1. **Basic Syntax, Conditional Statements and Loops**
   1. **Четене от конзолата**

int age = Integer.parseInt(scanner.nextLine()); //стринг с преобразуване към инт

String username = scanner.nextLine(); // стринг

double pricePerCapsule = Double.parseDouble(scanner.nextLine()); // стринг към дабъл

* 1. **Преобразуване между типове данни**

Кастване

char symbol = 'A';

int asciiValue1 = (int) symbol; // explicit casting, символ към аски код

int asciiValue2 = symbol; // implicit casting

String number = "145";

char symbol = number.charAt(0); //от стринг към символ

String symbolAsText = symbol + ""; // от символ към стринг

int digit = Integer.parseInt(symbolAsText); // от стринг към int

int digit = Integer.parseInt(inputNumber.charAt(position) + "") //позиция от стринг към инт

* 1. **Форматирано отпечатване**

System.out.printf("Name: %s, Age: %d, Grade: %.2f", name, age, averageGrade);

\n – нов ред

* 1. **Други**

group.equals("Students") // стрингът group е еднакъв на Students

!input.equals("Start") // стрингът input не е еднакъв със стрингът start

number % 2 != 0 // числото не е четно, %2 – остатък при делене на две

int lastDigit = number % 10 // последна цифра на число

number /= 10; // премахване на послдената цифра на число

if (Character.isUpperCase(symbol) // проверка дали започва с главна буква

1. **Масиви**
   1. **Обща информация**

//1. масивът е съвкупност от еднотипни елементи

//2. масивът има постоянна дължина array.length

//3. дължина на масив = максималния брой елементи, които можем да съхраним

//4. позиции / индекси -> 0 до последната (дължина на масива - 1)

//5. задавате стойност в масив: array[0] = 56;

//6. достъпвате стойност в масив: array[5]

* 1. **Пълнене на масив**

//вариант 1 за запълване на масив(статичен с предварително зададени елементи)

int [] dates = {4, 5, 6, 7};

//вариант 2 за запълване на масив (празен масив и добавяме елементи)

double[] prices = new double[10];

prices[0] = 34.5;

prices[1] = 23.5;

//вариант 3 за запълване на масив (елементите се въвеждат от конзолата на отделни редове)

int n = Integer.parseInt(scanner.nextLine());

int [] numbers = new int[n];

for (int position = 0; position <= numbers.length - 1; position++) { numbers[position] = Integer.parseInt(scanner.nextLine());

//вариант 4 за запълване на масив (елементите са на един ред)

String[] inputs = scanner.nextLine().split(" ");

int [] integerNumbers = Arrays.stream(scanner.nextLine().split(" ")).mapToInt(Integer::parseInt).toArray(); //int

double [] decimalNumbers = Аrrays.stream(scanner.nextLine().split(")).mapToDouble(Double::parseDouble).toArray(); //double

String names = "Desi Ivan Georgi Tanya";

String [] namesArray = names.split(" "); //["Desi", "Ivan", "Georgi", "Tanya"] //string

* 1. **Обединяване и преобразуване на елементите на масив**

System.out.println(String.join(" ", namesArray)); //"Desi Ivan Georgi Tanya" // обединяване

System.out.println(Arrays.toString(numbers).replace("[", "").replace("]", "") // Обединяване със замяна на разделител

"Desi".toCharArray() -> ['D', 'e', 's', 'i'] // Стринг към символен масив

* 1. **Извеждане на елементите на масив**

//използваме за обхождане for: има значение позицията на елемента

for (int position = 0; position <= daysOfWeek.length - 1; position++) {

System.out.println(daysOfWeek[position]);

//използваме за обхождане foreach: няма значение позицията на елемента

["Monday", "Tuesday", "Wednesday", "Thursday","Friday", "Saturday", "Sunday"]

for (String day : daysOfWeek) {

//какво искаме да повторим за всеки един елемент

System.out.println(day);

}

//2. отпечтване в обратен ред

//[10, 20, 30]

//обратен ред: последната позиция към първата

for (int position = numbers.length - 1; position >= 0; position--) {

System.out.print(numbers[position] + " ");