

1. .

```
import java.util.Random;
import java.util.Scanner;

public class Proj05p1 {
    /* este programa tem como objetivo gerar um array com números aleatórios de 0 a 100 e
    que produzir, sobre esses valores no output a Média dos valores, o Maior número existente
    * e o conjunto de todos os números ímpares no array.
    */

    public static void main(String[] args) { //pedir ao utilizador o número de
    elementos desejado no array
        Scanner kb = new Scanner(System.in);
        System.out.print("Número de elementos do array: ");
        int arrayDim = kb.nextInt(); // quantidade de elementos que serão
    inseridos: Dimensão do array
        int numbers[] = new int[arrayDim]; // definição e instanciação do array
        Random random = new Random();

        for (int idx = 0; idx < numbers.length; idx++) // gerar aleatoriamente os
    elementos do array
        {
            numbers [idx] = random.nextInt(101);
        }

        // Imprimir o array original
        System.out.print(" Array: [ ");
        for (int i = 0; i < numbers.length; i++)
            System.out.print((i == numbers.length-1)? (numbers[i] + " "):(numbers [i] +
    " "));

        //Cálculo da Média
        double average;
        int sum = 0;
        for (int idx = 0; idx < numbers.length; idx++) // gerar aleatoriamente os
    elementos do array
        {
            sum = sum + numbers [idx];
        }
        average = sum / numbers.length;
        System.out.println("\n Average: " + average);

        // Determinar maior número do array (porção do algoritmo Bubble sort)
        for (int idx = 1; idx < numbers.length; idx++)
        { // fazer comparação sucessiva dos elementos do array arrastando o maior
    deles para a última posição
            if(numbers[idx] < numbers[idx - 1])
            {
                int exchangeAux = numbers [idx-1]; //colocar na variável de apoio o
    valor da variável que está a ser testada
                numbers [idx-1] = numbers[idx]; //colocar no índice da variável que
    está a ser testada o valor do índice que é mais baixo
                numbers[idx] = exchangeAux;
            }
        }
        System.out.println("Highest number: " + numbers[arrayDim-1]);
    }
```

```
// Apresentar em output apenas os números ímpares do array
System.out.print("Odd Numbers: [ ");
for (int idx = 1; idx < numbers.length; idx++)
{
    System.out.print((numbers[idx]%2 != 0) ? (numbers[idx] + " "): (""));
}
System.out.print("] ");
}
}
```

Output:

Número de elementos do array: 5

Array: [20 60 17 27 45]

Average: 33.0

Highest number: 60

Odd Numbers: [17 27 45]

2. .

```
import java.util.Random;
import java.util.Scanner;

public class Proj05p2 {
    /* este programa tem como objetivo gerar um array com números aleatórios de 0 a 100 e
    * realizar o número de rotações que o utilizador indicar, apresentando o resultado
    */

    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("Número de elementos do array: ");
        int arrayDim = kb.nextInt(); // quantidade de elementos que serão
        inseridos: Dimensão do array
        int numbers[] = new int[arrayDim]; // definição e instanciação do array
        Random random = new Random();

        for (int idx = 0; idx < numbers.length; idx++) // gerar aleatoriamente os
        elementos do array
        {
            numbers[idx] = random.nextInt(101);
        }

        // Imprimir o array original
        System.out.print("Array: [ ");
        for (int i = 0; i < numbers.length; i++)
            System.out.print((i == numbers.length-1) ? (numbers[i] + " "): (numbers[i] +
            " "));
        System.out.println("");

        // Realizar Rotações
        System.out.print("Quantas rotações pretende realizar? ");
        int rotationNum = kb.nextInt();
        for (int i=1; i <= rotationNum; i++) // trocas em todas as rotações
        {
            int aux = numbers[numbers.length-1];
            for (int j=numbers.length-1; j >0; j--) //trocas efetuadas numa rotação
            {
                numbers[j]=numbers[j-1];
            }
            numbers[0] = aux;
        }
    }
}
```

```
// Imprimir o array original
System.out.print("Array: [ ");
for (int i = 0; i < numbers.length; i++)
    System.out.print((i == numbers.length-1)? (numbers[i] + " "): (numbers[i] +
" "));
}
```

Output

Número de elementos do array: 5

Array: [65 68 38 52 24]

Quantas rotações pretende realizar? 4

Array: [68 38 52 24 65]

Número de elementos do array: 9

Array: [29 90 6 74 67 9 37 34 71]

Quantas rotações pretende realizar? 81

Array: [29 90 6 74 67 9 37 34 71]

3. .

```
import java.util.Random;
import java.util.Scanner;

public class Proj05p3Pascal {
    /* Esta aplicação apresenta os elementos das n linhas do triângulo de Pascal, sendo o
    número de linhas apresentadas indicado pelo utilizador.
    */
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("Triangle's height: ");
        int rowNum = kb.nextInt(); // quantidade de linhas do Triângulo de Pascal
        int pascal [][] = new int [rowNum][rowNum];

        pascal[0][0] = 1;
        for(int row = 1; row<rowNum; row++) //linhas do triângulo de Pascal
        {
            for(int idx = 0; idx<=row; idx++) // determinar elementos de cada linha do
            triângulo de Pascal
            {
                pascal [row][idx] = (idx==0)?(pascal [row][idx] = 1):(pascal [row-1][idx-
1] + pascal [row-1][idx]);
            }
        }
        for(int row = 0; row<rowNum; row++) //imprimir triângulo
        {
            for(int idx = 0; idx<=row; idx
            {
                System.out.print(pascal[row][idx] + " ");
            }
            System.out.println("");
        }
    }
}
```

Output:

Triangle's height: 10

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1
1 9 36 84 126 126 84 36 9 1
```