1. Write a program to print the following pattern

Sample Input:

Enter the number to be printed: 1

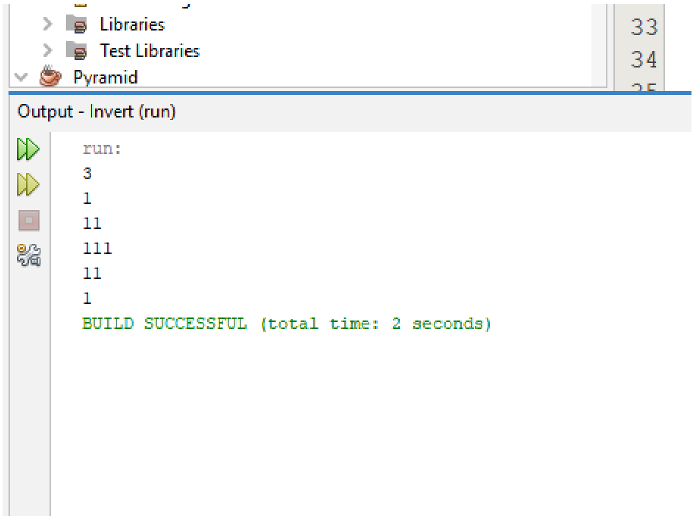
Max Number of time printed: 3

```
1
11
111
11
1
```

Program:

```
import java.util.*;
public class Invert {
    public static void main(String[] args) {
        // TODO code application logic here
        int i,j,r;
        Scanner s = new Scanner(System.in);
        r = s.nextInt();
        for(i=0;i<r;i++)
        {
            for(j=0;j<=i;j++)
            {
                System.out.print("1");
            }
            System.out.println();
        }
        for(i=r-1;i>=1;i--)
    {
      for(j=1;j<=i;j++)
      {
        System.out.print("1");
      }
      System.out.println();
    }
  }
}
```

Output:



```
run:
3
1
11
111
11
1
BUILD SUCCESSFUL (total time: 2 seconds)
```

2. Write a program to print the following pattern

Sample Input:

Enter the Character to be printed: %

Max Number of time printed: 3

%

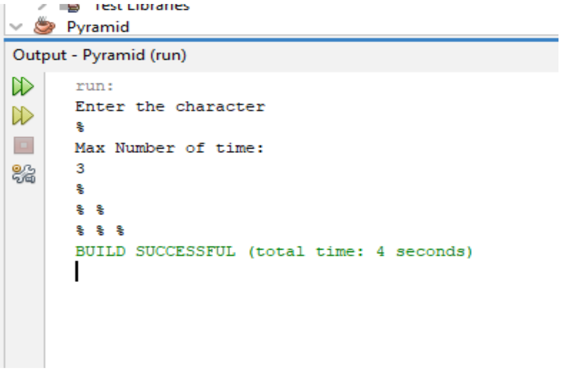
% %

% % %

Program:

```
import java.util.*;
public class Pyramid {
  public static void main(String[] args) {
    int i,j,a;
    String c;
    Scanner s = new Scanner(System.in);
    System.out.println("Enter the character");
    c = s.next();
    System.out.println("Max Number of time:");
    a = s.nextInt();

    for(i=0;i<a;i++)
    {
      for(j=0;j<=i;j++)
      {
        System.out.print(c+" ");
      }
      System.out.println();
    }
}
}
```



```
run:
Enter the character
$
Max Number of time:
3
$
$ $
$ $ $
BUILD SUCCESSFUL (total time: 4 seconds)
```

3. Write a program for matrix addition?

Sample Input:

Mat1 = 1 2

5 3

Mat2 = 2   3

4   1

Sample Output:

Mat Sum = 3   5

9 4

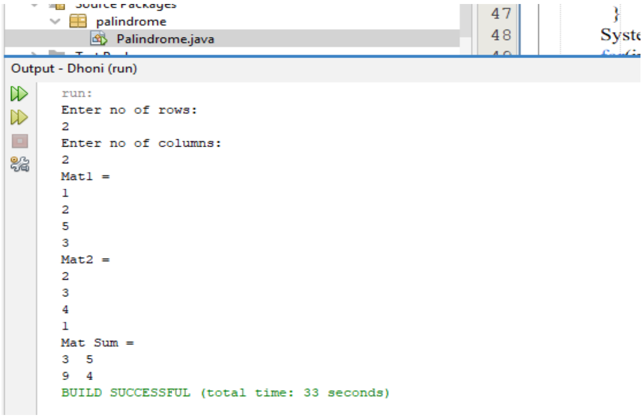
Program:

```
import java.util.Scanner;
public class Dhoni {
  public static void main(String[] args) {
   int i,j;
    Scanner s = new Scanner(System.in);
     System.out.println("Enter no of rows:");
     i = s.nextInt();
     System.out.println("Enter no of columns:");
     j = s.nextInt();
     int a[][] = new int[i][j];
     int b[][] = new int[i][j];
     int c[][] = new int[i][j];

    System.out.println("Mat1 =");
    for(i=0;i<2;i++)
    {
      for(j=0;j<2;j++)
      {
         a[i][j] = s.nextInt();
      }
    }
    System.out.println("Mat2 =");
    for(i=0;i<2;i++)
    {
      for(j=0;j<2;j++)
      {
        b[i][j] = s.nextInt();

      }
     }
    System.out.println("Mat Sum =");
    for(i=0;i<2;i++)
    {
      for(j=0;j<2;j++)
      {
        c[i][j] = a[i][j]+b[i][j];
        System.out.print(c[i][j]+"  ");
      }
      System.out.println();
    }
  }
}
```

Output:



```
run:
Enter no of rows:
2
Enter no of columns:
2
Mat1 =
1
2
5
3
Mat2 =
2
3
4
1
Mat Sum =
3  5
9  4
BUILD SUCCESSFUL (total time: 33 seconds)
```

4. Write a program to print the below pattern

1

2 2

3 3 3

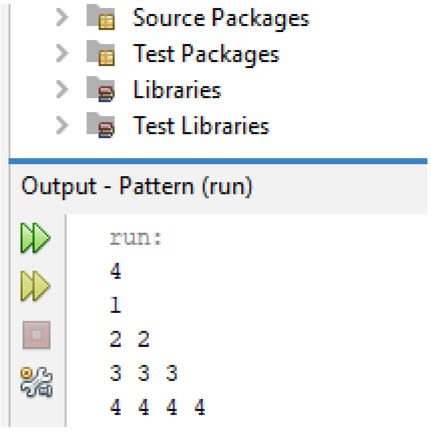
4 4   4   4

Program:

```
import java.util.*;
public class Pattern {
   public static void main(String[] args) {
      // TODO code application logic here
      int i,j,r;
      Scanner s = new Scanner(System.in);
      r = s.nextInt();
      for(i=1;i<=r;i++)
      {
         for(j=1;j<=i;j++)
         {
            System.out.print(i+" ");
         }
System.out.println();

    }
```

Output:



```
run:
4
1
2 2
3 3 3
4 4 4 4
```

5. Write a program to print the below pattern

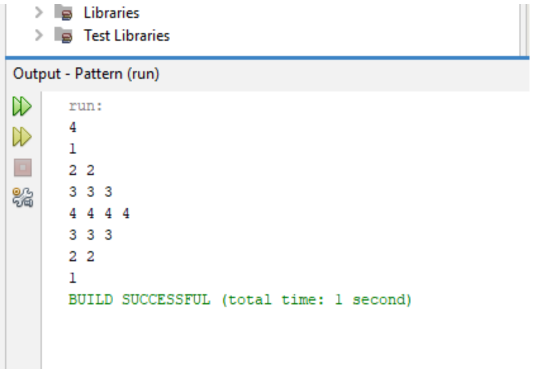
```
1
2 2
3  3  3
4  4   4   4
3  3  3
2  2
1
```

Program:

```
import java.util.*;
public class Pattern {
  public static void main(String[] args) {
    // TODO code application logic here
    int i,j,r;
    Scanner s = new Scanner(System.in);
    r = s.nextInt();
    for(i=1;i<=r;i++)
    {
      for(j=1;j<=i;j++)
      {
        System.out.print(i+" ");
      }
      System.out.println();
    }
    for(i=r-1;i>=1;i--)
    {
      for(j=0;j<i;j++)
      {
        System.out.print(i+" ");
      }
```

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

Output:



```
run:
4
1
2 2
3 3 3
4 4 4 4
3 3 3
2 2
1
BUILD SUCCESSFUL (total time: 1 second)
```

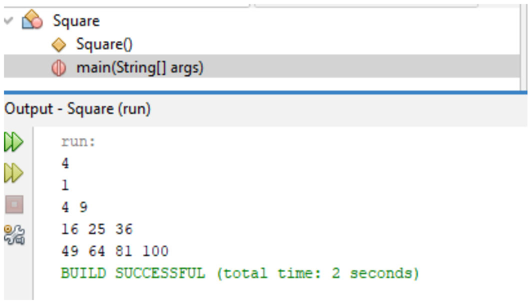
6. Write a program to print the below pattern

```
1
4    9
16   25   36
49   64   81   100
```

Program:

```
import java.util.*;
public class Square {
    public static void main(String[] args) {
        // TODO code application logic here
        int i,j,r,k=1;
        Scanner s = new Scanner(System.in);
        r = s.nextInt();
        for(i=1;i<=r;i++)
        {
            for(j=1;j<=i;j++)
            {
                System.out.print((int)Math.pow((k++),2)+" ");
            }
       System.out.println();
        }
    }
}
```

Output:



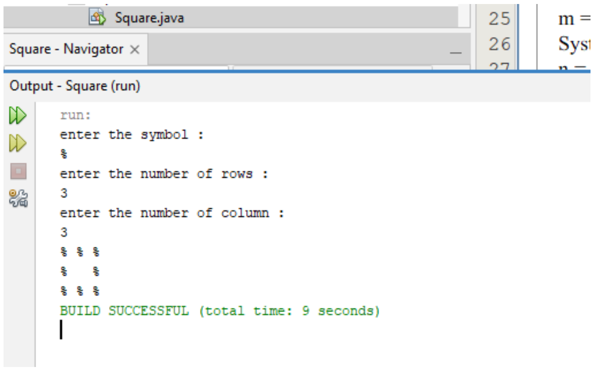
```
run:
4
1
4 9
16 25 36
49 64 81 100
BUILD SUCCESSFUL (total time: 2 seconds)
```

7. Write a program to print hollow Square Dollar pattern?

Program:

```
import java.util.*;
public class hallowsquare {
public static void main(String[]args)
{
int m,n,i,j;
String s;
Scanner v = new Scanner (System.in);
System.out.println("enter the symbol :");
s = v.nextLine();
System.out.println("enter the number of rows :");
m = v.nextInt();
System.out.println("enter the number of column :");
n = v.nextInt();
for (i=1;i<m+1;i++)
{
for (j=1;j<n+1;j++)
{
if(i == 1 || i == m || j == 1 || j == n)
{
System.out.print(s+" ");
}
else
{
System.out.print("  ");
}
}
System.out.println();
}
}
```

Output:



```
run:
enter the symbol :
%
enter the number of rows :
3
enter the number of column :
3
% % %
%   %
% % %
BUILD SUCCESSFUL (total time: 9 seconds)
```

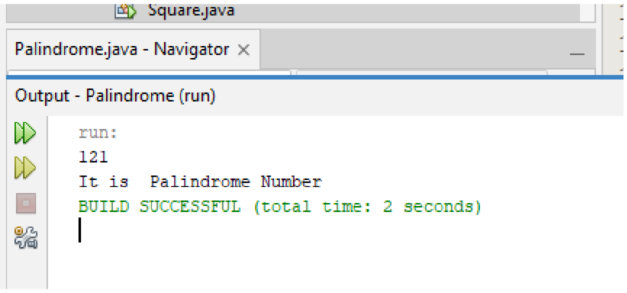
8 . Given an integer x, return true if x is a Palindrome , and false otherwise.

Program:

```
import java.util.*;
public class Palindrome {
   public static void main(String[] args) {
      // TODO code application logic here
      int num,r,reversed=0;
      Scanner s = new Scanner(System.in);
      num = s.nextInt();
      int original = num;
      while(num!=0)
      {
         r = num%10;
         reversed = reversed*10+r;
         num/=10;
      }

      if(original==reversed)
      {
         System.out.println("It is  Palindrome Number");
      }
      else
      {
         System.out.println("It is not Palindrome Number");
      }
   }
   }
```

Output:



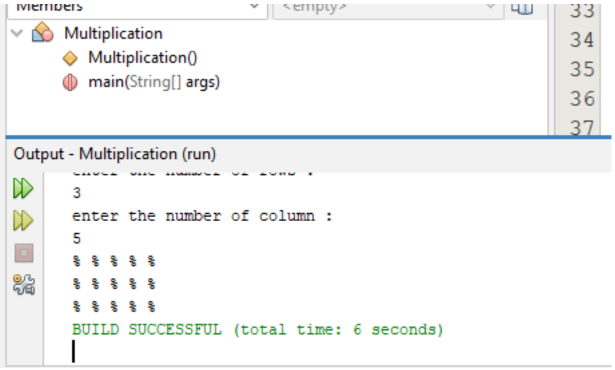
```
run:
121
It is  Palindrome Number
BUILD SUCCESSFUL (total time: 2 seconds)
```

9. Write a program to print rectangle symbol pattern. Get the symbol as input from user?

Program:

```
import java.util.*;
public class fullsquare {
public static void main(String[]args)
{
int m,n,i,j;
String s;
Scanner v = new Scanner (System.in);
System.out.println("enter the symbol :");
s = v.nextLine();
System.out.println("enter the number of rows :");
m = v.nextInt();
System.out.println("enter the number of column :");
n = v.nextInt();
for (i=1;i<m+1;i++)
{
for (j=1;j<n+1;j++)
{
System.out.print(s+" ");
}
System.out.println();
}
}
}
```

Output:



```
3
enter the number of column :
5
% % % % %
% % % % %
% % % % %
BUILD SUCCESSFUL (total time: 6 seconds)
```

10. Write a program to print inverted pyramid pattern.

Input: no of rows: 3

Output

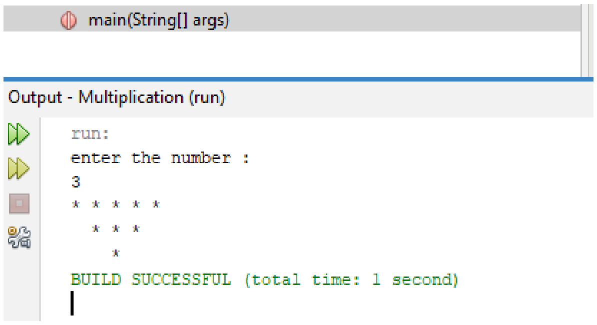
```
*****
 ***
  *
```

Program:

```
import java.util.*;
public class invertedtriangle
```

```
{
    public static void main(String[] args)
     {
       int rows;
       Scanner v = new Scanner(System.in);
       System.out.println("enter the number :");
       rows = v.nextInt();
       for (int i = rows; i >= 1; i--) {
          for (int j = 1; j <= rows-i; j++)
          {
             System.out.print("  ");
          }
          for (int k = 1; k <= 2*i-1; k++)
          {
             System.out.print("* ");
          }
          System.out.println();
       }
    }
  }
```

Output:



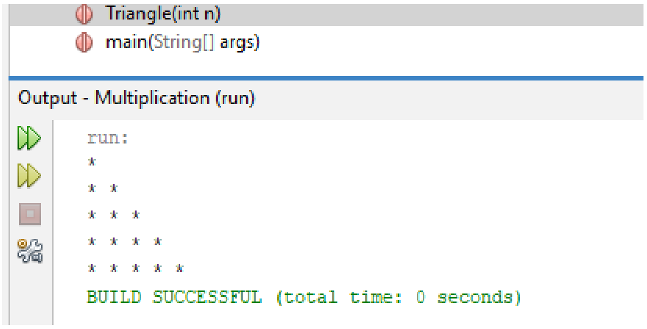
```
run:
enter the number :
3
* * * * *
  * * *
    *
BUILD SUCCESSFUL (total time: 1 second)
```

11. Write a program to print Right Triangle Star Pattern

Program:

```
class PyramidPattern
{
public static void Triangle(int n)
{
int i,j;
for (i=0;i<n;i++)
{
for (j=0;j<=i;j++)
{
System.out.print("* ");
}
System.out.println();
}
}
public static void main(String args[])
{
int k=5;
Triangle(k);
}
}
```

Output:



```
run:
*
* *
* * *
* * * *
* * * * *
BUILD SUCCESSFUL (total time: 0 seconds)
```

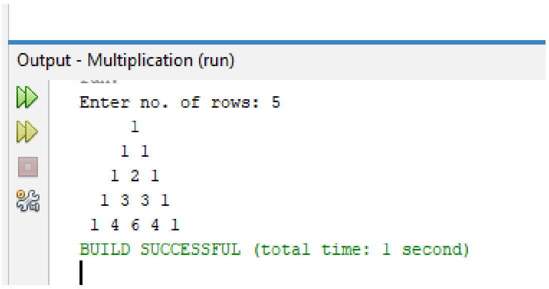
12. Write a program to print the below pattern?

```
             1
         1       1
     1       2       1
 1       3       3       1
1    4       6       4       1
```

Program:

```
import java.util.*;
public class pascal {
  public static void main(String[] args)
  {
    int row, i, j, space, num;
    Scanner sc=new Scanner(System.in);
    System.out.print("Enter no. of rows: ");
           if(!sc.hasNextInt())
           {
                      System.out.println("Invalid Enter only integers");
                      return;
           }
    row=sc.nextInt();
    for(i=0; i<row; i++)
    {
      for(space=row; space>i; space--)
      {
        System.out.print(" ");
      }
      num=1;
      for(j=0; j<=i; j++)
      {
        System.out.print(num+" ");
        num = num*(i-j)/(j+1);
      }
      System.out.println();
    }
  }
}
```

Output:



```
Enter no. of rows: 5
     1
    1 1
   1 2 1
  1 3 3 1
 1 4 6 4 1
BUILD SUCCESSFUL (total time: 1 second)
```

13. Write a program for matrix multiplication?

    Sample Input:

    Mat1 = 1 2

    5 3

    Mat2 = 2   3

    4   1

    Sample Output:

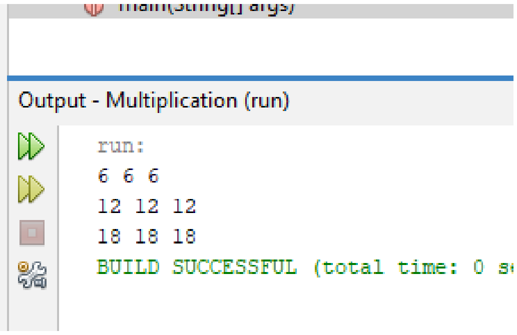
    Mat Sum = 10   5

    22   18

    Program:

```
public class matrixmulti {
public static void main(String args[])
{
int a[][]={{1,1,1},{2,2,2},{3,3,3}};
int b[][]={{1,1,1},{2,2,2},{3,3,3}};
int c[][]=new int[3][3];
for(int i=0;i<3;i++)
{
for(int j=0;j<3;j++)
{
c[i][j]=0;
for(int k=0;k<3;k++)
{
c[i][j]+=a[i][k]*b[k][j];
}
System.out.print(c[i][j]+" ");
}
System.out.println();
}
}
}
```

Output:



main(String[] args)

Output - Multiplication (run)

```
run:
6 6 6
12 12 12
18 18 18
BUILD SUCCESSFUL (total time: 0 s
```

14. Given a non-negative integer x, return the square root of x rounded down to the
    nearest integer. The returned integer should be non-negative as well.
    You must not use any built-in exponent function or operator.
    For example, do not use pow(x, 0.5) in c++ or x ** 0.5 in python.
    Program:

```
class Solution {

  public int mySqrt(int x) {

    if (x < 2) return x;

     int end = x / 2;

     int start = 1;

     while (start <= end) {
       int mid = (start + end) / 2;
       if ((long)mid*mid > x) {
         end = mid - 1;
       } else {
         start = mid + 1;
       }
     }
     return end;
  }
}
```

15. Find the Mean, Median, Mode of the array of numbers?
    Sample Input;:
    Array of elements = {16, 18, 27, 16, 23, 21, 19}

Sample Output:

Mean = 20

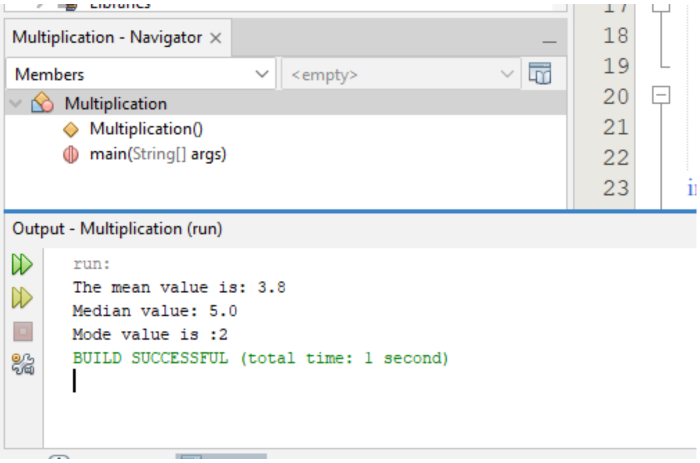
Median = 19

Mode = 16

Program:

```
import java.io.*;
import java.lang.*;
class Mean
{
public static void main(String[] args)
{
int[] invalue = new int[]{2,4,5,2,6};
int num_value=5;
double tot=0;
double mean=0;
for(int i=0; i<num_value; i++)
{
tot = tot+invalue[i];
}
mean = tot/num_value;
System.out.println("The mean value is: "+mean);
double median = 0;
double mid=0;
if(num_value%2 == 0)
{
int temp=(num_value/2)-1;
for(int i=0;i<num_value;i++)
{
if(temp==i || (temp+1)==i)
{
mid=mid+invalue[i];
}
}
mid=mid/2;
System.out.println("Median value is: "+mid);
}
else
{
int temp=(num_value/2);
for(int i=0;i<num_value;i++)
{
if(temp==i)
{
mid=invalue[i];
System.out.println("Median value: "+mid);
}
}
}
int i,j,z, tmp, maxCount, modeValue;
int[] tally=new int[num_value];
for(i=0;i<num_value;i++)
{
for(j=0;j<num_value-i;j++)
{
if(j+1!=num_value)
{
if(invalue[j]>invalue[j+1])
{
tmp=invalue[j];
invalue[j]=invalue[j+1];
invalue[j+1]=tmp;
}
}
}
}
for (i = 0; i < num_value; i++)
{
for(z=i+1;z<num_value;z++)
{
if(invalue[i]==invalue[z])
{
tally[i]++;
}
}
}
maxCount = 0;
modeValue = 0;
for (i = 0; i <num_value; i++)
{
if (tally[i] > maxCount)
{
maxCount = tally[i];
modeValue = invalue[i];
}
}
System.out.println("Mode value is :"+modeValue);
}
                                        }
```

Output:



```
run:
The mean value is: 3.8
Median value: 5.0
Mode value is :2
BUILD SUCCESSFUL (total time: 1 second)
```

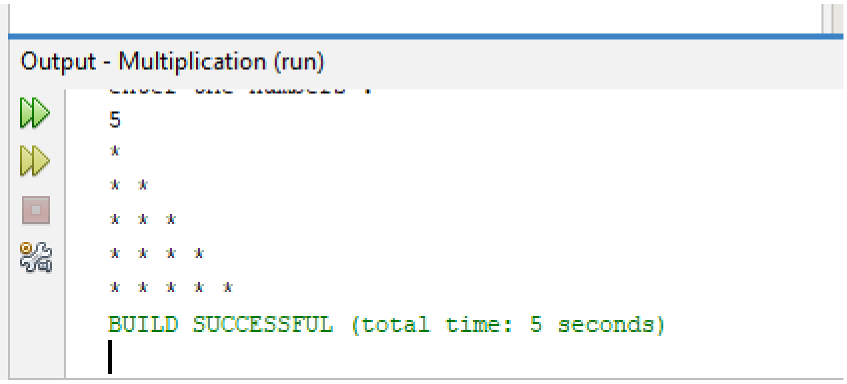
16. Right Triangle Star Pattern

Program:

```
import java.util.*;
public class rightangle {
  public static void main(String[] args) {
        int i,j,n;
        Scanner v = new Scanner(System.in);
        System.out.println("enter the numbers :");
        n = v.nextInt();
        for(i=0;i<n;i++)
        {
            for(j=0;j<=i;j++)
            {
                System.out.print("* ");
            }
            System.out.println();
        }

    }
}
```

Output:



```
enter the numbers :
5
* 
* * 
* * * 
* * * * 
* * * * * 
BUILD SUCCESSFUL (total time: 5 seconds)
```