

```

/*
Author: Jeffrey Huang
*/
import java.util.*;
public class PascalTriangleCreator
{
    public static long factorial(long n){
        /*
        The whole purpose of this method is to find the factorial of a number,
        since java does not have a built in method for it. Calculating n choose
        r is done using factorial, and since this code will be used repeatedly,
        it is wise to put it in a separate method.
        */
        long factorial;
        if (n==0){
            factorial=1;
        }
        else{
            factorial=1;
            for (int counter=1;counter<=n;counter++){
                factorial=factorial*counter;
            }
        }
        return factorial;
    }

    public static long FinalValue(long n, long r){
        //Calculates n choose r by calling the factorial method.
        return factorial(n) / ( factorial(n-r) * factorial(r) );
    }

    public static void main(String[] args) {
        Scanner sc=new Scanner (System.in);
        long rows=1;
        long i,j;
        while (rows!=0){
            System.out.println("How many rows of Pascal's triangle would you like to print? (0 to stop; 1-20
rows)");
            rows=sc.nextLong();
            //The following while loop ensures that the user cannot input an invalid number.
            while (rows<0||rows>20){
                System.out.println("Invalid input.");
            }
        }
    }
}

```

```

    System.out.println("How many rows of Pascal's triangle would you like to print? (0 to stop;
1-20 rows)");
    rows=sc.nextLong();
}
/*

```

The following if else block makes the code more efficient. Otherwise, if the user enters zero at any other point than at the start of the loop, the program will go through the long process of trying to print a triangle before terminating the program.

Using the following method, it is true that rows==0 is tested for twice, but it shortens the execution time immensely. And we know that when zero is true for the if statement, it is guaranteed to be true when breaking the loop.

```

*/
if (rows==0){
    System.out.println("Program terminated by user.");
}
else{
for(i = 0; i < rows; i++) {
    //Iterates through the number of rows required.
    for(j = 0; j <= rows-i; j++){
        System.out.print(" ");
        //Iterates the printing of spaces.
    }
    for(j = 0; j <= i; j++){
        if ((FinalValue(i, j))>9999) {
            System.out.print(" ");
        }
        else if ((FinalValue(i, j))>999){
            System.out.print(" ");
        }
        else if ((FinalValue(i, j))>99){
            System.out.print(" ");
        }
        else if ((FinalValue(i, j))>9){
            System.out.print(" ");
        }
        else{
            System.out.print(" ");
        }
        System.out.print(FinalValue(i, j));
        //Prints a number of spaces plus a number.
    }
}

```

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

}
}
```