {
	//RIGHT ANGLE PASCAL'S PATTERN
	public static void main(String[] args) {
		int n=5;
		// There are two outer for loops in this program
		// This is Outer Loop prints the first half of
		// The Right Pascal triangle pattern
		for(int i=0;i<=n-1;i++)
		{
			for(int j=0;j<=i;j++)
			{
				System.out.print("* ");
			}
			System.out.println();
		}
		//This Outer Loop Prints second half of the triangle
		for(int i=n-1;i>=0;i--)
		{
			for(int j=0;j<=i-1;j++)
			{
				System.out.print("* ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

25) LEFT ANGLE PASCAL'S PATTERN

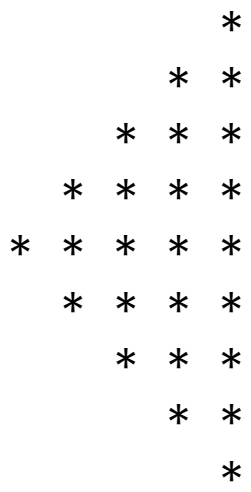
```
package patternProgramsPractice;

public class Pattern25 {
	//LEFT ANGLE PASCAL'S PATTERN
	public static void main(String[] args) {
		int n=5;
		//There are two outer for loops in this program
		//This Outer Loop prints the first half of
		// the Left Pascal triangle pattern
		for(int i=1;i<=n;i++)
		{
			for(int j=i;j<n;j++)
			{
				System.out.print("  ");
			}
			//Prints the stars of each row
			for(int k=1;k<=i;k++)
			{
				System.out.print("* ");
			}
			System.out.println();
		}
		//This Outer Loop Prints second half of the
triangle
		for(int i=n;i>=1;i--)
		{

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

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

Output:



26) K NUMERICAL PATTERN

```
package patternProgramsPractice;

public class Pattern26 {
	//K PATTERN WITH NUMBER
	public static void main(String[] args) {
		int n=5;
		// Printing Upper Half for n rows
		// This Loop is to iterate over each row in reverse
order
		for(int i=n;i>=1;i--){
			for(int j=1;j<=i;j++){
				System.out.print(j+" ");
			}
			System.out.println();
		}
		// Printing Lower Half for n-1 rows
		// This Loop is to iterate over each row

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

Output:

```
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

27) DIAMOND NUMERICAL PATTERN

```
package patternProgramsPractice;

public class Pattern27 {
	//DIAMOND NUMERIC PATTERN
	public static void main(String[] args) {
		int n=5;
		//Outer Loop to handle number of rows for 1st half
		for(int i=1;i<=n;i++)
		{
			// This inner loop prints the spaces
			for(int j=n-i;j>=1;j--)
			{
				System.out.print(" ");
			}

			for(int j=1;j<=2*i-1;j++)
			{
				System.out.print(i);
			}
			System.out.println();
		}
		//This outer loop handles number of rows for lower
half
		for(int i=n-1;i>=1;i--)
		{
			for(int j=n-i;j>=1;j--)
			{
				System.out.print(" ");
			}
			for(int j=1;j<=2*i-1;j++)
			{
				System.out.print(i);
			}
			System.out.println();
		}
	}
}
```

Output:

```
     1
    222
   33333
  4444444
555555555
  4444444
   33333
    222
     1
```

28) PYRAMID ALPHABET PATTERN

```
public class Pattern28 {
     //PYRAMID ALPHABET PATTERN
     public static void main(String[] args) {
          int n =6;
          char ch='A';

          for(int i=0;i<n;i++) {
               // printing spaces
               for(int j=0;j<n-i-1;j++) {
                    System.out.print(" ");
               }
               // printing alphabets
               for (int k=0;k<2*i+1;k++) {
                    System.out.print((char)(ch+k));
               }
               System.out.println();
          }
     }
}
```

Output:

```
     A
    ABC
   ABCDE
  ABCDEFG
 ABCDEFGHI
ABCDEFGHIJK
```

29) DIAMOND ALPHABET PATTERN

```
package patternProgramsPractice;

public class Pattern29 {
	//DIAMOND ALPHABET PATTERN
	public static void main(String[] args) {
		int n=6;
		char ch='A';

		//upper pyramid
		for(int i=1;i<=n;i++) {
			//printing spaces
			for(int j=n;j>i;j--) {
				System.out.print(" ");
			}
			//printing alphabets
			for(int k=0;k<i*2-1;k++) {
				System.out.print((char)(ch+k));
			}
			System.out.println();
		}

		//lower pyramid
		for(int i=1;i<=n-1;i++) {
			//printing spaces
			for(int j=0;j<i;j++) {
				System.out.print(" ");
```

```
                }
                //printing alphabets
                for (int k=0;k<(n-i)*2-1;k++) {
                     System.out.print((char)+(ch+k));
                }
                System.out.println();
            }
        }
}
```

Output:

```
      A
     ABC
    ABCDE
   ABCDEFG
  ABCDEFGHI
ABCDEFGHIJK
  ABCDEFGHI
   ABCDEFG
    ABCDE
     ABC
      A
```

30) K SHAPE ALPHABET PATTERN

```
package patternProgramsPractice;

public class Pattern30 {
	//K SHAPE ALPHABET PATTERN
	public static void main(String[] args) {
		int n=5,i,j;
		char ch='A';

		for (i=n-1;i>=0;i--)
		{
			for (j=0;j<=i;j++)
			{
				System.out.print((char)(ch+j)+" ");
			}
			System.out.println();
		}
		for (i=1;i<n;i++)
		{
			for (j=0;j<=i;j++)
			{
				System.out.print((char)(ch+j)+" ");
			}
			System.out.println();
		}
	}
}
```

Output:

```
A B C D E
A B C D
A B C
A B
A
A B
A B C
A B C D
A B C D E
```

Thank You