# Pattern Programs

## SquarePattern

```
a a a a
b b b b
c c c c
d d d d
```

Here, we want Row and Col. --- Logic, Nested for-loop:

1st loop ---- int i = 1; i <= numberOfRows; i++

2nd loop ---- int j = 1; j <= numberOfCols; j++

Outer give extra sysout statement. All this give our exapted result.

```
package pattern.letter;


public class Patternaaaa {
    public static void main(String[] args) {
        int numberOfRows = 4;
        int numberOfCols = 4;
        char letter = 'a';

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

```
a b c d
```

Here, we want Row and Col. --- Logic, Nested for-loop:

a b c d 1st loop ---- int i = 1; i <= numberOfRows; i++

a b c d 2nd loop ---- char ch = 'a'; ch < 'a' + numberOfCols; ch++

a b c d Outer give extra sysout statement. All this give our exapted result.

```
package pattern.letter;


public class Patternabcd {
    public static void main(String[] args) {
        int numberOfRows = 4;
        int numberOfCols = 4;


        for (int i = 1; i <= numberOfRows; i++) {
            for (char ch = 'a'; ch < 'a' + numberOfCols; ch++) {
                System.out.print(ch + " ");
            }
            System.out.println();
        }
    }
}
```

a b c d Here, we want Row and Col. --- Logic, Nested for-loop:

f g h I 1st loop ---- int i = 1; i <= numberOfRows; i++

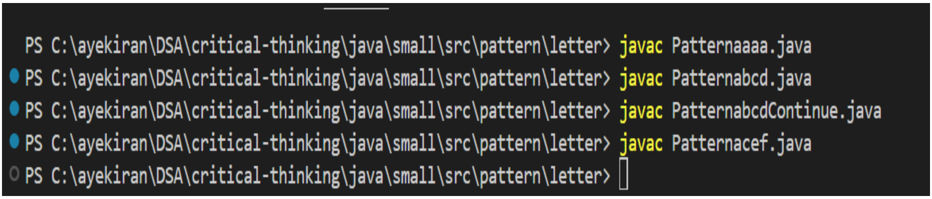
k l m n 2nd loop ---- char ch = 'a'; ch < 'a' + numberOfCols; ch++

p q r s Outer give extra sysout statement. All this give our exapted result.

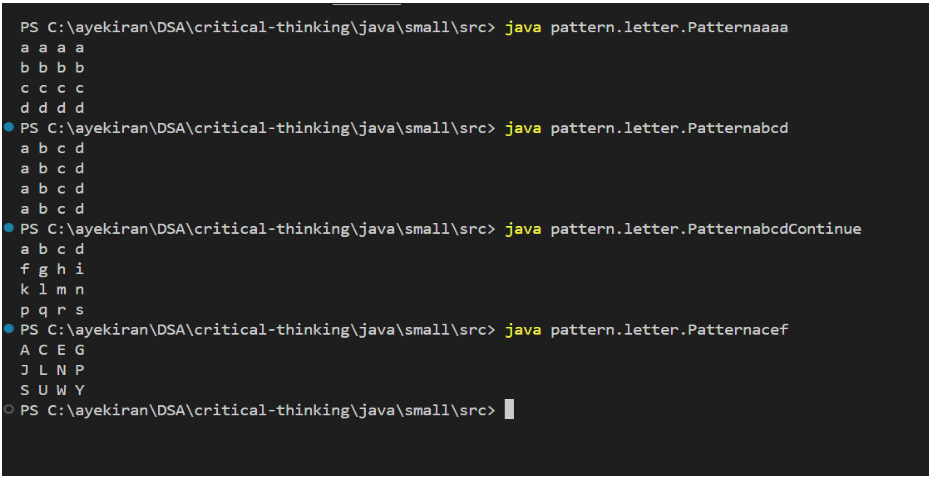
```
package pattern.letter;


public class PatternabcdContinue {
    public static void main(String[] args) {
        int numberOfRows = 4;
        int numberOfCols = 4;
        char letter = 'a';


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



```
PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac Patternaaaa.java
PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac Patternabcd.java
PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac PatternabcdContinue.java
PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac Patternacef.java
PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> 
```



```
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.Patternaaaa
a a a a
b b b b
c c c c
d d d d
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.Patternabcd
a b c d
a b c d
a b c d
a b c d
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.PatternabcdContinue
a b c d
f g h i
k l m n
p q r s
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.Patternacef
A C E G
J L N P
S U W Y
PS C:\ayekiran\DSA\critical-thinking\java\small\src> 
```

# Triangular Pattern

```
a
a b
a b c
a b c d
```

Here, we want Row and Col. --- Logic, Nested for-loop:

1st loop ---- int i = 1; i <= numberOfRows; i++

2nd loop ---- char ch = 'a'; ch < 'a' + i; ch++)

Outer give extra sysout statement. All this give our exapted result.

```
a b c d
a b c
a b
a
```

Here, we want Row and Col. --- Logic, Nested for-loop:

Logic is bit of change give results. Try it yourself, you could slove see on our github account. 🔗 Github

```
a
a b
a b c
a b c d
a b c
a b
a
```

Here, we want Row and Col. --- Logic, Nested for-loop:

Logic is bit of change give results. Try it yourself, you could slove see on our github account. 🔗 Github

```
a b c d
```

```
a b c
a b
a
a b
a b c
a b c d
```

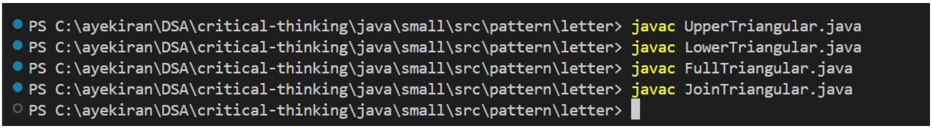
Here, we want Row and Col. --- Logic, Nested for-loop

Logic is bit of change give results. Try it yourself, you could slove see on our github account. 🔗 Github

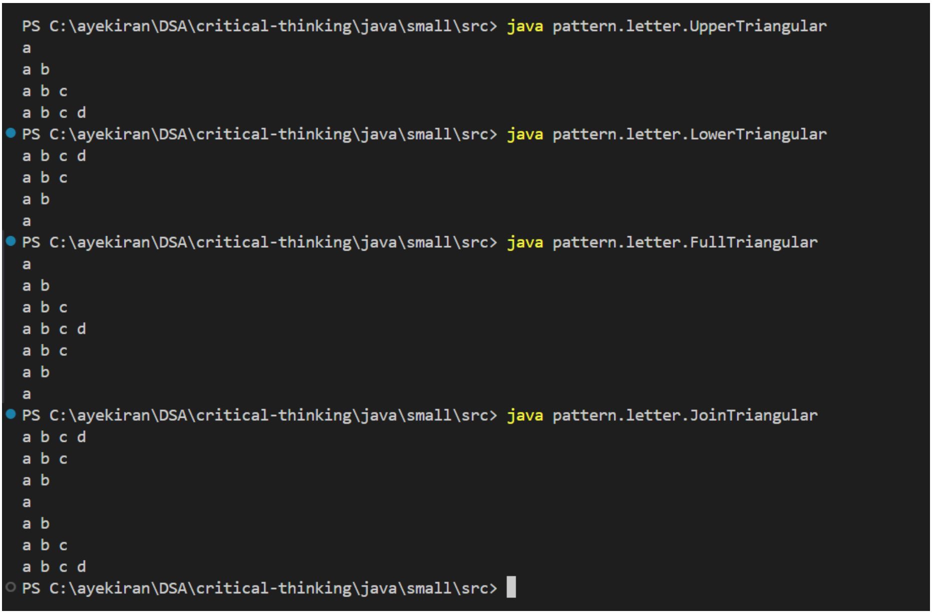
```java
package pattern.letter;


public class UpperTriangular {
    public static void main(String[] args) {
        int numberOfRows = 4;


        for(int i = 1; i <= numberOfRows; i++) {
            for(char ch = 'a'; ch < 'a' + i; ch++) {
                System.out.print(ch + " ");
            }
            System.out.println();
        }
    }
}
```



```
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac UpperTriangular.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac LowerTriangular.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac FullTriangular.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac JoinTriangular.java
○ PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> 
```



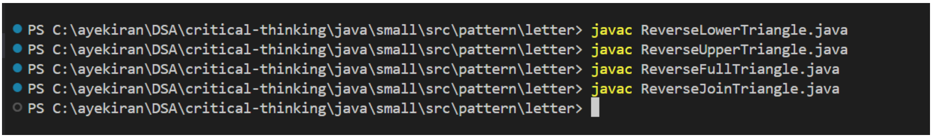
```
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.UpperTriangular
a
a b
a b c
a b c d
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.LowerTriangular
a b c d
a b c
a b
a
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.FullTriangular
a
a b
a b c
a b c d
a b c
a b
a
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.JoinTriangular
a b c d
a b c
a b
a
a b
a b c
a b c d
PS C:\ayekiran\DSA\critical-thinking\java\small\src>
```

## Reverse Triangular Pattern

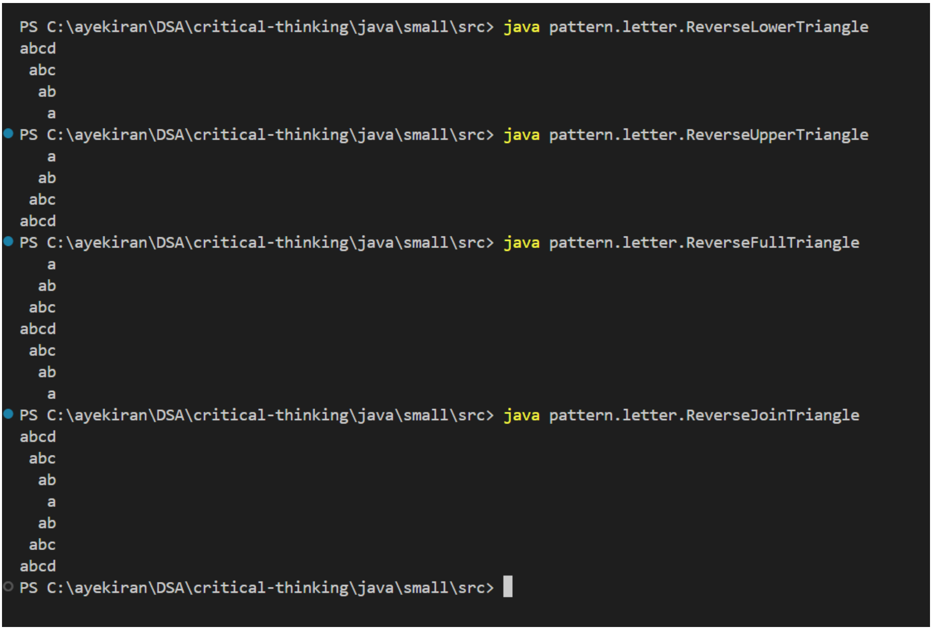
If all you observe clearly, try to solve below given problems.  So, that could help  practically solution.

```
abcd              a
 abc             a b
  ab            a b c
   a           a b c d


   a
  ab
 abc
abcd
 abc
  ab
   a
```



```
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac ReverseLowerTriangle.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac ReverseUpperTriangle.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac ReverseFullTriangle.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac ReverseJoinTriangle.java
○ PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> 
```



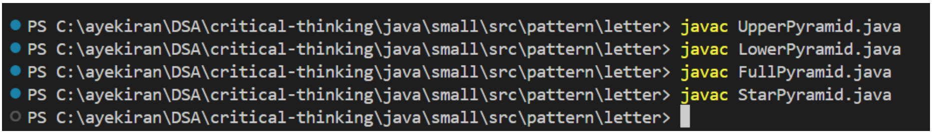
```
  PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.ReverseLowerTriangle
  abcd
   abc
    ab
     a
● PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.ReverseUpperTriangle
     a
    ab
   abc
  abcd
● PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.ReverseFullTriangle
     a
    ab
   abc
  abcd
   abc
    ab
     a
● PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.ReverseJoinTriangle
  abcd
   abc
    ab
     a
    ab
   abc
  abcd
○ PS C:\ayekiran\DSA\critical-thinking\java\small\src> 
```

## PyramidPattern

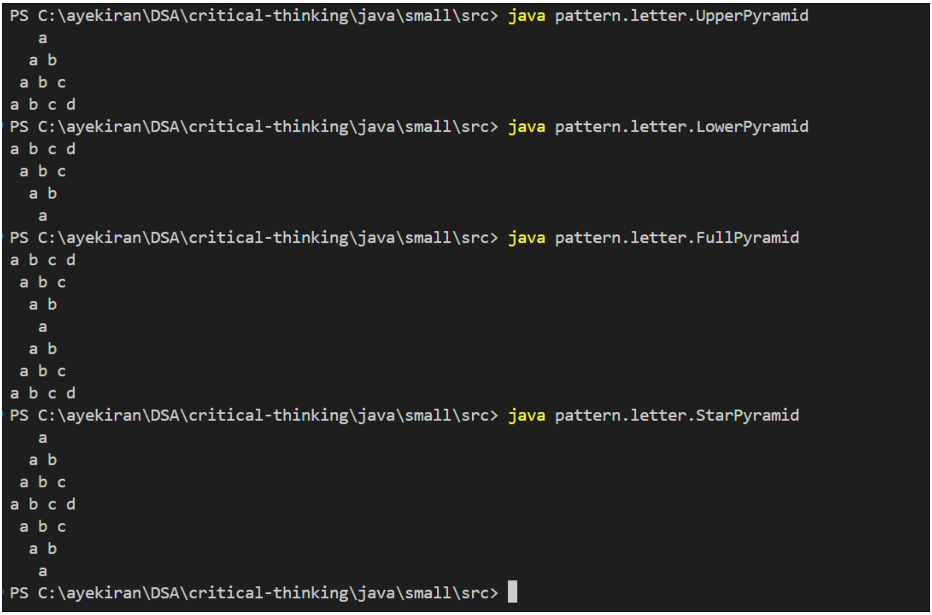
```
   a
  a b
 a b c
a b c d
```

```
package pattern.letter;

public class UpperPyramid {
    public static void main(String[] args) {
        int numberOfRows = 4;
        for(int i = 1; i <= numberOfRows; i++) {
         for (int spaces = 0; spaces < numberOfRows - i; spaces++) {
            System.out.print(" ");
         }
           for(char ch = 'a'; ch < 'a' + i; ch++) {
            System.out.print(ch + " ");
         }
         System.out.println();
      }
   }
}
```



```
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac UpperPyramid.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac LowerPyramid.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac FullPyramid.java
● PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> javac StarPyramid.java
○ PS C:\ayekiran\DSA\critical-thinking\java\small\src\pattern\letter> 
```



```
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.UpperPyramid
   a
  a b
 a b c
a b c d
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.LowerPyramid
a b c d
 a b c
  a b
   a
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.FullPyramid
a b c d
 a b c
  a b
   a
  a b
 a b c
a b c d
PS C:\ayekiran\DSA\critical-thinking\java\small\src> java pattern.letter.StarPyramid
   a
  a b
 a b c
a b c d
 a b c
  a b
   a
PS C:\ayekiran\DSA\critical-thinking\java\small\src> 
```

@ayekiran