# Abstraction

Arrays, Strings, Methods, Matrices and Multidimensional Arrays
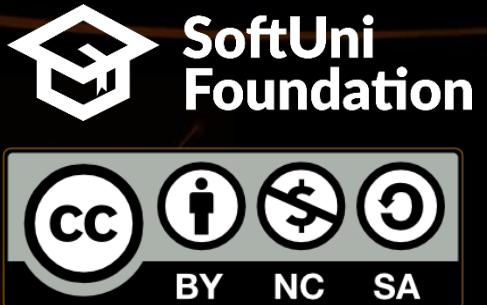
Advanced Java

SoftUni Team

Technical Trainers

Software University

http://softuni.bg

SoftUni Foundation
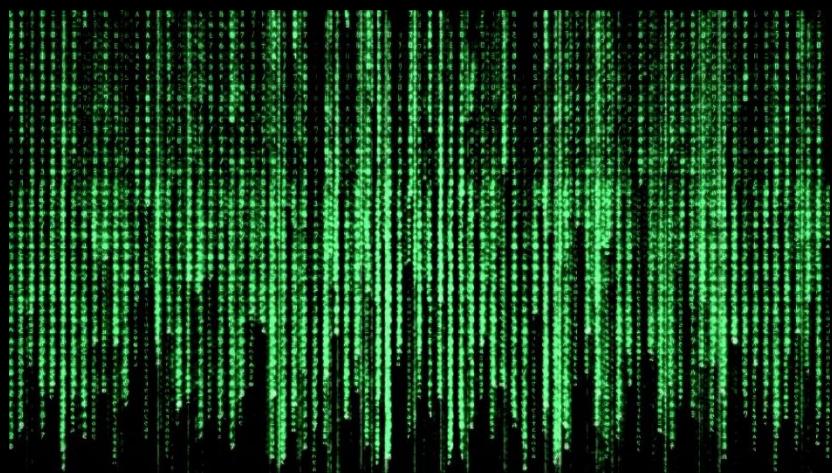
Table of Contents

1. What is Abstraction?

2. Methods

3. Arrays

4. Matrices and Multidimensional Arrays
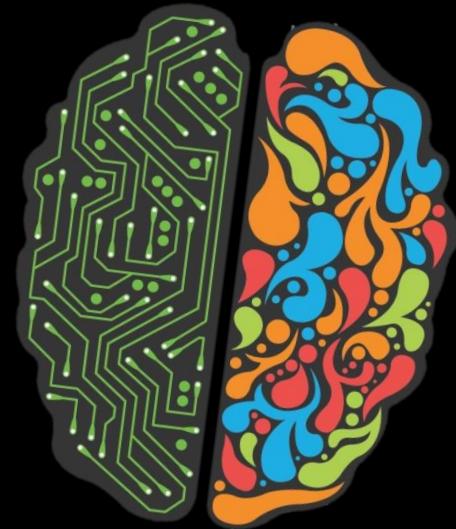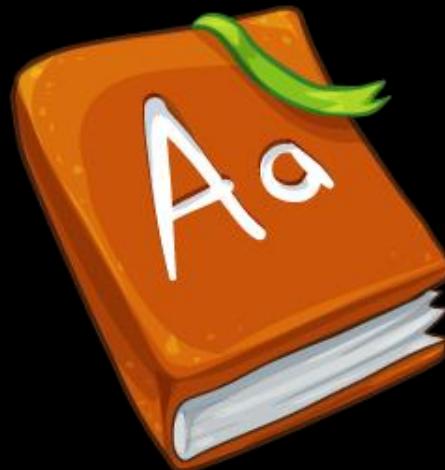
sli.do

#JavaAdvanced

Abstraction

Levels of abstraction in computer science

What is Abstraction?

- Using **bits** we can represent the different **numerical** types:
  - Integer numbers (**byte**, **int**, **short**, **long**)
  - Floating-point numbers (**float**, **double**)

That's **esier**!
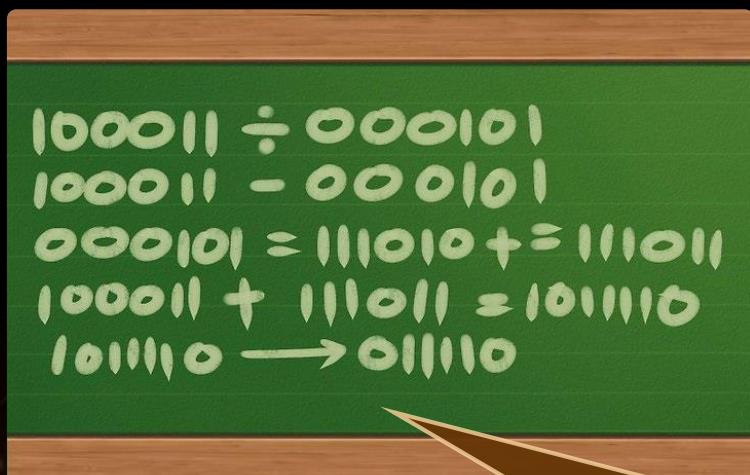
Calculating with bits is **hard**, isn't it?
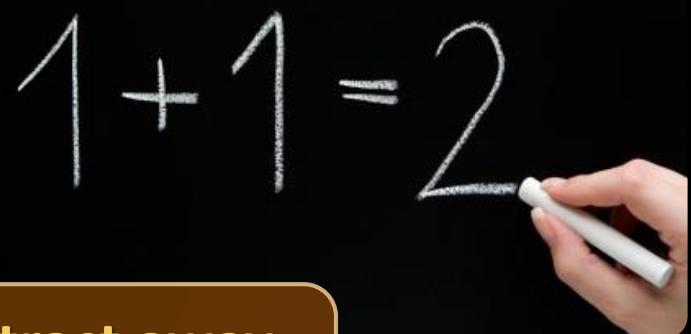
Let's **abstract away complex** details

What is Abstraction? (2)

- **Numbers** can be used to represent :
  - Characters
  - Colors

**Color** = amount of **red**, **green** and **blue**

Hiding the **irrelevant** data

Encoding = turning numbers into **characters**
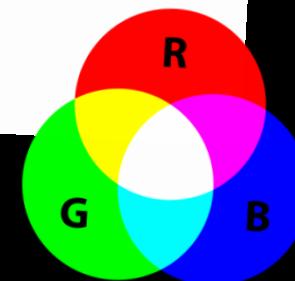
What is Abstraction? (3)

- By chaining **characters** we can make **strings**
  - depending on the **context** bit sequences have different **meaning**

Even **higher** level of **abstraction**
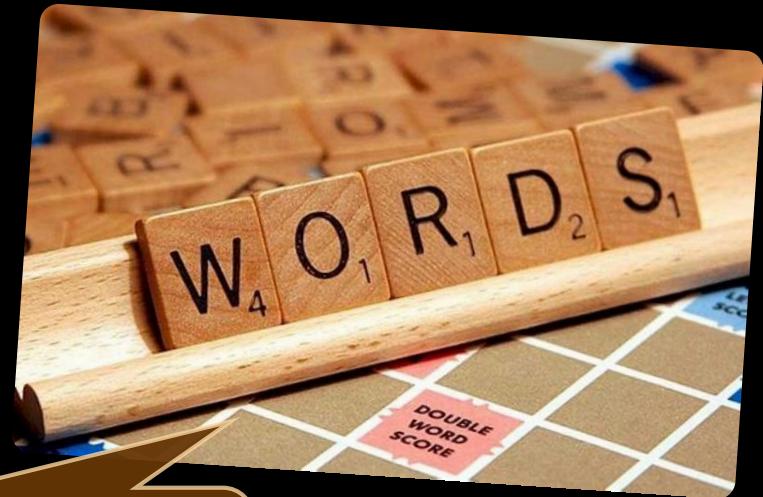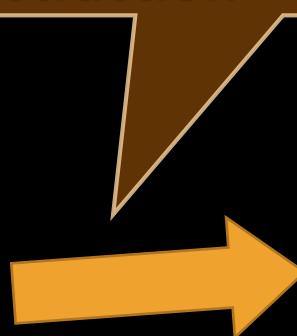
Programming becomes **human readable**

What is Abstraction? (4)

- Strings have build-in methods (**split()**, **substring()**, **contains**()...)

- We can write **custom** methods as well

  - **Reuse** code by packing it into **methods**

No need to create everything on your own
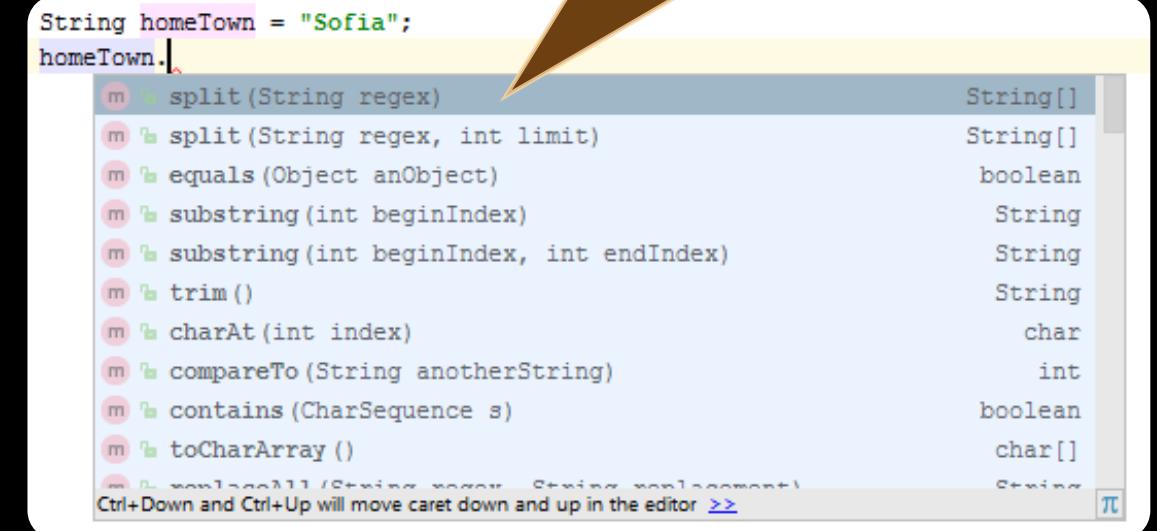
```
String homeTown = "Sofia";
homeTown.
    m  split(String regex)                          String[]
    m  split(String regex, int limit)               String[]
    m  equals(Object anObject)                        boolean
    m  substring(int beginIndex)                       String
    m  substring(int beginIndex, int endIndex)         String
    m  trim()                                          String
    m  charAt(int index)                                 char
    m  compareTo(String anotherString)                    int
    m  contains(CharSequence s)                       boolean
    m  toCharArray()                                   char[]
    m  replaceAll(String regex, String replacement)    String
Ctrl+Down and Ctrl+Up will move caret down and up in the editor  >>
```

Methods
Using methods to achieve abstraction

SoftUni Foundation

```
public static void main(String[] args)
    int[][] matrix = new int[4][4];
    int l = 0, n = matrix.length – 1, index = 1;
    int k = 0, m = matrix[0].length - 1;

    while (k <= m && l <= n)
        for (int c = k; c <= m; c++)
            matrix[l][c] = index++;
        l++;
        for (int r = l; r <= n; r++)
            matrix[r][m] = index++;
        m--;
        for (int c = m; c >= k ; c--)
            matrix[n][c] = index++;
        n--;
        for (int r = n; r >= l ; r--)
            matrix[r][k] = index++;
        k++;
```

Don't write code like this!

1	2	3	4
12	13	14	5
11	16	15	6
10	9	8	7

Methods – Print Spiral Matrix (2)

SoftUni Foundation

Understandable code

Name is self-explanatory

Lower code complexity

```java
public static void main(String[] args) {
    int[][] matrix = new int[7][7];
    int top = 0, bottom = matrix.length - 1;
    int left = 0, right = matrix[0].length - 1, index = 1;

    while (left <= right && top <= bottom) {
        index = fillFirstRow(matrix, index, left, right, top);

        index = fillLastCol(matrix, index, ++top, bottom, right);

        index = fillLastRow(matrix, index, left, --right, bottom);

        index = fillFirstCol(matrix, index, top, --bottom, left);
        left++;
    }
    printMatrix(matrix);
}
```

Executes only one task

Method Parameters and return types

Return type

Method names in Java must be **camelCase**

```java
private static int fillFirstRow(int[][] matrix, int index, int
left, int right, int top) {
    for (int col = left; col <= right; col++) {
        matrix[top][col] = index++;
    }
    return index;
}
```

Return statement can be of any type

Multiple parameters separated by comma

Problem: Calculate Triangle Area Method

- Create a method that calculates a triangle area by a given:

  - Base

  - Height

- Format the result to the second digit after the decimal separator

Input
4.00 2

Output
Area = 4.00

Check your solution here: https://judge.softuni.bg/Contests/775
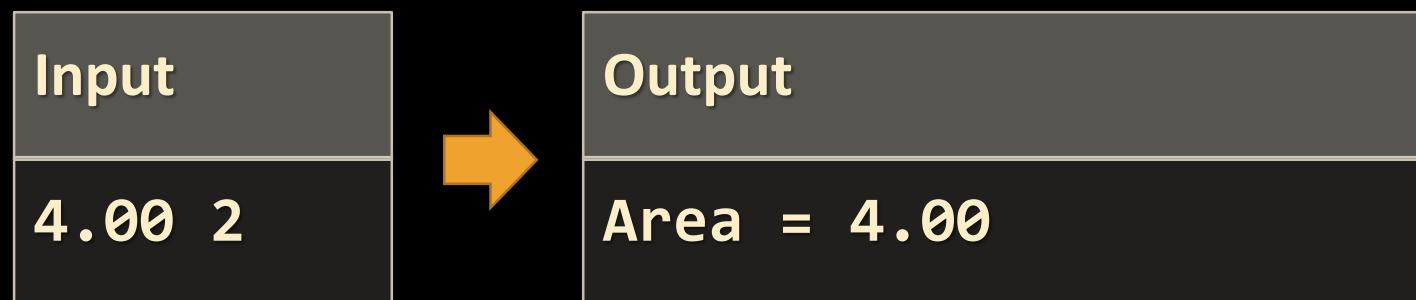
Solution: Calculate Triangle Area Method

```java
static double calcTriangleArea(double width, double height) {
    return width * height / 2;
}

public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    double width = input.nextDouble();
    double height = input.nextDouble();
    double area = calcTriangleArea(width, height);
    System.out.printf("Area = %.2f", area);
}
```

Check your solution here: https://judge.softuni.bg/Contests/775

What is Abstraction? (5)

- Data types can be packed in different **collections**

  - **Array** is the simplest collection type

Data is well **organized**

Data is **easier** to use
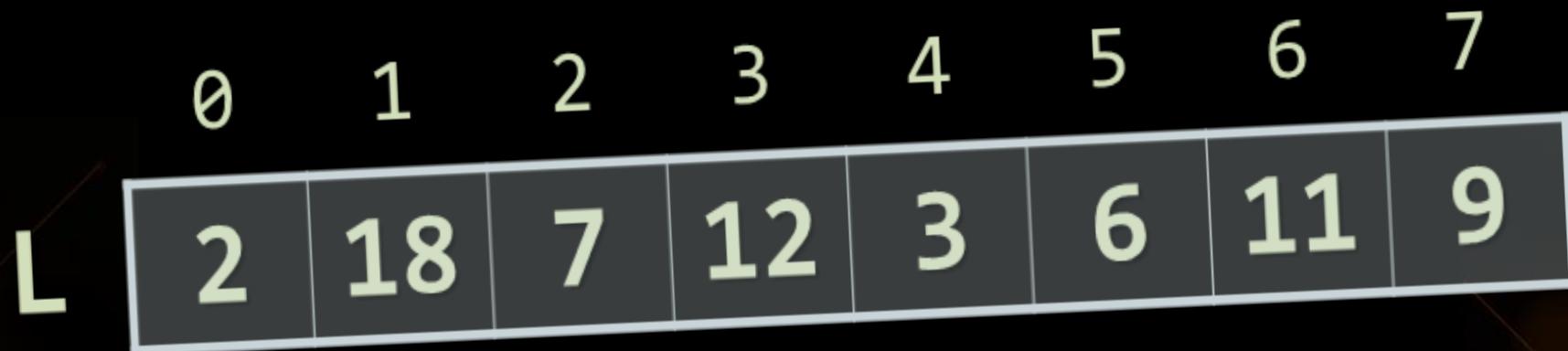
Arrays

Working with Arrays in Java

- Allocating an array :

Array of **10** elements

```java
int[] numbers = new int[10];
```

- Assigning values to the array elements:

```java
for (int i = 0; i < numbers.length; i++)
    numbers[i] = i + 1;
```

All elements are of the **same type**

- Accessing array elements :

Element **index**

```java
numbers[3] = 20;
numbers[5] = numbers[2] + numbers[7];
```

Problem: Encrypt, Sort and Print Array

- Write a program that:

  - Encrypts an array of strings → **sum** of:

    - Vowels code ***** current string length

    - Consonants code **/** current string length

  - **Sorts** the array

```
//Maya = 825
//M = 77 / 4 = 19
//a = 97 * 4 = 388
//y = 121 / 4 = 30
//a = 97 * 4 = 388
```

```
4
Peter
Maria
Katya
Maya
```

```
825
1032
1071
1532
```

Solution: Encrypt, Sort and Print Array

```java
Scanner scanner = new Scanner(System.in);
int n = scanner.nextInt();
String[] lines = new String[n];
for (int i = 0; i <= n; i++) {
   lines[i] = scanner.nextLine();
}

int[] linesSorted = encryptArray(lines);
Arrays.sort(linesSorted);
printArray(linesSorted);

   //TODO: write the methods on your own
```

Check your solution here: https://judge.softuni.bg/Contests/775

Practice: Methods and Arrays

Live Exercises in Class (Lab)
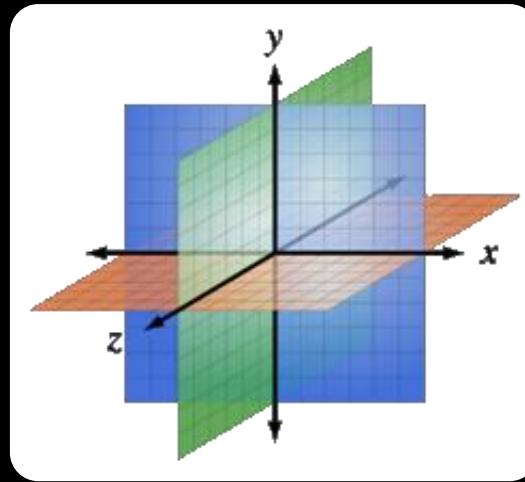
Multidimensional Arrays

Using Array of Arrays, Matrices and Cubes

What is Multidimensional Array?

- An **array** is a systematic arrangement of similar objects

- Arrays can have more than one dimension, e.g. **matrices**

- The most used **multidimensional** arrays are the **2-dimensional**

Matrix	COLUMS			
R O W S	[0][0]	[0][1]	[0][2]	[0][3]
	[1][0]	[1][1]	[1][2]	[1][3]
	[2][0]	[2][1]	[2][2]	[2][3]
	[3][0]	[3][1]	[3][2]	[3][3]

Row Index

Column Index
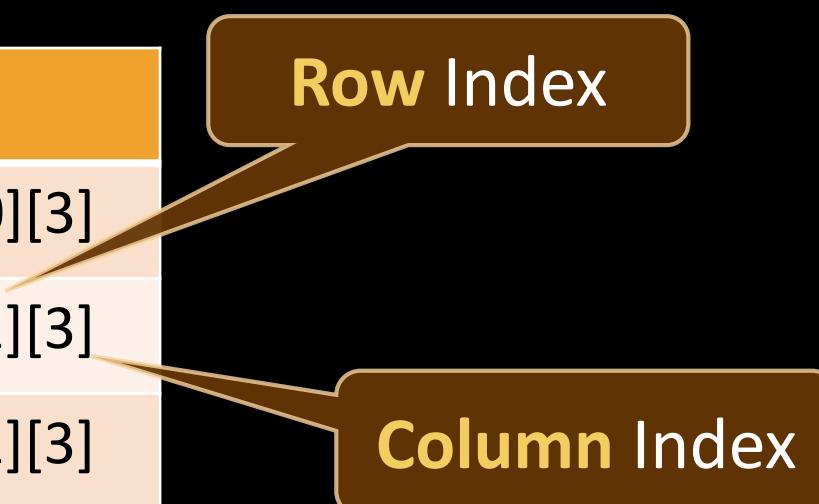
Declaring and Creating Multidimensional Arrays

- Declaring multidimensional arrays:

```
int[][] intMatrix;
float[][] floatMatrix;
String[][][] strCube;
```

- Creating a multidimensional array

  - Use **new** keyword

  - Must specify the size of each dimension

```
int[][] intMatrix = new int[3][4];
float[][] floatMatrix = new float[8][2];
String[][][] stringCube = new String[5][5][5];
```

Initializing Multidimensional Arrays

- Initializing a multidimensional array with values :

```
int[][] matrix = {
    {1, 2, 3, 4}, // row 0 values
    {5, 6, 7, 8}  // row 1 values
};
```

- Matrices are represented by a list of rows

  - Each row consists of a list of values

Accessing Elements

- Accessing N-dimensional array element:

```
nDimensionalArray[index₁] … [indexₙ]
```

- Getting element value example:

```
int[][] array = {{1, 2}, {3, 4}}
int element = array[1][1]; // element₁₁ = 4
```

	0	1
0	1	2
1	3	4

- Setting element value example:

```
int[][] array = new int[3][4];
for (int row = 0; row < array.length; row++)
  for (int col = 0; col < array[0].length; col++)
    array[row][col] = row + col;
```

Reading a Matrix – Example

```java
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    int rows = Integer.parseInt(scanner.nextLine());
    int cols = Integer.parseInt(scanner.nextLine());

    int[][] matrix = new int[rows][cols];

    for (int row = 0; row < rows; row++) {
        String[] inputTokens = scanner.nextLine().split(" ");
        for (int column = 0; column < cols; column++) {
            matrix[row][column] =
                Integer.parseInt(inputTokens[column]);
        }
    }
}
```

Problem: Sum of All Elements of Matrix

- Read a matrix from the console

- Print the number of **rows**

- Print the number of **columns**

- Print the sum of all **elements**

```
int[][] matrix = {
    { 5, 2, 3, 1 },
    { 1, 9, 2, 4 },
    { 9, 8, 6, 11 }
};
```

```
3
4
61
```

Check your solution here: https://judge.softuni.bg/Contests/775

Solution: Sum of All elements of Matrix

```java
public static void main(String[] args) {
    int[][] matrix = new int[4][4];
    System.out.println(matrix.length);
    System.out.println(matrix[0].length);

    int sum = 0;
    for (int row = 0; row < matrix.length; row++) {
        for (int col = 0; col < matrix[row].length; col++) {
            sum += matrix[row][col];
        }
        System.out.println(sum);
    }
}
```

Gets length of 0th dimension (**rows**)

Gets length of 1st dimension (**columns**)

Check your solution here: https://judge.softuni.bg/Contests/775

Problem: Find Specific Square in Matrix

- Find the 2x2 square with max sum in a given matrix

  - Read the matrix from the console

  - Find the biggest **sum** of 2x2 submatrix

  - Print the result in form of a new matrix

```
int[][] matrix = {
  {7, 1, 3, 3, 2, 1},
  {1, 3, 9, 8, 5, 6},
  {4, 6, 7, 9, 1, 0}
};
```

➡️

```
9, 8
7, 9
```

Check your solution here: https://judge.softuni.bg/Contests/775

Solution: Find Specific Square in Matrix

- Finding maximal sum of 2x2 submatrix

```java
int bestSum = Integer.MIN_VALUE;
int resultRow;
int resultCol;
for (int row = 0; row < matrix.length - 1; row++)
  for (int col = 0; col < matrix[row].length - 1; col++)
    int sum = matrix[row][col] + matrix[row][col + 1] +
        matrix[row + 1][col] + matrix[row + 1][col + 1];
    if (sum > bestSum)
      bestSum = sum;
      resultRow = row;
      resultCol = col;
```

Check your solution here: https://judge.softuni.bg/Contests/775

Problem: Pascal's Triangle

- Write a program that prints the Pascal Triangle
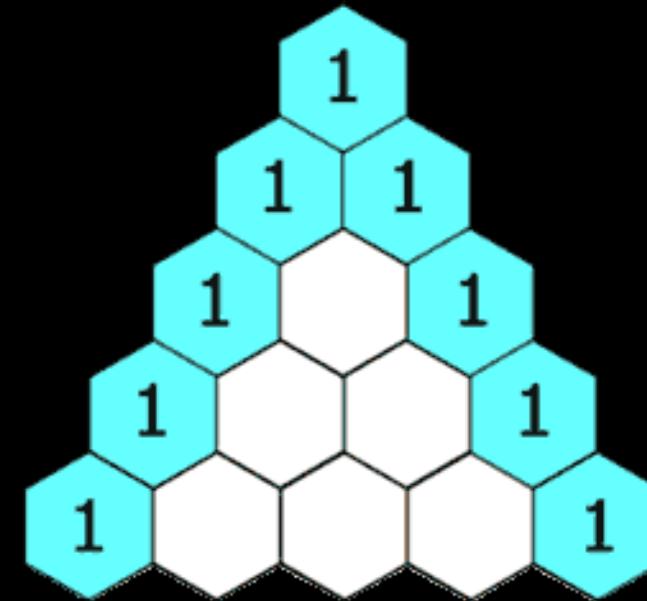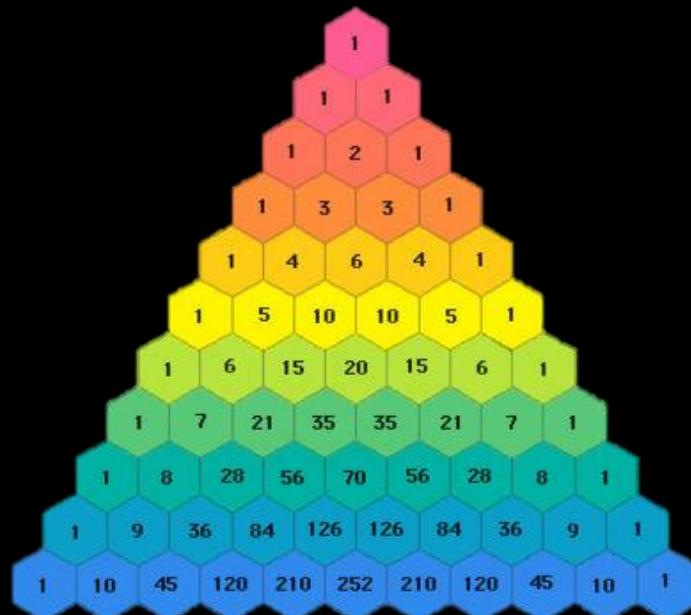
Solution: Pascal's Triangle

```
long[][] pascalTriangle = new long[height][height];
for (int currentHeight = 0; currentHeight < height; currentHeight++)
   pascalTriangle[currentHeight][0] = 1;
   pascalTriangle[currentHeight][currentHeight] = 1;
   if (currentHeight > 1)
     for (int i = 1; i < currentHeight; i++)
        long[] previousRow = pascalTriangle[currentHeight - 1];
        long previousRowSum = previousRow[i] + previousRow[i - 1];
        pascalTriangle[currentHeight][i] = previousRowSum;


   // TODO: print the triangle
```

Check your solution here: https://judge.softuni.bg/Contests/775

```java
public static void main(String[] args) {
    int[][] matrix = {
            {7, 1, 3, 3, 2, 1},
            {1, 3, 9, 8, 5, 6},
            {4, 6, 7, 9, 1, 0}

    };
```

Practice: Using Multidimensional Arrays

Live Exercises in Class (Lab)

What is Abstraction? (6)

- **Complex** details are *abstracted away* so the programmer works with **only** what is **necessary** to him

- A key part of **computer science** and life in general

- **Numbers**, **characters**, **strings** and **methods** represent different levels of abstraction

HOW DOES COMPUTER PROGRAMMING WORK?

MAGIC.
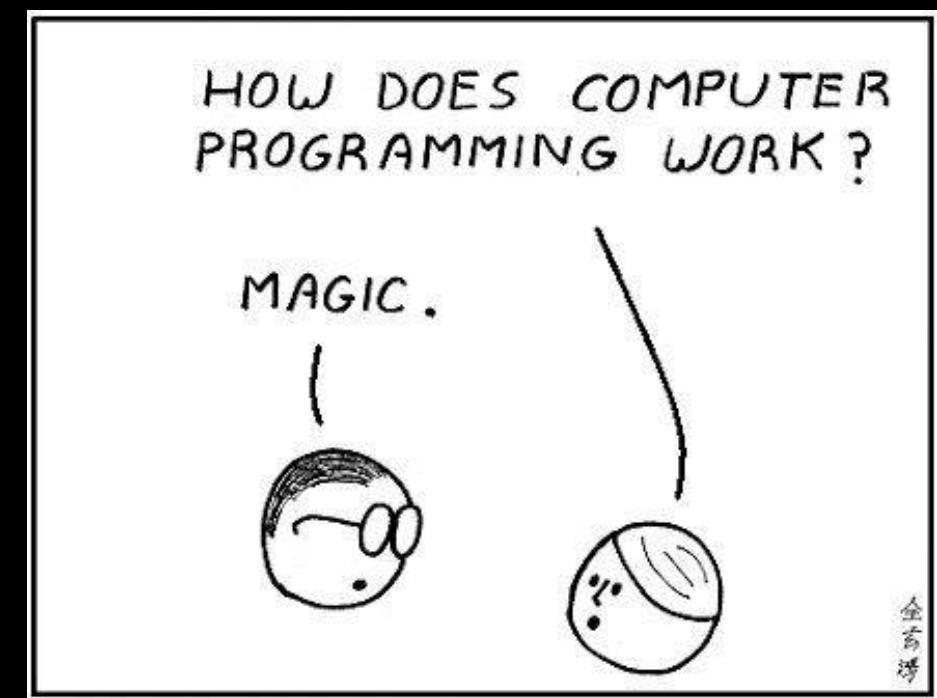
Summary

- Abstraction
  - **hides** all irrelevant data
  - **reduces complexity**
  - increases efficiency
- Methods → a tool for **achieving abstraction**
- Arrays can be **multidimensional**
- Matrices are like **tables**

Abstraction

SoftUni Foundation

Questions?

XS software

SmartIT
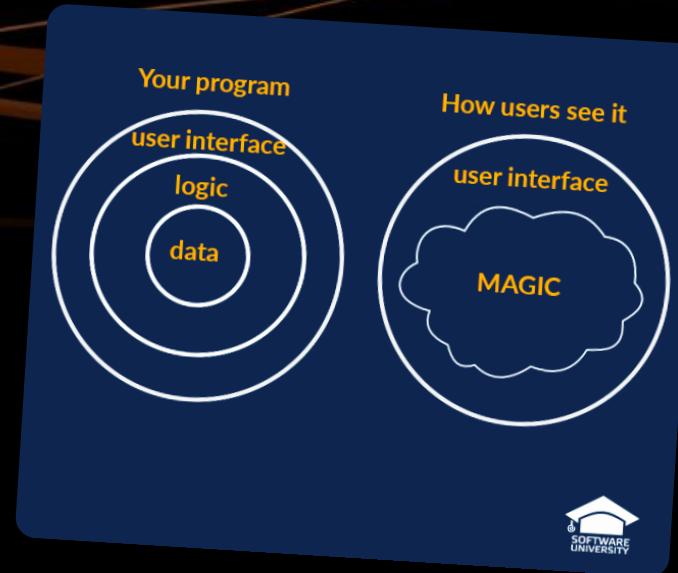
NETPEAK
SEO and PPC for Business

SUPERHOSTING.BG

INDEAVR
Serving the high achievers

telenor

SOFTWARE GROUP

INFRAGISTICS
DESIGN / DEVELOP / EXPERIENCE

https://softuni.bg/courses/

Trainings @ Software University (SoftUni)

- Software University – High-Quality Education, Profession and Job for Software Developers
  - softuni.bg
- Software University Foundation
  - http://softuni.foundation/
- Software University @ Facebook
  - facebook.com/SoftwareUniversity
- Software University Forums
  - forum.softuni.bg