



[Home](#) » [Java Basic](#) » Sự khác nhau giữa JDK, JRE và JVM

Sự khác nhau giữa JDK, JRE và JVM

Posted on Tháng Mười Một 18, 2017

Sự khác nhau giữa JDK, JRE và JVM. JDK, JRE và JVM là gì? phân biệt chúng như nào?

1. JVM

JVM (Java Virtual Machine): là máy ảo Java. Nó được dùng để thực thi các chương trình Java.

Mỗi nền tảng/hệ điều hành khác nhau (Windows, IOS, Linux...) lại có một loại JVM khác nhau. Hiểu nôm na thì các chương trình Java của các bạn chạy trên JVM. Nói Java đa nền tảng, thực chất thì nó được hỗ trợ JVM trên nhiều nền tảng. Chương trình Java chạy được trên Window/Linux/IOS vì nó có JVM chạy được trên các nền tảng đó.

Giả sử bây giờ có 1 hệ điều hành mới, thì người ta phải xây dựng JVM cho hệ điều hành đó trước thì chương trình Java mới có thể chạy được trên hệ điều hành đó.

Vậy JVM thực hiện những gì khi chạy chương trình Java?

JVM thực hiện các công việc chính sau đây:

- Tải code (các class, resource)
- Kiểm tra code (kiểm tra code có đúng cú pháp không, có bị lỗi không, tất nhiên nếu code có lỗi thì sẽ không chạy được chương trình rồi)
- Thực thi code
- Cung cấp môi trường runtime

Ví dụ: tôi có class Hello.java

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.print("Hello world!");  
    }  
}
```

Muốn chạy được class này cần phải biên dịch nó thành file Hello.class bằng công cụ javac hoặc các IDE như Eclipse, Netbean vì JVM chỉ có thể hiểu được file .class.

JVM sẽ đọc file .class và kiểm tra code -> thực thi code (JVM sẽ thực hiện thông dịch file .class thành ngôn ngữ mà hệ



StackJava
1.1k likes



Like Page



Apache



hibernate

Be the first of your friends to like this

Tìm kiếm ...

QC ▼

SWINBURNE
UNIVERSITY
TECHNOLOGY

Partner with  PPT Education

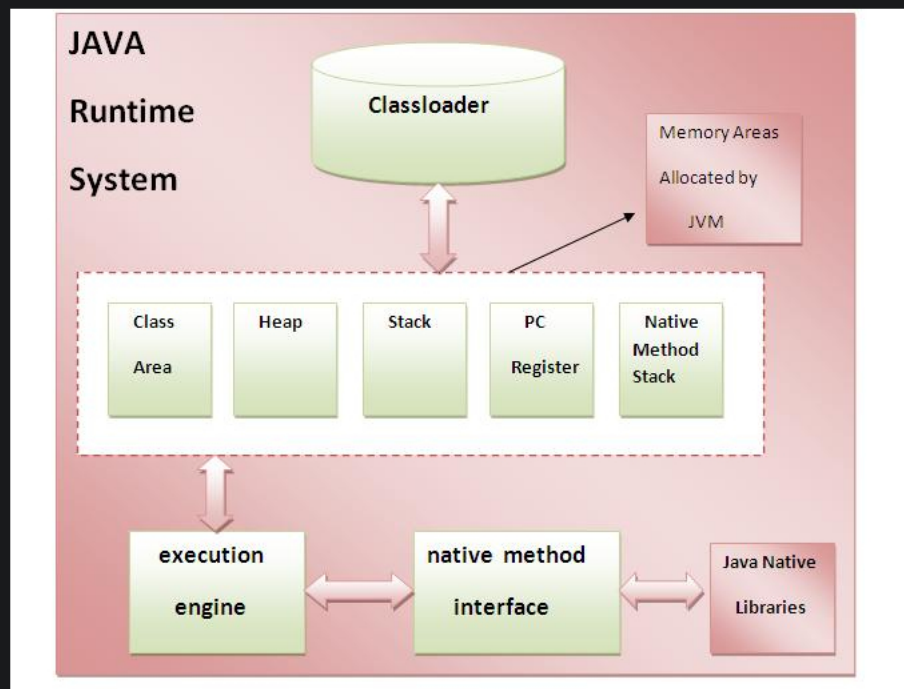
SWINBURNE VIỆT NAM
XÉT TUYỂN ĐẠI HỌC
ĐĂNG KÝ NGAY
BẰNG QUỐC TẾ CẤP BỞI
SWINBURNE AUSTRALIA
TOP 400 ĐH HÀNG ĐẦU THẾ GIỚI



JAVA BASIC,
UNCATEGORIZED

điều hành có thể hiểu được)

Cấu trúc của JVM:



– Classloader

Classloader là 1 hệ thống con của JVM, nó được dùng để tải các file *.class

– Class(Method) Area

Class(Method) Area lưu trữ cấu trúc các class như constant pool, field, method, data...

– Heap

Heap là vùng dữ liệu runtime bao gồm các object được cấp phát

– Stack

Stack chứa các biến

– Program Counter Register

PC (program counter) register: bao gồm địa chỉ của các lệnh máy ảo java đang được thực thi

– Native Method Stack

Bao gồm tất cả các native method được sử dụng trong ứng dụng (native method là các method viết bằng ngôn ngữ không phải Java, ví dụ bạn vẫn có thể sử dụng một số thư viện của C trong Java)

– Execution Engine

Cron expression là gì? Hướng dẫn cú pháp cron expression

Monstache là gì? Đồng bộ mongodb sang Elasticsearch với Monstache

Tạo ứng dụng facebook để đăng nhập thay tài khoản

Sự khác nhau giữa định dạng file zip và rar

Go, Golang là gì? Cài đặt Golang trên Ubuntu

AOP là gì? Aspect Oriented Programming trong Java

Các plugin hay dùng trong eclipse

Các bài tập Java phân lớp – hướng đối tượng

Bài tập Java phần xử lý ký tự, code ví dụ

Các bài tập Java cơ bản, code ví dụ

ProcessBuilder trong Java, chạy lệnh cmd trong Java

ORM là gì? Tổng quan về ORM Framework

Sự khác nhau giữa String, StringBuffer, StringBuilder?

JPA là gì? Sự khác nhau giữa JPA với Hibernate

Sự khác nhau giữa JDK, JRE và JVM

Một số câu hỏi hay gặp khi phỏng vấn vị trí lập trình Java

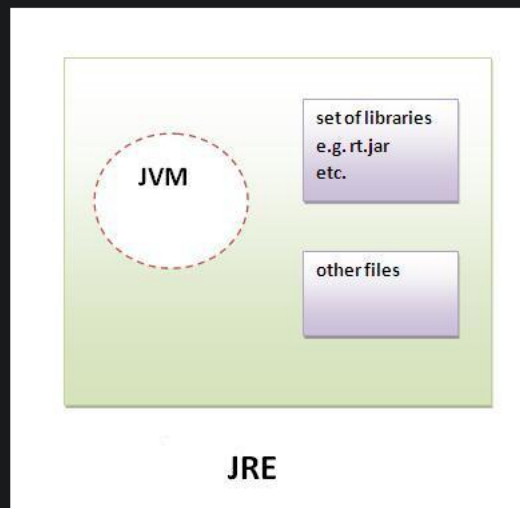
Lập trình cấu trúc là gì? Ưu nhược điểm.

Execution Engine

Bao gồm 1 vi xử lý ảo; 1 trình thông dịch để đọc các byte code và thực thi chúng; trình compiler Just-In-Time(JIT): dùng để cải thiện hiệu năng.

2. JRE

JRE (Java Runtime Environment): Môi trường chạy Java gồm các thư viện và các file được sử dụng khi runtime



3. JDK

JDK (Java Development Kit): bộ công cụ phát triển Java chứa cả JRE và JVM, đây là phần lõi của Môi trường Java và cung cấp tất cả các công cụ, thực thi chương trình, biên dịch file, cho phép debug... Tóm lại, JDK là 1 nền tảng, đó là lý do tại sao chúng ta chia ra các bộ cài đặt JDK trên các hệ điều hành khác nhau như window, Mac, Unix...

Kết luận:

- JDK là dùng cho mục đích phát triển (lập trình, debug), nếu bạn chỉ cần chạy chương trình Java thì ko nhất thiết phải cài JDK mà chỉ cần cài JRE
- JRE là dùng cho chạy chương trình Java, JDK và JRE đều chứa JVM
- JVM là trái tim của ngôn ngữ của Java và được cung cấp tùy thuộc vào nền tảng/hệ điều hành mà nó chạy.

References:

<https://www.javatpoint.com/internal-details-of-jvm>

<https://dzone.com/articles/jvm-architecture-explained>

<http://www.oracle.com/technetwork/java/javase/tech/index-jsp-140763.html>

This entry was posted in *Java Basic, Uncategorized* and tagged *basic, java*. Bookmark the *permalink*.

[← Functional Interface là gì? Functional Interface API trong Java 8](#)

[JPA là gì? Sự khác nhau giữa JPA với Hibernate →](#)

[Phân biệt sự khác nhau giữa worksheet và workbook – Excel](#)

[Tự động bật ứng dụng lúc khởi động Windows 10](#)

[Phân biệt giữa Java ME, Java SE và Java EE](#)

[So sánh sự khác nhau giữa J2ME, J2SE và J2EE](#)

[HTTP Status code là gì? Các loại http status code](#)

CHUYÊN MỤC

[Algorithm](#)

[Apache](#)

[Apache JMeter](#)

[Apache Kafka](#)

[AWS](#)

[C/C++](#)

[CDI](#)

[Clean Code](#)

[Demo](#)

[Design Pattern](#)

[Docker](#)

[Eclipse](#)

[Elasticsearch](#)

[Excel](#)

[FAQ](#)

[Framework](#)

[Freemaker](#)

[FreeMarker](#)

[Gantt](#)

Gradle

Hibernate

HttpClient

HttpComponents

Install

IntelliJ IDEA

Java

Java Basic

Java Core

Java8

JavaScript

jooq

JSF

JSP-Servlet

JUnit

Library

Linux

Maven

MongoDB

MySQL

Network Programming

Node.js

OOP

PostgreSQL

PrimeFaces

Principle

Python

quartz

Redis

SDKMan

Security

SocketCluster

Spring

Spring Boot

Spring Core

Spring Data

Spring Hibernate

Spring JDBC

Spring MVC

Spring Security

Thymeleaf

Tomcat

Uncategorized

Web Service

WebSocket

Wordpress

