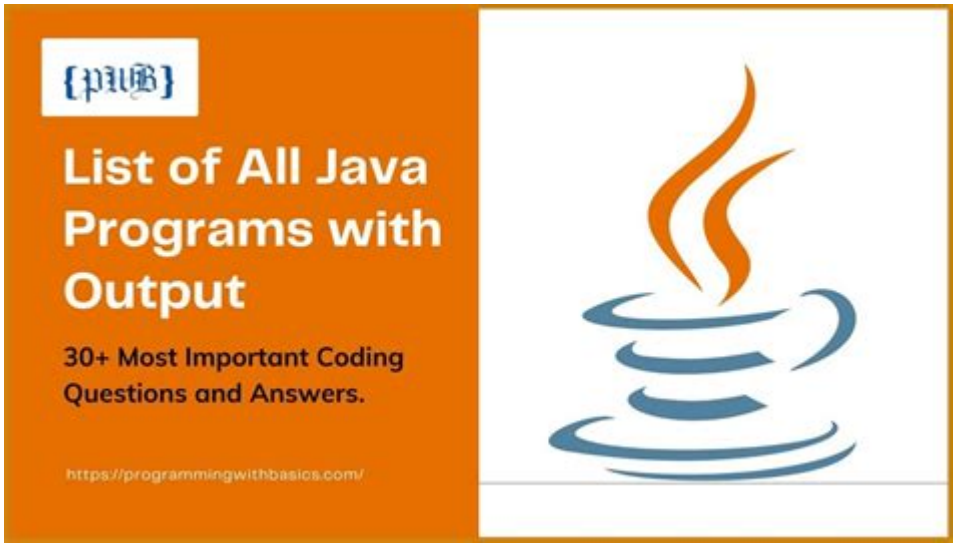
Java Programs List With Solutions



Java programs list with solutions is an essential resource for both novice and experienced developers looking to enhance their coding skills and problem-solving capabilities. Java, one of the most popular programming languages, is widely used in various applications, from web development to mobile apps. This article presents a comprehensive list of Java programs along with their solutions, designed to help programmers understand key concepts and improve their coding proficiency.

Why Learn Java Through Programs?

Learning Java through practical programming exercises is an effective way to grasp the language's concepts. Here are some reasons why working on Java programs is beneficial:

- **Hands-on Experience:** Writing code provides practical experience that theoretical learning cannot offer.
- **Problem-Solving Skills:** Engaging with real-world problems enhances critical thinking and problem-solving abilities.

- **Understanding Concepts:** Implementing various algorithms and data structures solidifies your understanding of Java fundamentals.

- **Portfolio Development:** Completed programs can serve as projects in your portfolio, demonstrating your skills to potential employers.

Java Programs List

Below is a curated list of Java programs that cover various levels of complexity, from beginner to advanced. Each program includes a brief description and a solution.

1. Hello World Program

The "Hello World" program is a simple yet essential starting point for any programming language.

Problem: Write a program that prints "Hello, World!" to the console.

Solution:

```java
public class HelloWorld {
public static void main(String[] args) {
System.out.println("Hello, World!");
}
}
```

2. Swap Two Numbers

Swapping two numbers is a basic exercise that helps beginners understand variable manipulation.

Problem: Write a program to swap two numbers without using a temporary variable.

Solution:

```java
public class SwapNumbers {
public static void main(String[] args) {
int a = 5;
int b = 10;

a = a + b; // a now becomes 15
b = a - b; // b becomes 5
a = a - b; // a becomes 10

System.out.println("After swapping: a = " + a + ", b = " + b);
}
}
```

3. Factorial of a Number

Calculating the factorial is a common exercise that introduces recursion.

Problem: Write a program to find the factorial of a given number using recursion.

Solution:

```java
public class Factorial {
public static void main(String[] args) {
int number = 5;
System.out.println("Factorial of " + number + " is " + factorial(number));
}

public static int factorial(int n) {
if (n == 0) {
return 1;
}
return n factorial(n - 1);
}
}
```

4. Fibonacci Series

The Fibonacci series is a classic example of iterative and recursive programming.

Problem: Write a program to display the Fibonacci series up to a certain number.

Solution:

```java
public class Fibonacci {
public static void main(String[] args) {
int n = 10, t1 = 0, t2 = 1;
System.out.println("Fibonacci Series till " + n + ":");

for (int i = 1; i <= n; ++i) {
```

```
System.out.print(t1 + ", ");
int sum = t1 + t2;
t1 = t2;
t2 = sum;
}
}
}
```

5. Prime Number Check

Checking whether a number is prime is an excellent exercise for practicing loops and conditionals.

Problem: Write a program to check if a number is prime.

Solution:

```java
import java.util.Scanner;

public class PrimeCheck {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a number: ");
int num = scanner.nextInt();
boolean isPrime = true;

for (int i = 2; i <= num / 2; i++) {
if (num % i == 0) {
isPrime = false;
break;
```

```
}
}

if (isPrime && num > 1) {
System.out.println(num + " is a prime number.");
} else {
System.out.println(num + " is not a prime number.");
}
}
}
```

6. Palindrome Checker

This program checks if a string is a palindrome, providing a great introduction to string manipulation.

Problem: Write a program to check if a string is a palindrome.

Solution:

```java
import java.util.Scanner;

public class Palindrome {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a string: ");
String str = scanner.nextLine();
String reversedStr = new StringBuilder(str).reverse().toString();

if (str.equals(reversedStr)) {
```

```
System.out.println(str + " is a palindrome.");
} else {
System.out.println(str + " is not a palindrome.");
}
}
}
```

7. Bubble Sort Algorithm

Sorting algorithms are fundamental in programming. Bubble sort is one of the simplest.

Problem: Write a program to perform bubble sort on an array.

Solution:

```java
public class BubbleSort {
public static void main(String[] args) {
int[] arr = {64, 34, 25, 12, 22, 11, 90};
int n = arr.length;
bubbleSort(arr, n);
System.out.println("Sorted array: ");
for (int i : arr) {
System.out.print(i + " ");
}
}

public static void bubbleSort(int[] arr, int n) {
for (int i = 0; i < n - 1; i++) {
for (int j = 0; j < n - i - 1; j++) {
```

```
if (arr[j] > arr[j + 1]) {
// swap arr[j+1] and arr[j]
int temp = arr[j];
arr[j] = arr[j + 1];
arr[j + 1] = temp;
}
}
}
}
}
```

8. Simple Calculator

Creating a simple calculator introduces users to methods and user input.

Problem: Write a program that performs basic arithmetic operations.

Solution:

```java
import java.util.Scanner;

public class SimpleCalculator {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter first number: ");
double num1 = scanner.nextDouble();
System.out.print("Enter second number: ");
double num2 = scanner.nextDouble();
```

```
System.out.println("Choose an operation: +, -, , /");
char operation = scanner.next().charAt(0);

double result;
switch (operation) {
case '+':
result = num1 + num2;
break;
case '-':
result = num1 - num2;
break;
case '':
result = num1 num2;
break;
case '/':
result = num1 / num2;
break;
default:
System.out.println("Invalid operation");
return;
}
System.out.println("Result: " + result);
}
}
```

9. Count Vowels and Consonants

This program helps in understanding string manipulation and conditional statements.

Problem: Write a program to count vowels and consonants in a string.

Solution:

```java
import java.util.Scanner;

public class CountVowelsConsonants {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a string: ");
String str = scanner.nextLine().toLowerCase();

int vowels = 0, consonants = 0;

for (char ch : str.toCharArray()) {
if (ch >= 'a' && ch <= 'z') {
if ("aeiou".indexOf(ch) != -1) {
vowels++;
} else {
consonants++;
}
}
}

System.out.println("Vowels: " + vowels);
System.out.println("Consonants: " + consonants);
}
}
```

10. Find the Largest Element in an Array

This program introduces array handling and searching algorithms.

Problem: Write a program to find the largest element in an array.

Solution:

```java
public class LargestElement {
public static void main(String[] args) {
int[] arr = {5, 7, 2, 8, 10, 3};
int max = arr[0];

for (int i =
```

Frequently Asked Questions

What are some common Java programs for beginners?

Common Java programs for beginners include: Hello World, Prime Number Checker, Factorial Calculator, Fibonacci Series, and Simple Calculator.

How can I write a Java program to find the factorial of a number?

You can use recursion or a loop to write a factorial program in Java. Here's a simple example using a loop:

```java
import java.util.Scanner;

public class Factorial {
```

```
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a number: ");
int number = scanner.nextInt();
long factorial = 1;
for(int i = 1; i <= number; i++) {
factorial = i;
}
System.out.println("Factorial of " + number + " is " + factorial);
}
}
```

What is a Java program to check if a number is prime?

You can check for prime numbers by dividing the number by all integers up to its square root. Here's an example:

```java
public class PrimeChecker {
public static void main(String[] args) {
int number = 29;
boolean isPrime = true;
for (int i = 2; i <= Math.sqrt(number); i++) {
if (number % i == 0) {
isPrime = false;
break;
}
}
System.out.println(number + " is prime: " + isPrime);
}
}
```

```

## How do I create a simple calculator in Java?

You can create a simple calculator using switch-case statements. Here's a basic example:

```java
import java.util.Scanner;

public class Calculator {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter first number: ");
double num1 = scanner.nextDouble();
System.out.print("Enter second number: ");
double num2 = scanner.nextDouble();
System.out.print("Enter an operator (+, -, , /): ");
char operator = scanner.next().charAt(0);
double result;

switch (operator) {
case '+':
result = num1 + num2;
break;
case '-':
result = num1 - num2;
break;
case '':
result = num1 num2;
break;
case '/':
result = num1 / num2;
```
```

```
break;
default:
System.out.println("Invalid operator");
return;
}
System.out.println("Result: " + result);
}
}
```

What is a Java program to print Fibonacci series?

You can print the Fibonacci series using a loop. Here's an example:

```java
public class Fibonacci {
public static void main(String[] args) {
int n = 10, firstTerm = 0, secondTerm = 1;
System.out.println("Fibonacci Series:");
for (int i = 1; i <= n; ++i) {
System.out.print(firstTerm + ", ");
int nextTerm = firstTerm + secondTerm;
firstTerm = secondTerm;
secondTerm = nextTerm;
}
}
}
```

How can I reverse a string in Java?

You can reverse a string using StringBuilder. Here's a quick example:

```java
public class StringReversal {
public static void main(String[] args) {
String original = "Hello";
String reversed = new StringBuilder(original).reverse().toString();
System.out.println("Reversed String: " + reversed);
}
}
```

What is a Java program to find the largest element in an array?

You can find the largest element in an array by iterating through the array. Here's a sample program:

```java
public class LargestInArray {
public static void main(String[] args) {
int[] numbers = {3, 5, 7, 2, 8};
int largest = numbers[0];
for (int number : numbers) {
if (number > largest) {
largest = number;
}
}
System.out.println("Largest number is: " + largest);
}
}
```

How do I sort an array in Java?

You can sort an array using the Arrays.sort() method. Here's an example:

```java
import java.util.Arrays;

public class ArraySort {
public static void main(String[] args) {
int[] numbers = {5, 3, 8, 1, 2};
Arrays.sort(numbers);
System.out.println("Sorted Array: " + Arrays.toString(numbers));
}
}
```

What is a Java program to find the sum of digits of a number?

You can find the sum of the digits of a number using a loop. Here's a simple example:

```java
import java.util.Scanner;

public class SumOfDigits {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a number: ");
int number = scanner.nextInt();
int sum = 0;

while (number != 0) {
sum += number % 10;
number /= 10;
}
System.out.println("Sum of digits: " + sum);
}
```

}

```

Find other PDF article:
https://soc.up.edu.ph/37-lead/files?ID=ZmX20-2653&title=life-guides-crossword-clue.pdf

# Java Programs List With Solutions

自学 *Java* 怎么入门？ - 知乎
在大学自学 Java，看视频教程和代码能看懂，自己写就写不出来，怎么样能够入门？

预测一下*2025*年*Java*就业趋势？ - 知乎
Jan 6, 2025 · Java曾经是IT行业最大的就业岗位，但是现在这个行业马上就要没了，一本的软件工程专业搞java得就业率还不到30%，未来几年java都不会起来了。

**Java**开发者社区**-CSDN**社区云
Dec 30, 2024 · 深知自学Java很难，所以最近整理了这份Java学习路线，适配2023年新版，是我自学Java时跟着学的路线，绝对是Java入门最佳选择 当然，你可能不需要学习全部的视频，可 …

**Java LTS**版本有哪些？ - 知乎
Java LTS版本 (长期支持版本)对于企业和开发者来说至关重要，能提供稳定的开发和生产环境，并在较长时间内获得官方支持，包括安全更新、Bug修复和性能提升，目前主要的Java LTS版本 …

**Java**社区**-CSDN**社区云
CSDNJava社区,Java论坛,为中国软件开发者打造学习和成长的家园

Java真的是要没落了吗？2024年还有希望吗？ - 知乎
Java真的是要没落了吗？ 2024年还有希望吗？作为SpringCloudAlibaba微服务架构实战派上下册和RocketMQ消息中间件实战派上下册的作者胡弦，最近很多从事Java的技术小伙伴都跑… 显 …

**Java**后端技术壁垒有哪些？ - 知乎
1 单机版的Java后端，比如基于spring boot的增删改查，中专生经过培训，半年能写很熟，外加能解决问题，这块没有技术壁垒。 2 顺带第1点说出去，JavaEE（就集合异常处理等）部分 …

**A Java Exception has occurred.**怎么解决啊**...-CSDN**社区
Feb 7, 2010 · 解决打包后双击提示"a java exception has occurred"的问题了。方法是删掉1.7版本的jdk，换上1.6版本的jdk（虽然我不确定此问题跟jdk有关）。换jdk版本后eclipse会出现错误 …

求助*!!! JDK*双击没反应*!-CSDN*社区
Jun 2, 2014 · 以下内容是CSDN社区关于求助!!! JDK双击没反应!相关内容，如果想了解更多关于Java SE社区其他内容，请访问CSDN社区。

*Spring Boot*使用*Redis*的*Lettuce*连接池，隔一段时间再连接就报连 …
Apr 13, 2019 · 以下内容是CSDN社区关于Spring Boot使用Redis的Lettuce连接池，隔一段时间再连接就报连接超时相关内容，如果想了解更多关于Java社区其他内容，请访问CSDN社区。
```

自学 **Java** 怎么入门？ - 知乎
在大学自学 Java，看视频教程和代码能看懂，自己写就写不出来，怎么样能够入门？

预测一下**2025**年**Java**就业趋势？ - 知乎
Jan 6, 2025 · Java曾经是IT行业最大的就业岗位，但是现在这个行业马上就要没了，一本的软件工程专业搞java得就业率还不到30%，未来几年java都不会起来了。

Java开发者社区-CSDN社区云
Dec 30, 2024 · 深知自学Java很难，所以最近整理了这份Java学习路线，适配2023年新版，是我自学Java时跟着学的路线，绝对是Java入门最佳选择 当然，你可能不需要学习全部的视频，可 …

Java LTS版本有哪些？ - 知乎
Java LTS版本 (长期支持版本)对于企业和开发者来说至关重要，能提供稳定的开发和生产环境，并在较长时间内获得官方支持，包括安全更新、Bug修复和性能提升，目前主要的Java LTS版本 …

Java社区**-CSDN**社区云
CSDNJava社区,Java论坛,为中国软件开发者打造学习和成长的家园

*Java*真的是要没落了吗？*2024*年还有希望吗？ - 知乎
Java真的是要没落了吗？ 2024年还有希望吗？作为SpringCloudAlibaba微服务架构实战派上下册和RocketMQ消息中间件实战派上下册的作者胡弦，最近很多从事Java的技术小伙伴都跑… 显 …

*Java*后端技术壁垒有哪些？ - 知乎
1 单机版的Java后端，比如基于spring boot的增删改查，中专生经过培训，半年能写很熟，外加能解决问题，这块没有技术壁垒。 2 顺带第1点说出去，JavaEE（就集合异常处理等）部分 …

A Java Exception has occurred.怎么解决啊...-CSDN社区
Feb 7, 2010 · 解决打包后双击提示"a java exception has occurred"的问题了。方法是删掉1.7版本的jdk，换上1.6版本的jdk（虽然我不确定此问题跟jdk有关）。换jdk版本后eclipse会出现错误 …

求助**!!! JDK**双击没反应**!-CSDN**社区
Jun 2, 2014 · 以下内容是CSDN社区关于求助!!! JDK双击没反应!相关内容，如果想了解更多关于Java SE社区其他内容，请访问CSDN社区。

*Spring Boot*使用*Redis*的*Lettuce*连接池，隔一段时间再连接就报连 …
Apr 13, 2019 · 以下内容是CSDN社区关于Spring Boot使用Redis的Lettuce连接池，隔一段时间再连接就报连接超时相关内容，如果想了解更多关于Java社区其他内容，请访问CSDN社区。

Explore our comprehensive Java programs list with solutions designed to enhance your coding skills. Learn more and tackle real-world programming challenges today!

Back to Home