

ΠΑΡΑΔΕΙΓΜΑΤΑ - ΑΣΚΗΣΕΙΣ

- Προσθέστε δύο αριθμούς

Μάθετε πώς να προσθέσετε δύο αριθμούς στην Java:

Παράδειγμα

```
int x = 5;
int y = 6;
int sum = x + y;
System.out.println(sum); // Print the sum of x + y
```

- Προσθέστε δύο αριθμούς με εισαγωγή χρήστη

Μάθετε πώς μπορείτε να προσθέσετε δύο αριθμούς με εισαγωγή χρήστη:

Παράδειγμα

```
import java.util.Scanner; // Import the Scanner class

class MyClass {
    public static void main(String[] args) {
        int x, y, sum;

        Scanner myObj = new Scanner(System.in); // Create a Scanner
        object

        System.out.println("Type a number:");

        x = myObj.nextInt(); // Read user input
```

```
System.out.println("Type another number:");
y = myObj.nextInt(); // Read user input

sum = x + y; // Calculate the sum of x + y
System.out.println("Sum is: " + sum); // Print the sum
}
}
```

- Μετρήστε τον αριθμό των λέξεων σε μια συμβολοσειρά

Μπορείτε εύκολα να μετρήσετε τον αριθμό των λέξεων σε μια συμβολοσειρά με το ακόλουθο παράδειγμα:

Παράδειγμα

```
String words = "One Two Three Four";
int countWords = words.split("\\s").length;
System.out.println(countWords);
```

- Αντιστρέψτε μια συμβολοσειρά

Μπορείτε εύκολα να αντιστρέψετε μια συμβολοσειρά με χαρακτήρες με το ακόλουθο παράδειγμα:

Παράδειγμα

```
String originalStr = "Hello";
String reversedStr = "";

for (int i = 0; i < originalStr.length(); i++) {
```

```
reversedStr = originalStr.charAt(i) + reversedStr;
}
System.out.println("Reversed string: "+ reversedStr);
```

- Υπολογίστε το άθροισμα ενός πίνακα

Λάβετε το άθροισμα των στοιχείων του πίνακα:

Παράδειγμα

```
int[] myArray = {1, 5, 10, 25};
int sum = 0;
int i;

// Loop through the array elements and store the sum in the sum
variable
for (i = 0; i < myArray.length; i++) {
    sum += myArray[i];
}

System.out.println("The sum is: " + sum);
```

- Μετατροπή συμβολοσειράς σε πίνακα

Υπάρχουν πολλοί τρόποι μετατροπής μιας συμβολοσειράς σε πίνακα. Ο απλούστερος τρόπος είναι να χρησιμοποιήσετε τη `toCharArray()` μέθοδο:

Παράδειγμα

Μετατροπή συμβολοσειράς σε `char` πίνακα:

```
// Create a string
String myStr = "Hello";

// Convert the string to a char array
char[] myArray = myStr.toCharArray();

// Print the first element of the array
System.out.println(myArray[0]);
```

Μπορείτε επίσης να κάνετε κύκλο μέσα από τον πίνακα για να εκτυπώσετε όλα τα στοιχεία του πίνακα:

Παράδειγμα

```
// Create a string
String myStr = "Hello";

// Convert the string to a char array
char[] myArray = myStr.toCharArray();

// Print array elements
for (char i : myArray) {
    System.out.println(i);
}
```

Πώς να ταξινομήσετε έναν πίνακα

Μπορείτε να χρησιμοποιήσετε τη `sort()` μέθοδο, που βρίσκεται στο `java.util.Arrays`, για να ταξινομήσετε έναν πίνακα:

Παράδειγμα

```
import java.util.Arrays;
```

```
public class Main {  
    public static void main(String[] args) {  
        String[] cars = {"Volvo", "BMW", "Tesla", "Ford", "Fiat",  
"Mazda", "Audi"};  
        Arrays.sort(cars);  
        for (String i : cars) {  
            System.out.println(i);  
        }  
    }  
}
```

- Πώς να υπολογίσετε τον μέσο όρο των στοιχείων πίνακα

Δημιουργήστε ένα πρόγραμμα που υπολογίζει τον μέσο όρο διαφορετικών ηλικιών:

Παράδειγμα

```
// An array storing different ages  
int ages[] = {20, 22, 18, 35, 48, 26, 87, 70};  
  
float avg, sum = 0;  
  
// Get the length of the array  
int length = ages.length;  
  
// Loop through the elements of the array  
for (int age : ages) {
```

```
    sum += age;
}

// Calculate the average by dividing the sum by the length
avg = sum / length;

// Print the average
System.out.println("The average age is: " + avg);
```

- Πώς να βρείτε το μικρότερο στοιχείο σε έναν πίνακα

Δημιουργήστε ένα πρόγραμμα που βρίσκει τη χαμηλότερη ηλικία μεταξύ διαφορετικών ηλικιών:

Παράδειγμα

```
// An array storing different ages
int ages[] = {20, 22, 18, 35, 48, 26, 87, 70};

// Create a 'lowest age' variable and assign the first array
element of ages to it
int lowestAge = ages[0];

// Loop through the elements of the ages array to find the lowest
age
for (int age : ages) {
    // Check if the current age is smaller than the current 'lowest
age'
    if (lowestAge > age) {
```

```
        // If the smaller age is found, update 'lowest age' with that
        element

        lowestAge = age;
    }
}

// Output the value of the lowest age

System.out.println("The lowest age in the array is: " +
lowestAge);
```

• Βρόχος μέσω μιας ArrayList

Κάντε βρόχο μέσα από τα στοιχεία ενός `ArrayList`:

Παράδειγμα

```
public class Main {

    public static void main(String[] args) {

        ArrayList<String> cars = new ArrayList<String>();

        cars.add("Volvo");

        cars.add("BMW");

        cars.add("Ford");

        cars.add("Mazda");

        for (String i : cars) {

            System.out.println(i);

        }

    }

}
```

• Βρόχος μέσω ενός HashMap

Κάντε βρόχο στα στοιχεία του a `HashMap` με έναν βρόχο **για κάθε** .

Σημείωση: Χρησιμοποιήστε τη `keySet()` μέθοδο εάν θέλετε μόνο τα κλειδιά και χρησιμοποιήστε τη `values()` μέθοδο εάν θέλετε μόνο τις τιμές:

Παράδειγμα

```
// Print keys
for (String i : capitalCities.keySet()) {
    System.out.println(i);
}
```

Παράδειγμα

```
// Print values
for (String i : capitalCities.values()) {
    System.out.println(i);
}
```

Παράδειγμα

```
// Print keys and values
for (String i : capitalCities.keySet()) {
    System.out.println("key: " + i + " value: " +
capitalCities.get(i));
}
```

Βρόχος μέσω Enum

Ο τύπος enum έχει μια `values()` μέθοδο, η οποία επιστρέφει έναν πίνακα με όλες τις σταθερές enum. Αυτή η μέθοδος είναι χρήσιμη όταν θέλετε να κάνετε βρόχο μέσω των σταθερών ενός enum:

Παράδειγμα

```
enum Level {  
    LOW,  
    MEDIUM,  
    HIGH  
}  
  
public class Main {  
    public static void main(String[] args) {  
        for (Level myVar : Level.values()) {  
            System.out.println(myVar);  
        }  
    }  
}
```

- Ελέγξτε εάν ένας αριθμός είναι ζυγός ή μονός

Μάθετε αν ένας αριθμός είναι άρτιος ή περιττός:

Παράδειγμα

```
int number = 5;  
  
// Find out if the number above is even or odd  
if (number % 2 == 0) {  
    System.out.println(number + " is even.");  
} else {  
    System.out.println(number + " is odd.");  
}
```

• Τετράγωνη ρίζα ενός αριθμού

Μπορείτε να χρησιμοποιήσετε `Math.sqrt()` για να βρείτε την τετραγωνική ρίζα ενός αριθμού:

Παράδειγμα

Βρείτε την τετραγωνική ρίζα του 64:

```
Math.sqrt(64);
```

• Πώς να δημιουργήσετε έναν τυχαίο αριθμό

Μπορείτε να χρησιμοποιήσετε `Math.random()` τη μέθοδο για να δημιουργήσετε έναν τυχαίο αριθμό.

`Math.random()` επιστρέφει έναν τυχαίο αριθμό μεταξύ 0,0 (συμπεριλαμβανομένου) και 1,0 (αποκλειστικά):

Παράδειγμα

```
Math.random();
```

Για να αποκτήσετε περισσότερο έλεγχο στον τυχαίο αριθμό, για παράδειγμα, εάν θέλετε μόνο έναν τυχαίο αριθμό μεταξύ 0 και 100, μπορείτε να χρησιμοποιήσετε τον ακόλουθο τύπο:

Παράδειγμα

```
int randomNum = (int)(Math.random() * 101); // 0 to 100
```