|  |  |  |
| --- | --- | --- |
| Stt | Câu hỏi | Trả lời |
| 1 | Sự khác nhau giữa JDK,JRE và JVM? | |  |  |  |  | | --- | --- | --- | --- | |  | JDK | JRE | JVM | | Chức năng | Phát triển và chạy chương trình java | Chạy chương trình java | Chạy mã bytecode | | Thành phần | Trình biên dịch javac: Biên dịch mã Java (.java) thành bytecode (.class).  javadoc: Tạo tài liệu từ mã nguồn.  jdb: Công cụ gỡ lỗi.  JRE | JVM, Thư viện Java | Công cụ chuyển đổi bytecode | | Người sử dụng | nhà phát triển | Người dùng chạy ứng dụng | Chạy ứng dụng nội bộ | |
| 2 | Sự khác nhau giữa bộ nhớ heap và stack trong java? | |  |  |  | | --- | --- | --- | | Đặc điểm | Stack | Heap | | Mục đích | Lưu biến cục bộ, tham số hàm, lời gọi hàm  Nó hoạt động theo nguyên tắc LIFO (Last In, First Out). | Lưu đối tượng, biến toàn cục  Là một vùng nhớ động được quản lý linh hoạt | | Kích thước | Nhỏ hơn | Lớn hơn | | Tốc độ | Nhanh hơn | Chậm hơn | | Quản lý bộ nhớ | Tự động (phương thức kết thúc) | Dựa và Garbage Collector | | Phạm vi | Mỗi luồng có Stack riêng | Chia sẻ giữa tất cả các luồng | | Thời gian sống | Tồn tại trong thời gian phương thức thực thi | Tồn tại cho đến khi bị thu hồi bởi Garbage Collector | | Dữ liệu lưu trữ | Tham chiếu và biến cục bộ | Đối tượng và biến tĩnh |   Ví dụ:  public class MemoryExample {  public static void main(String[] args) {  int a = 10; // Lưu trong Stack  int b = 20; // Lưu trong Stack  MyClass obj = new MyClass(); // `obj` (tham chiếu) trong Stack, đối tượng ‘MyClass()’ trong Heap  obj.value = 30; // Dữ liệu của đối tượng trong Heap } } class MyClass { int value; // Lưu trong Heap } |
| 3 | Trình biên dịch JIT là gì? | [Trình biên dịch Just-In-Time (JIT) là một loại trình biên dịch động được sử dụng trong nhiều môi trường runtime hiện đại như .NET và Java1](https://kb.pavietnam.vn/kien-thuc-co-ban-ve-trinh-bien-dich-jit.html" \t "https://edgeservices.bing.com/edgesvc/_blank). [Thay vì biên dịch toàn bộ mã nguồn trước khi chạy chương trình, JIT biên dịch mã ngay khi cần thiết trong quá trình thực thi2](https://itigic.com/vi/just-in-time-jit-compile-how-it-works-in-programming/" \t "https://edgeservices.bing.com/edgesvc/_blank).  [Điều này giúp tối ưu hóa hiệu suất bằng cách chỉ biên dịch những phần mã thực sự cần thiết tại thời điểm đó](https://viblo.asia/p/trinh-bien-dich-javascript-jit-just-in-time-jvElaXRdZkw" \t "https://edgeservices.bing.com/edgesvc/_blank). [JIT kết hợp ưu điểm của cả trình biên dịch và trình thông dịch, giúp chương trình khởi động nhanh hơn và vẫn duy trì được hiệu suất cao trong quá trình chạy](https://viblo.asia/p/trinh-bien-dich-javascript-jit-just-in-time-jvElaXRdZkw" \t "https://edgeservices.bing.com/edgesvc/_blank). |
| \* | Phân biệt trình biên dịch và thông dịch | Trình biên dịch và trình thông dịch đều có nhiệm vụ chuyển đổi mã nguồn từ ngôn ngữ lập trình cấp cao sang mã máy, nhưng chúng hoạt động theo cách khác nhau: Trình biên dịch (Compiler)  * [Cách hoạt động: Chuyển đổi toàn bộ mã nguồn thành mã máy trước khi chương trình chạy](https://www.guru99.com/vi/difference-compiler-vs-interpreter.html" \t "https://edgeservices.bing.com/edgesvc/_blank). * [Tốc độ: Chương trình biên dịch chạy nhanh hơn vì mã đã được chuyển đổi hoàn toàn trước khi thực thi](https://www.guru99.com/vi/difference-compiler-vs-interpreter.html" \t "https://edgeservices.bing.com/edgesvc/_blank). * [Phát hiện lỗi: Hiển thị tất cả các lỗi sau khi quá trình biên dịch hoàn tất1](https://www.guru99.com/vi/difference-compiler-vs-interpreter.html" \t "https://edgeservices.bing.com/edgesvc/_blank). * [Bộ nhớ: Yêu cầu nhiều bộ nhớ hơn để lưu trữ mã đối tượng trung gian2](https://viblo.asia/p/su-khac-nhau-giua-trinh-bien-dich-va-trinh-thong-dich-Eb85oxG2K2G" \t "https://edgeservices.bing.com/edgesvc/_blank). * [Ngôn ngữ phổ biến: C, C++, C#, Java2](https://viblo.asia/p/su-khac-nhau-giua-trinh-bien-dich-va-trinh-thong-dich-Eb85oxG2K2G" \t "https://edgeservices.bing.com/edgesvc/_blank).  Trình thông dịch (Interpreter)  * [Cách hoạt động: Chuyển đổi từng dòng mã nguồn thành mã máy và thực thi ngay lập tức1](https://www.guru99.com/vi/difference-compiler-vs-interpreter.html" \t "https://edgeservices.bing.com/edgesvc/_blank). * [Tốc độ: Chạy chậm hơn vì mã được chuyển đổi và thực thi từng dòng một](https://www.guru99.com/vi/difference-compiler-vs-interpreter.html" \t "https://edgeservices.bing.com/edgesvc/_blank). * [Phát hiện lỗi: Hiển thị lỗi từng dòng một trong quá trình thực thi1](https://www.guru99.com/vi/difference-compiler-vs-interpreter.html" \t "https://edgeservices.bing.com/edgesvc/_blank). * [Bộ nhớ: Yêu cầu ít bộ nhớ hơn vì không tạo mã đối tượng trung gian](https://www.guru99.com/vi/difference-compiler-vs-interpreter.html" \t "https://edgeservices.bing.com/edgesvc/_blank)[2](https://viblo.asia/p/su-khac-nhau-giua-trinh-bien-dich-va-trinh-thong-dich-Eb85oxG2K2G" \t "https://edgeservices.bing.com/edgesvc/_blank). * [Ngôn ngữ phổ biến: Python, Ruby, PHP](https://www.guru99.com/vi/difference-compiler-vs-interpreter.html" \t "https://edgeservices.bing.com/edgesvc/_blank). |
| 4 | Platform là gì? | Platform (Nền tảng) là một môi trường hoặc hệ thống mà các ứng dụng, dịch vụ, hoặc sản phẩm khác có thể được xây dựng, triển khai và vận hành. Nó cung cấp các công cụ, dịch vụ, và cơ sở hạ tầng cần thiết để hỗ trợ sự phát triển và thực thi của các ứng dụng hoặc dịch vụ.  Các loại Platform chính:  Operating System (Hệ điều hành)  Application Platform (Nền tảng ứng dụng)  Cloud Platform (Nền tảng đám mây)  Web Platform (Nền tảng web)  Hardware Platform (Nền tảng phần cứng)  Mobile Platform (Nền tảng di động)  Social Media Platform (Nền tảng mạng xã hội)  Business Platform (Nền tảng kinh doanh) |
| 5 | Sự khác nhau giữa Java platform và các platform khác? | Java Platform có một số đặc điểm khác biệt so với các nền tảng khác: Java Platform  1. [Độc lập nền tảng: Java Platform dựa trên phần mềm, cho phép các ứng dụng Java chạy trên nhiều nền tảng phần cứng khác nhau nhờ vào Máy ảo Java (JVM) 1](https://quizlet.com/vn/964929770/cau-hoi-phong-van-java-flash-cards/" \t "https://edgeservices.bing.com/edgesvc/_blank). 2. [Bytecode: Mã nguồn Java được biên dịch thành bytecode, một ngôn ngữ trung gian, giúp nó có thể chạy trên bất kỳ nền tảng nào có JVM 1](https://quizlet.com/vn/964929770/cau-hoi-phong-van-java-flash-cards/" \t "https://edgeservices.bing.com/edgesvc/_blank). 3. [API phong phú: Java cung cấp một bộ API phong phú, hỗ trợ nhiều chức năng từ cơ bản đến nâng cao](https://quizlet.com/vn/964929770/cau-hoi-phong-van-java-flash-cards/" \t "https://edgeservices.bing.com/edgesvc/_blank) [2](https://tinasoft.io/vi/blogs/phan-biet-giua-java-me-java-se-java-ee-va-javafx/" \t "https://edgeservices.bing.com/edgesvc/_blank). 4. [Viết một lần, chạy mọi nơi: Tính chất “Write Once, Run Anywhere” giúp mã Java có thể chạy trên bất kỳ hệ điều hành nào mà không cần thay đổi 1](https://quizlet.com/vn/964929770/cau-hoi-phong-van-java-flash-cards/" \t "https://edgeservices.bing.com/edgesvc/_blank).  Các nền tảng khác  1. [Nền tảng cụ thể: Nhiều nền tảng khác như .NET, iOS, Android thường được thiết kế để chạy trên một hệ điều hành hoặc phần cứng cụ thể 1](https://quizlet.com/vn/964929770/cau-hoi-phong-van-java-flash-cards/" \t "https://edgeservices.bing.com/edgesvc/_blank). 2. [Ngôn ngữ biên dịch trực tiếp: Các nền tảng như C++ thường biên dịch mã nguồn trực tiếp thành mã máy, phụ thuộc vào hệ điều hành và phần cứng cụ thể 1](https://quizlet.com/vn/964929770/cau-hoi-phong-van-java-flash-cards/" \t "https://edgeservices.bing.com/edgesvc/_blank). 3. [API giới hạn: Một số nền tảng có thể có bộ API giới hạn hơn so với Java, tùy thuộc vào mục đích sử dụng và phạm vi hỗ trợ 1](https://quizlet.com/vn/964929770/cau-hoi-phong-van-java-flash-cards/" \t "https://edgeservices.bing.com/edgesvc/_blank). 4. [Tính tương thích: Các nền tảng khác có thể yêu cầu điều chỉnh mã nguồn khi chuyển đổi giữa các hệ điều hành hoặc phần cứng khác nhau](https://quizlet.com/vn/964929770/cau-hoi-phong-van-java-flash-cards/" \t "https://edgeservices.bing.com/edgesvc/_blank) . |
| 6 | Tính chất "viết một lần chạy nhiều nơi" của java là gì? | là một đặc điểm nổi bật của ngôn ngữ Java, thể hiện khả năng của Java trong việc viết mã nguồn một lần và chạy được trên nhiều hệ điều hành hoặc nền tảng khác nhau mà không cần chỉnh sửa mã. |
| 7 | Classloader trong java là gì? |  |
| 8 | File có tên trống ".java" có hợp lệ không? |  |
| 9 | Các từ delete, next, main, exit và null có phải là từ khóa trong java không? |  |
| 10 | Nếu không cung cấp bất kỳ đối số nào trên command line, thì mảng String của hàm main là empty hay null? |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |