

Java Training

✓ Java Tutorial

- What is Java
- History of Java
- C++ vs Java
- Hello Java Program
- Program Internal
- How to set path?
- JDK, JRE and JVM
- Internal Details of JVM
- Java Variables
- Java Data Types
- Unicode System
- Operators
- Keywords

✓ Control Statements

- Java If-else
- Java Switch
- Java For Loop
- Java While Loop
- Java Do While Loop
- Java Break
- Java Continue
- Java Comments
- Java Programs

✓ Java Object Class

- Java OOPs Concepts
- Naming Convention
- Object and Class
- static keyword
- this keyword

✓ Java Inheritance

- Inheritance (IS-A)
- Aggregation (HAS-A)

✓ Java Polymorphism

- Method Overloading
- Covariant Return Type
- super keyword
- Instance Initializer block
- final keyword
- Runtime Polymorphism
- Dynamic Binding
- instanceof operator

✓ Java Abstraction

- Abstract class
- Interface
- Abstract vs Interface

✓ Java Encapsulation

- Package
- Access Modifiers
- Encapsulation

✓ Java Array

- Java Array

✓ Java OOPs Misc

- Object class
- Object Cloning
- Math class
- Wrapper Class
- Java Recursion
- Call By Value
- strictfp keyword
- javacdoc tool
- Command Line Arg
- Object vs Class
- Overloading vs Overriding

✓ Java String

✓ Java Regex

✓ Exception Handling

✓ Java Inner classes

✓ Java Multithreading

✓ Java I/O

✓ Java Networking

✓ Java AWT & Events

✓ Java Swing

✓ JavaFX

✓ Java Applet

✓ Java Reflection

✓ Java Date

✓ Java Conversion

✓ Java Collection

✓ Java JDBC

✓ Java Misc

✓ Java New Features

✓ RMI

✓ Internationalization

✓ Interview Questions

AMCAT Test For Fresher Hiring

Get Started

Features of Java

12

1 Object Oriented

2 Simple

3 Secured

4 Platform Independent

5 Robust

6 Portable

7 Architecture Neutral

8 Dynamic

9 Interpreted

10 High Performance

11 Multithreaded

12 Distributed

Features of Java - Javatpoint

Features of Java

The features of Java are also known as java buzzwords.

A list of most important features of Java language is given below.

1 ) Simple

2 ) Object-Oriented

3 ) Portable

4 ) Platform Independent

5 ) Secured

6 ) Robust

7 ) Architecture Neutral

8 ) Dynamic

9 ) Interpreted

10 ) High Performance

11 ) Multithreaded

12 ) Distributed

Simple

Java is very easy to learn, and its syntax is simple, clean and easy to understand. According to Sun, Java language is a simple programming language because:

Are you a Doer + Innovator?

Learn more

- Java syntax is based on C++ (so easier for programmers to learn it after C++).
- Java has removed many complicated and rarely-used features, for example, explicit pointers, operator overloading, etc.
- There is no need to remove unreferenced objects because there is an Automatic Garbage Collection in Java.

Object-oriented

Java is an object-oriented programming language. Everything in Java is an object. Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behavior.

Object-oriented programming (OOPs) is a methodology that simplifies software development and maintenance by providing some rules.

Basic concepts of OOPs are:

1. Object

2. Class

3. Inheritance

4. Polymorphism

5. Abstraction

6. Encapsulation

Be a Certified CLOUD EXPERT

Accredited By IBM, MICROSOFT

ENROLL NOW

splunk> G G python aws

Platform Independent

Java is platform independent because it is different from other languages like C, C++, etc. which are compiled into platform specific machines while Java is a write once, run anywhere language. A platform is the hardware or software environment in which a program runs.

There are two types of platforms software-based and hardware-based. Java provides a software-based platform.

The Java platform differs from most other platforms in the sense that it is a software-based platform that runs on the top of other hardware-based platforms. It has two components:

1. Runtime Environment

2. API(Application Programming Interface)

Java code can be run on multiple platforms, for example, Windows, Linux, Sun Solaris, Mac/OS, etc. Java code is compiled by the compiler and converted into bytecode. This bytecode is a platform-independent code because it can be run on multiple platforms, i.e., Write Once and Run Anywhere(WORA).

Class File

Mac/OS JVM

windows JVM

Linux JVM

Apple

Windows

Penguin

Secured

Java is best known for its security. With Java, we can develop virus-free systems. Java is secured because:

- No explicit pointer
- Java Programs run inside a virtual machine sandbox

Uses Runtime Environment of operating system

C++ Application

OS

Uses Runtime Environment of its own

JAVA Application

JVM

OS

- Classloader: Classloader in Java is a part of the Java Runtime Environment(JRE) which is used to load Java classes into the Java Virtual Machine dynamically. It adds security by separating the package for the classes of the local file system from those that are imported from network sources.
- Bytecode Verifier: It checks the code fragments for illegal code that can violate access right to objects.
- Security Manager: It determines what resources a class can access such as reading and writing to the local disk.

Java language provides these securities by default. Some security can also be provided by an application developer explicitly through SSL, JAAS, Cryptography, etc.

Robust

Robust simply means strong. Java is robust because:

- It uses strong memory management.
- There is a lack of pointers that avoids security problems.
- There is automatic garbage collection in Java which runs on the Java Virtual Machine to get rid of objects which are not being used by a Java application anymore.
- There are exception handling and the type checking mechanism in Java. All these points make Java robust.

Architecture-neutral

Java is architecture neutral because there are no implementation dependent features, for example, the size of primitive types is fixed.

In C programming, int data type occupies 2 bytes of memory for 32-bit architecture and 4 bytes of memory for 64-bit architecture. However, it occupies 4 bytes of memory for both 32 and 64-bit architectures in Java.

Portable

Java is portable because it facilitates you to carry the Java bytecode to any platform. It doesn't require any implementation.

High-performance

Java is faster than other traditional interpreted programming languages because Java bytecode is "close" to native code. It is still a little bit slower than a compiled language (e.g., C++). Java is an interpreted language that is why it is slower than compiled languages, e.g., C, C++, etc.

Distributed

Java is distributed because it facilitates users to create distributed applications in Java. RMI and EJB are used for creating distributed applications. This feature of Java makes us able to access files by calling the methods from any machine on the internet.

Multi-threaded

A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multi-media, Web applications, etc.

Dynamic

Java is a dynamic language. It supports dynamic loading of classes. It means classes are loaded on demand. It also supports functions from its native languages, i.e., C and C++.

Java supports dynamic compilation and automatic memory management (garbage collection).

Next Topic

C++ vs Java

prev

next

Now almost every company is shortlisting Sravan

Where are you?

Book Now

amcat

Please Share

f t g+ p

Join Javatpoint Test Series

Placement Papers

TCS

HCL

Infosys

IBM

Accenture

AMCAT

eLitimas

Java

Python

C Programming

Networking

Bank PO/Clerk

UPSSSC

Government Exams

SSC

Civil Services

SBI

GATE

NEET

CAT

Railway

CTET

ITT JEE

Learn Latest Tutorials

NGINX

SSRS

Blockchain

ETL

Bugzilla

Hive

Jenkins

Pytorch

QA

Agile

JIRA

Tableau

Preparation

Aptitude

Reasoning

Verbal A.

Interview

Company

Trending Technologies

AI

AWS

Selenium

IoT

Cloud

Hadoop

ReactJS

React Native

Node.js

D. Science

Angular 7

B.Tech / MCA

DBMS

DS

DAA

OS

C. Network

Compiler D.

COA

D. Math.

E. Hacking

C. Graphics

Software E.

Web Tech.

Cyber Sec.

Automata

C

C++

Java

.NET

Python

Programs

Control S.

Grammar & Spelling Check

grammarly.com

Write better online and avoid embarrassing mistakes

OPEN

THE MINX & THE MAESTRO

WATCH "SHARPEST KNIFE" TODAY

Play Video

Ad by REVENUEUNIT

Be a Certified CLOUD EXPERT

Accredited By IBM, MICROSOFT

ENROLL NOW

splunk> G G python aws

AMCAT Test For Fresher Hiring

Get Started

Features of Java

12

1 Object Oriented

2 Simple

3 Secured

4 Platform Independent

5 Robust

6 Portable

7 Architecture Neutral

8 Dynamic

9 Interpreted

10 High Performance

11 Multithreaded

12 Distributed

Features of Java - Javatpoint

Features of Java

The features of Java are also known as java buzzwords.

A list of most important features of Java language is given below.

1 ) Simple

2 ) Object-Oriented

3 ) Portable

4 ) Platform Independent

5 ) Secured

6 ) Robust

7 ) Architecture Neutral

8 ) Dynamic

9 ) Interpreted

10 ) High Performance

11 ) Multithreaded

12 ) Distributed

Simple

Java is very easy to learn, and its syntax is simple, clean and easy to understand. According to Sun, Java language is a simple programming language because:

Are you a Doer + Innovator?

Learn more

- Java syntax is based on C++ (so easier for programmers to learn it after C++).
- Java has removed many complicated and rarely-used features, for example, explicit pointers, operator overloading, etc.
- There is no need to remove unreferenced objects because there is an Automatic Garbage Collection in Java.

Object-oriented

Java is an object-oriented programming language. Everything in Java is an object. Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behavior.

Object-oriented programming (OOPs) is a methodology that simplifies software development and maintenance by providing some rules.

Basic concepts of OOPs are:

1. Object

2. Class

3. Inheritance

4. Polymorphism

5. Abstraction

6. Encapsulation

Be a Certified CLOUD EXPERT

Accredited By IBM, MICROSOFT

ENROLL NOW

splunk> G G python aws

Platform Independent

Java is platform independent because it is different from other languages like C, C++, etc. which are compiled into platform specific machines while Java is a write once, run anywhere language. A platform is the hardware or software environment in which a program runs.

There are two types of platforms software-based and hardware-based. Java provides a software-based platform.

The Java platform differs from most other platforms in the sense that it is a software-based platform that runs on the top of other hardware-based platforms. It has two components:

1. Runtime Environment

2. API(Application Programming Interface)

Java code can be run on multiple platforms, for example, Windows, Linux, Sun Solaris, Mac/OS, etc. Java code is compiled by the compiler and converted into bytecode. This bytecode is a platform-independent code because it can be run on multiple platforms, i.e., Write Once and Run Anywhere(WORA).

Class File

Mac/OS JVM

windows JVM

Linux JVM

Apple

Windows

Penguin

Secured

Java is best known for its security. With Java, we can develop virus-free systems. Java is secured because:

- No explicit pointer
- Java Programs run inside a virtual machine sandbox

Uses Runtime Environment of operating system

C++ Application

OS

Uses Runtime Environment of its own

JAVA Application

JVM

OS

- Classloader: Classloader in Java is a part of the Java Runtime Environment(JRE) which is used to load Java classes into the Java Virtual Machine dynamically. It adds security by separating the package for the classes of the local file system from those that are imported from network sources.
- Bytecode Verifier: It checks the code fragments for illegal code that can violate access right to objects.
- Security Manager: It determines what resources a class can access such as reading and writing to the local disk.

Java language provides these securities by default. Some security can also be provided by an application developer explicitly through SSL, JAAS, Cryptography, etc.

Robust

Robust simply means strong. Java is robust because:

- It uses strong memory management.
- There is a lack of pointers that avoids security problems.
- There is automatic garbage collection in Java which runs on the Java Virtual Machine to get rid of objects which are not being used by a Java application anymore.
- There are exception handling and the type checking mechanism in Java. All these points make Java robust.

Architecture-neutral

Java is architecture neutral because there are no implementation dependent features, for example, the size of primitive types is fixed.

In C programming, int data type occupies 2 bytes of memory for 32-bit architecture and 4 bytes of memory for 64-bit architecture. However, it occupies 4 bytes of memory for both 32 and 64-bit architectures in Java.

Portable

Java is portable because it facilitates you to carry the Java bytecode to any platform. It doesn't require any implementation.

High-performance

Java is faster than other traditional interpreted programming languages because Java bytecode is "close" to native code. It is still a little bit slower than a compiled language (e.g., C++). Java is an interpreted language that is why it is slower than compiled languages, e.g., C, C++, etc.

Distributed

Java is distributed because it facilitates users to create distributed applications in Java. RMI and EJB are used for creating distributed applications. This feature of Java makes us able to access files by calling the methods from any machine on the internet.

Multi-threaded

A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multi-media, Web applications, etc.

Dynamic

Java is a dynamic language. It supports dynamic loading of classes. It means classes are loaded on demand. It also supports functions from its native languages, i.e., C and C++.

Java supports dynamic compilation and automatic memory management (garbage collection).

Next Topic

C++ vs Java

prev

next

Now almost every company is shortlisting Sravan

Where are you?

Book Now

amcat

Please Share

f t g+ p

Join Javatpoint Test Series

Placement Papers

TCS

HCL

Infosys

IBM

Accenture

AMCAT

eLitimas

Java

Python

C Programming

Networking

Bank PO/Clerk

UPSSSC

Government Exams

SSC

Civil Services

SBI

GATE

NEET

CAT

Railway

CTET

ITT JEE

Learn Latest Tutorials

NGINX

SSRS

Blockchain

ETL

Bugzilla

Hive

Jenkins

Pytorch

QA

Agile

JIRA

Tableau

Preparation

Aptitude

Reasoning

Verbal A.

Interview

Company

Trending Technologies

AI

AWS

Selenium

IoT

Cloud

Hadoop

ReactJS

React Native

Node.js

D. Science

Angular 7

B.Tech / MCA

DBMS

DS

DAA

OS

C. Network

Compiler D.

COA

D. Math.

E. Hacking

C. Graphics

Software E.

Web Tech.

Cyber Sec.

Automata

C

C++

Java

.NET

Python

Programs

Control S.

Grammar & Spelling Check

grammarly.com

Write better online and avoid embarrassing mistakes

OPEN

AMCAT Test For Fresher Hiring

Get Started

Features of Java

12

1 Object Oriented

2 Simple

3 Secured

4 Platform Independent

5 Robust

6 Portable

7 Architecture Neutral

8 Dynamic

9 Interpreted

10 High Performance

11 Multithreaded

12 Distributed

Features of Java - Javatpoint

Features of Java

The features of Java are also known as java buzzwords.

A list of most important features of Java language is given below.

1 ) Simple

2 ) Object-Oriented

3 ) Portable

4 ) Platform Independent

5 ) Secured

6 ) Robust

7 ) Architecture Neutral

8 ) Dynamic

9 ) Interpreted

10 ) High Performance

11 ) Multithreaded

12 ) Distributed

Simple

Java is very easy to learn, and its syntax is simple, clean and easy to understand. According to Sun, Java language is a simple programming language because:

Are you a Doer + Innovator?

Learn more

- Java syntax is based on C++ (so easier for programmers to learn it after C++).
- Java has removed many complicated and rarely-used features, for example, explicit pointers, operator overloading, etc.
- There is no need to remove unreferenced objects because there is an Automatic Garbage Collection in Java.

Object-oriented

Java is an object-oriented programming language. Everything in Java is an object. Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behavior.

Object-oriented programming (OOPs) is a methodology that simplifies software development and maintenance by providing some rules.

Basic concepts of OOPs are:

1. Object

2. Class

3. Inheritance

4. Polymorphism

5. Abstraction

6. Encapsulation

Be a Certified CLOUD EXPERT

Accredited By IBM, MICROSOFT

ENROLL NOW

splunk> G G python aws

Platform Independent

Java is platform independent because it is different from other languages like C, C++, etc. which are compiled into platform specific machines while Java is a write once, run anywhere language. A platform is the hardware or software environment in which a program runs.

There are two types of platforms software-based and hardware-based. Java provides a software-based platform.

The Java platform differs from most other platforms in the sense that it is a software-based platform that runs on the top of other hardware-based platforms. It has two components:

1. Runtime Environment

2. API(Application Programming Interface)

Java code can be run on multiple platforms, for example, Windows, Linux, Sun Solaris, Mac/OS, etc. Java code is compiled by the compiler and converted into bytecode. This bytecode is a platform-independent code because it can be run on multiple platforms, i.e., Write Once and Run Anywhere(WORA).

Class File

Mac/OS JVM

windows JVM

Linux JVM

Apple

Windows

Penguin

Secured

Java is best known for its security. With Java, we can develop virus-free systems. Java is secured because:

- No explicit pointer
- Java Programs run inside a virtual machine sandbox

Uses Runtime Environment of operating system

C++ Application

OS

Uses Runtime Environment of its own

JAVA Application

JVM

OS

- Classloader: Classloader in Java is a part of the Java Runtime Environment(JRE) which is used to load Java classes into the Java Virtual Machine dynamically. It adds security by separating the package for the classes of the local file system from those that are imported from network sources.
- Bytecode Verifier: It checks the code fragments for illegal code that can violate access right to objects.
- Security Manager: It determines what resources a class can access such as reading and writing to the local disk.

Java language provides these securities by default. Some security can also be provided by an application developer explicitly through SSL, JAAS, Cryptography, etc.

Robust

Robust simply means strong. Java is robust because:

- It uses strong memory management.
- There is a lack of pointers that avoids security problems.
- There is automatic garbage collection in Java which runs on the Java Virtual Machine to get rid of objects which are not being used by a Java application anymore.
- There are exception handling and the type checking mechanism in Java. All these points make Java robust.

Architecture-neutral

Java is architecture neutral because there are no implementation dependent features, for example, the size of primitive types is fixed.

In C programming, int data type occupies 2 bytes of memory for 32-bit architecture and 4 bytes of memory for 64-bit architecture. However, it occupies 4 bytes of memory for both 32 and 64-bit architectures in Java.

Portable

Java is portable because it facilitates you to carry the Java bytecode to any platform. It doesn't require any implementation.

High-performance

Java is faster than other traditional interpreted programming languages because Java bytecode is "close" to native code. It is still a little bit slower than a compiled language (e.g., C++). Java is an interpreted language that is why it is slower than compiled languages, e.g., C, C++, etc.

Distributed

Java is distributed because it facilitates users to create distributed applications in Java. RMI and EJB are used for creating distributed applications. This feature of Java makes us able to access files by calling the methods from any machine on the internet.

Multi-threaded

A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multi-media, Web applications, etc.

Dynamic

Java is a dynamic language. It supports dynamic loading of classes. It means classes are loaded on demand. It also supports functions from its native languages, i.e., C and C++.

Java supports dynamic compilation and automatic memory management (garbage collection).

Next Topic

C++ vs Java

prev

next

Now almost every company is shortlisting Sravan

Where are you?

Book Now

amcat

Please Share

f t g+ p

Join Javatpoint Test Series

Placement Papers

TCS

HCL

Infosys

IBM

Accenture

AMCAT

eLitimas

Java

Python

C Programming

Networking

Bank PO/Clerk

UPSSSC

Government Exams

SSC

Civil Services

SBI

GATE

NEET

CAT

Railway

CTET

ITT JEE

Learn Latest Tutorials

NGINX

SSRS

Blockchain

ETL

Bugzilla

Hive

Jenkins

Pytorch

QA

Agile

JIRA

Tableau

Preparation

Aptitude

Reasoning

Verbal A.

Interview

Company

Trending Technologies

AI

AWS

Selenium

IoT

Cloud

Hadoop

ReactJS

React Native

Node.js

D. Science

Angular 7

B.Tech / MCA

DBMS

DS

DAA

OS

C. Network

Compiler D.

COA

D. Math.

E. Hacking

C. Graphics

Software E.

Web Tech.

Cyber Sec.

Automata

C

C++

Java

.NET

Python

Programs

Control S.

Grammar & Spelling Check

grammarly.com

Write better online and avoid embarrassing mistakes

OPEN

AMCAT Test For Fresher Hiring

Get Started

Features of Java

12

1 Object Oriented

2 Simple

3 Secured

4 Platform Independent

5 Robust

6 Portable

7 Architecture Neutral

8 Dynamic

9 Interpreted

10 High Performance

11 Multithreaded

12 Distributed

Features of Java - Javatpoint

Features of Java

The features of Java are also known as java buzzwords.

A list of most important features of Java language is given below.

1 ) Simple

2 ) Object-Oriented

3 ) Portable

4 ) Platform Independent

5 ) Secured

6 ) Robust

7 ) Architecture Neutral

8 ) Dynamic

9 ) Interpreted

10 ) High Performance

11 ) Multithreaded

12 ) Distributed

Simple

Java is very easy to learn, and its syntax is simple, clean and easy to understand. According to Sun, Java language is a simple programming language because:

Are you a Doer + Innovator?

Learn more

- Java syntax is based on C++ (so easier for programmers to learn it after C++).
- Java has removed many complicated and rarely-used features, for example, explicit pointers, operator overloading, etc.
- There is no need to remove unreferenced objects because there is an Automatic Garbage Collection in Java.

Object-oriented

Java is an object-oriented programming language. Everything in Java is an object. Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behavior.

Object-oriented programming (OOPs) is a methodology that simplifies software development and maintenance by providing some rules.

Basic concepts of OOPs are:

1. Object

2. Class

3. Inheritance

4. Polymorphism

5. Abstraction

6. Encapsulation

Be a Certified CLOUD EXPERT

Accredited By IBM, MICROSOFT

ENROLL NOW

splunk> G G python aws

Platform Independent

Java is platform independent because it is different from other languages like C, C++, etc. which are compiled into platform specific machines while Java is a write once, run anywhere language. A platform is the hardware or software environment in which a program runs.

There are two types of platforms software-based and hardware-based. Java provides a software-based platform.

The Java platform differs from most other platforms in the sense that it is a software-based platform that runs on the top of other hardware-based platforms. It has two components:

1. Runtime Environment

2. API(Application Programming Interface)

Java code can be run on multiple platforms, for example, Windows, Linux, Sun Solaris, Mac/OS, etc. Java code is compiled by the compiler and converted into bytecode. This bytecode is a platform-independent code because it can be run on multiple platforms, i.e., Write Once and Run Anywhere(WORA).

Class File

Mac/OS JVM

windows JVM

Linux JVM

Apple

Windows

Penguin

Secured

Java is best known for its security. With Java, we can develop virus-free systems. Java is secured because:

- No explicit pointer
- Java Programs run inside a virtual machine sandbox

Uses Runtime Environment of operating system

C++ Application

OS

Uses Runtime Environment of its own

JAVA Application

JVM

OS

- Classloader: Classloader in Java is a part of the Java Runtime Environment(JRE) which is used to load Java classes into the Java Virtual Machine dynamically. It adds security by separating the package for the classes of the local file system from those that are imported from network sources.
- Bytecode Verifier: It checks the code fragments for illegal code that can violate access right to objects.
- Security Manager: It determines what resources a class can access such as reading and writing to the local disk.

Java language provides these securities by default. Some security can also be provided by an application developer explicitly through SSL, JAAS, Cryptography, etc.

Robust

Robust simply means strong. Java is robust because:

- It uses strong memory management.
- There is a lack of pointers that avoids security problems.
- There is automatic garbage collection in Java which runs on the Java Virtual Machine to get rid of objects which are not being used by a Java application anymore.
- There are exception handling and the type checking mechanism in Java. All these points make Java robust.

Architecture-neutral

Java is architecture neutral because there are no implementation dependent features, for example, the size of primitive types is fixed.

In C programming, int data type occupies 2 bytes of memory for 32-bit architecture and 4 bytes of memory for 64-bit architecture. However, it occupies 4 bytes of memory for both 32 and 64-bit architectures in Java.

Portable

Java is portable because it facilitates you to carry the Java bytecode to any platform. It doesn't require any implementation.

High-performance

Java is faster than other traditional interpreted programming languages because Java bytecode is "close" to native code. It is still a little bit slower than a compiled language (e.g., C++). Java is an interpreted language that is why it is slower than compiled languages, e.g., C, C++, etc.

Distributed

Java is distributed because it facilitates users to create distributed applications in Java. RMI and EJB are used for creating distributed applications. This feature of Java makes us able to access files by calling the methods from any machine on the internet.

Multi-threaded

A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multi-media, Web applications, etc.

Dynamic

Java is a dynamic language. It supports dynamic loading of classes. It means classes are loaded on demand. It also supports functions from its native languages, i.e., C and C++.

Java supports dynamic compilation and automatic memory management (garbage collection).

Next Topic

C++ vs Java

prev

next

Now almost every company is shortlisting Sravan

Where are you?

Book Now

amcat

Please Share

f t g+ p

Join Javatpoint Test Series

Placement Papers

TCS

HCL

Infosys

IBM

Accenture

AMCAT

eLitimas

Java

Python

C Programming

Networking

Bank PO/Clerk

UPSSSC

Government Exams

SSC

Civil Services

SBI

GATE

NEET

CAT

Railway

CTET

ITT JEE

Learn Latest Tutorials

NGINX

SSRS

Blockchain

ETL

Bugzilla

Hive

Jenkins

Pytorch

QA

Agile

JIRA

Tableau

Preparation

Aptitude

Reasoning

Verbal A.

Interview

Company

Trending Technologies

AI

AWS

Selenium

IoT

Cloud

Hadoop

ReactJS

React Native

Node.js

D. Science

Angular 7

B.Tech / MCA

DBMS

DS

DAA

OS

C. Network

Compiler D.

COA

D. Math.

E. Hacking

C. Graphics

Software E.

Web Tech.

Cyber Sec.

Automata

C

C++

Java

.NET

Python

Programs

Control S.

Grammar & Spelling Check

grammarly.com

Write better online and avoid embarrassing mistakes

OPEN

AMCAT Test For Fresher Hiring

Get Started

Features of Java

12

1 Object Oriented

2 Simple

3 Secured

4 Platform Independent

5 Robust

6 Portable

7 Architecture Neutral

8 Dynamic

9 Interpreted

10 High Performance

11 Multithreaded

12 Distributed

Features of Java - Javatpoint

Features of Java

The features of Java are also known as java buzzwords.

A list of most important features of Java language is given below.

1 ) Simple

2 ) Object-Oriented

3 ) Portable

4 ) Platform Independent

5 ) Secured

6 ) Robust

7 ) Architecture Neutral

8 ) Dynamic

9 ) Interpreted

10 ) High Performance

11 ) Multithreaded

12 ) Distributed

Simple

Java is very easy to learn, and its syntax is simple, clean and easy to understand. According to Sun, Java language is a simple programming language because:

Are you a Doer + Innovator?

Learn more

- Java syntax is based on C++ (so easier for programmers to learn it after C++).
- Java has removed many complicated and rarely-used features, for example, explicit pointers, operator overloading, etc.
- There is no need to remove unreferenced objects because there is an Automatic Garbage Collection in Java.

Object-oriented

Java is an object-oriented programming language. Everything in Java is an object. Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behavior.

Object-oriented programming (OOPs) is a methodology that simplifies software development and maintenance by providing some rules.

Basic concepts of OOPs are:

1. Object

2. Class

3. Inheritance

4. Polymorphism

5. Abstraction

6. Encapsulation

Be a Certified CLOUD EXPERT

Accredited By IBM, MICROSOFT

ENROLL NOW

splunk> G G python aws

Platform Independent

Java is platform independent because it is different from other languages like C, C++, etc. which are compiled into platform specific machines while Java is a write once, run anywhere language. A platform is the hardware or software environment in which a program runs.

There are two types of platforms software-based and hardware-based. Java provides a software-based platform.

The Java platform differs from most other platforms in the sense that it is a software-based platform that runs on the top of other hardware-based platforms. It has two components:

1. Runtime Environment

2. API(Application Programming Interface)

Java code can be run on multiple platforms, for example, Windows, Linux, Sun Solaris, Mac/OS, etc. Java code is compiled by the compiler and converted into bytecode. This bytecode is a platform-independent code because it can be run on multiple platforms, i.e., Write Once and Run Anywhere(WORA).

Class File

Mac/OS JVM

windows JVM

Linux JVM

Apple

Windows

Penguin

Secured

Java is best known for its security. With Java, we can develop virus-free systems. Java is secured because:

- No explicit pointer
- Java Programs run inside a virtual machine sandbox

Uses Runtime Environment of operating system

C++ Application

OS

Uses Runtime Environment of its own

JAVA Application

JVM

OS

- Classloader: Classloader in Java is a part of the Java Runtime Environment(JRE) which is used to load Java classes into the Java Virtual Machine dynamically. It adds security by separating the package for the classes of the local file system from those that are imported from network sources.
- Bytecode Verifier: It checks the code fragments for illegal code that can violate access right to objects.
- Security Manager: It determines what resources a class can access such as reading and writing to the local disk.

Java language provides these securities by default. Some security can also be provided by an application developer explicitly through SSL, JAAS, Cryptography, etc.

Robust

Robust simply means strong. Java is robust because:

- It uses strong memory management.
- There is a lack of pointers that avoids security problems.
- There is automatic garbage collection in Java which runs on the Java Virtual Machine to get rid of objects which are not being used by a Java application anymore.
- There are exception handling and the type checking mechanism in Java. All these points make Java robust.

Architecture-neutral

Java is architecture neutral because there are no implementation dependent features, for example, the size of primitive types is fixed.

In C programming, int data type occupies 2 bytes of memory for 32-bit architecture and 4 bytes of memory for 64-bit architecture. However, it occupies 4 bytes of memory for both 32 and 64-bit architectures in Java.

Portable

Java is portable because it facilitates you to carry the Java bytecode to any platform. It doesn't require any implementation.

High-performance

Java is faster than other traditional interpreted programming languages because Java bytecode is "close" to native code. It is still a little bit slower than a compiled language (e.g., C++). Java is an interpreted language that is why it is slower than compiled languages, e.g., C, C++, etc.

Distributed</