- **Giấy phép phần mềm:**

Là một bản hợp đồng được chủ sở hữu phần mềm cấp cho người muốn sử dụng phần mềm. Nó bao gồm các điều khoản và điều kiện mô tả những gì mà chủ sở hữu phần mềm cho phép bạn khai thác trên phiên bản phần mềm mà bạn được cấp giấy phép sử dụng. Nó qui định về những quyền mà bạn có thể có được trên phần mềm đó.

**Giấy phép nguồn mở:**

+ Là tài liệu xác nhận bản quyền của tác giả gốc đối với phần mềm, tuy nhiên được đưa thêm các điều khoản để các hành vi sử dụng, sao chép, sửa đổi, phân phối… các phần mềm này trở thành hợp pháp. Các giấy phép nguồn mở được sử dụng để đảm bảo rằng các phần mềm, mã nguồn có sử dụng giấy phép này luôn là mã nguồn mở, phù hợp với OSD.

**Ba quy tắc trong mã nguồn mở**

* Tự do sử dụng
* Tự do phân phối
* Tự do sửa đổi

**+ Các giấy phép mã nguồn mở đảm bảo các quyền sau cho người sử dụng: (5 quyền)**

* Người sử dụng có quyền tự do sử dụng phần mềm mã nguồn mở cho bất kỳ mục đích nào.
* Người sử dụng có quyền sao chép phần mềm mã nguồn mở và phân phối chúng mà không cần thanh toán bất kỳ chi phí nào cho bên cấp phép.
* Người sử dụng có quyền tạo ra các tác phẩm phái sinh từ phần mềm mã nguồn mở và phân phối chúng mà không cần thanh toán bất kỳ chi phí nào cho bên cấp phép.
* Người sử dụng có quyền truy cập và sử dụng mã nguồn của phần mềm mã nguồn mở.
* Người sử dụng có quyền kết hợp phần mềm mã nguồn mở với các phần mềm khác.

**Đặc điểm của giấy phép Permissive? Nêu một số ví dụ?**

**Giấy phép Permissive** (hay còn gọi là giấy phép phần mềm tự do) là loại giấy phép phần mềm cho phép người dùng sử dụng, chia sẻ, sửa đổi phần mềm mà không bị giới hạn nhiều. Đặc điểm chính của giấy phép Permissive:

- Cho phép sử dụng, sao chép, sửa đổi, phân phối phần mềm mà không cần phải chia sẻ mã nguồn.

- Không yêu cầu giữ lại bản quyền, thông báo hoặc giấy phép khi phân phối lại.

- Có thể sử dụng trong các sản phẩm thương mại.

Một số ví dụ về giấy phép Permissive:

MIT License

BSD Licenses (2-Clause và 3-Clause)

Apache License 2.0

ISC License

**Đặc điểm của giấy phép Reciprocal? Nêu một số ví dụ?**

**Giấy phép Reciprocal** (hay còn gọi là giấy phép phần mềm tự do đối xứng) là loại giấy phép yêu cầu bất kỳ ai sử dụng, sửa đổi, hoặc phân phối lại phần mềm phải chia sẻ lại mã nguồn của phần mềm đó dưới cùng một giấy phép. Đặc điểm chính của giấy phép Reciprocal:

- Yêu cầu chia sẻ lại mã nguồn khi sử dụng, sửa đổi, hoặc phân phối lại phần mềm.

- Yêu cầu giữ lại bản quyền, thông báo và giấy phép khi phân phối lại.

- Có thể sử dụng trong các sản phẩm thương mại nhưng phải tuân theo các điều khoản của giấy phép.

Một số ví dụ về giấy phép Reciprocal:

IBM Public License v.1.0

Apple Public Source License v.1.2

GNU General Public License v.2.0

Nokia Open Source License v.1.0a

**Quy trình thông qua một giấy phép nguồn mở tại OSI?**

+ Cộng đồng thẩm định giấy phép sẽ thảo luận trong ít nhất 30 ngày.

+ Các ý kiến từ cộng đồng sẽ được tổng kết và đưa lên ban giám đốc OSI.

+ Ban giám đốc OSI sẽ đưa ra quyết định cuối cùng, hoặc yêu cầu các thông tin bổ sung, trong lần họp định kì tháng sau.

+ Cộng đồng thẩm định sẽ được thông báo về quyết định của ban giám đốc OSI.

+ Nếu giấy phép đó được chấp thuận, nó sẽ được đưa lên website của OSI.

- **Một số phần mềm mã nguồn mở**:

**Nêu khái niệm và lợi ích của cộng đồng mã nguồn mở?**

Cộng đồng là một nhóm cá nhân chia sẻ một/vài sự quan tâm chung nào đó. Trong đó, các thành viên tương tác, trao đổi, học hỏi lẫn nhau

**Lợi ích:**

• Người tham gia phát triển đông đảo: an toàn

• Dễ dàng sao chép: miễn phí/chi phí thấp

• Có thể sửa đổi mã nguồn

* Nhóm Hệ điều hành:

|  |  |  |
| --- | --- | --- |
| Ubuntu | Unix | gentoo linux |
| BSD | MEPIS | Debian |

* Nhóm CSDL:

|  |  |  |
| --- | --- | --- |
| MySQL | Cassandra | PostgreSQL |

* Nhóm Ứng dụng Desktop:

|  |  |
| --- | --- |
| Apache OpenOffice | GIMP |
| GZIP | eclipse |

* Nhóm Server:
* Web server: NginX, Lighttpd
* Mail server: Sendmail, qmail, exim, postfix
* File server: xtreemfs, openfiler
* CMS: Joomla, Drupal, WebsiteBaker, PHP-Fusion, MODX, Contao
* Nhóm Blog

|  |  |  |
| --- | --- | --- |
| WordPress | Dotclear | SitePad |
| HTMLY | Pubvana | B2evolution |

* Nhóm Forum

|  |  |  |
| --- | --- | --- |
| phpBB | SMF | MyBB |
| AEF | XMB | miniBB |

* Nhóm eCommerce

|  |  |  |
| --- | --- | --- |
| OpenCart | Zen Cart | LiteCart |
| CubeCart | Blesta | Quick.Cart |

* Nhóm ERP

|  |  |  |
| --- | --- | --- |
| Dolibarr | X2CRM | Tine 2.0 |
| Zurmo | webERP | EPESI |

* Nhóm Education

|  |  |  |
| --- | --- | --- |
| Moodle | Omeka | TCExam |
| ILIAS | FormLMS | Gibbon |

* Nhóm Framework

|  |  |  |
| --- | --- | --- |
| Laravel | Yii | Bootstrap |
| CakePHP | Zend | Ulkit |

* Nhóm Social network

|  |  |  |
| --- | --- | --- |
| Etano | Jcow | UNA |
| Humhub | Elgg3 |  |

**+ So sánh Laravel và CakePHP?**

Laravel và CakePHP đều là các framework PHP phổ biến

**- Điểm giống nhau:**

+ Đều là các framework PHP, hỗ trợ xây dựng ứng dụng web mạnh mẽ và linh hoạt.

+ Đều tuân theo mô hình MVC (Model-View-Controller), giúp tổ chức code một cách rõ ràng.

+ Đều hỗ trợ ORM (Object-Relational Mapping) giúp làm việc với cơ sở dữ liệu dễ dàng hơn.

+ Đều hỗ trợ routing, validation, caching, sessions, và nhiều tính năng khác.

**- Điểm khác nhau:**

+ Độ phổ biến và cộng đồng hỗ trợ: Laravel có một cộng đồng lớn hơn và được sử dụng rộng rãi hơn so với CakePHP. Điều này có nghĩa là bạn có thể tìm thấy nhiều tài liệu học, thư viện và hỗ trợ từ cộng đồng khi sử dụng Laravel.

+ Tính năng và công nghệ: Laravel sử dụng nhiều công nghệ và tính năng hiện đại hơn như hỗ trợ ORM Eloquent, hỗ trợ tạo RESTful API, hỗ trợ tạo và quản lý task với Laravel Scheduler, hỗ trợ xây dựng ứng dụng realtime với Laravel Echo. Trong khi đó, CakePHP tập trung vào việc giữ cho code đơn giản và dễ hiểu.

+ Hiệu suất: Cả hai framework đều có hiệu suất tốt, nhưng Laravel có thể hơi chậm hơn một chút do sử dụng nhiều tính năng và công nghệ hiện đại hơn.

+ Học và sử dụng: Laravel có độ dốc học tập hơi cao hơn so với CakePHP do có nhiều tính năng và công nghệ hơn. Tuy nhiên, nếu bạn đã quen với các công nghệ hiện đại như Composer, Namespaces, và ORM, bạn sẽ thấy dễ dàng hơn khi học Laravel.

- **CMS** (Content Management System): Hệ quản trị nội dung của một trang web.

+ Nhiệm vụ kiểm soát , điều khiển các hoạt động liên quan đến nội dung, thông tin của trang web.

+ Với CMS, người dùng ko cần chỉnh sửa trực tiếp những dòng code. Người dùng được trang bị những công cụ chỉnh sửa trực tiếp.Ngoài ra ko cần trực tiếp truy cập vào server, chỉ việc chỉnh sửa qua giao diện có sẵn.

+ Cho phép người dùng quản lý nội dung từ giao diện người dùng nội bộ hoặc trang tổng quan (Dashboard).

+ Chức năng chính:

|  |  |
| --- | --- |
| Tạo nội dung trên trang Web | Lưu trữ nội dung |
| Chia sẻ và di chuyển nội dung | Quản lý, phân quyền giữa các người dùng |

+ Cách thức hoạt động của CMS:

- CMS hoạt động dựa trên mô hình client-server. Khi một người dùng truy cập vào website, trình duyệt web (client) sẽ gửi yêu cầu đến máy chủ (server) nơi CMS đang chạy. CMS sẽ truy vấn cơ sở dữ liệu để lấy nội dung, sau đó tạo ra trang web dựa trên nội dung đó và gửi trở lại cho trình duyệt web.

- Khi người dùng muốn chỉnh sửa nội dung, họ sẽ đăng nhập vào hệ thống quản lý của CMS (thường là một giao diện web), tạo hoặc chỉnh sửa nội dung thông qua giao diện đó. CMS sau đó sẽ lưu nội dung mới vào cơ sở dữ liệu.

+ Ưu/ nhược điểm:

* Ưu điểm:
* Dễ dàng cài đặt, sửa đổi và cập nhật mã nguồn
* Ko cần khả năng lập trình từ người sử dụng CMS
* Có sự tuỳ biến từ bộ giao diện có sẵn
* Dễ dàng phân quyền người dùng
* Có cộng đồng những người sử dụng lớn
* Khả năng mở rộng chức năng thông qua các công cụ mở rộng
* Nhược điểm:
* Không có giao diện đặc thù cho trang Web
* Các themes được thiết kế sẵn thường bị rập khuôn.
* Không tối ưu tốt cho tốc độ tải trang Web.
* Có thể làm Website chậm lại.
* Độ linh hoạt thấp hơn so với những Website tự code.

**Joomla, Drupal, WebsiteBaker, MODX, Contao**

\* **Joomla**: (Đồng tâm hiệp lực) Độ tin cậy cao, dễ dàng cài đặt & quản lý

- Là một hệ quản trị nội dung (CMS) mã nguồn mở

- Được viết bằng ngôn ngữ PHP và hệ CSDL MySQL

**\* Drupal:** là một CMS dành cho dân chuyên nghiệp, được dùng nhiều bởi các tập đoàn lớn trên thế giới.

- **Wordpress**: CMS miễn phí, dễ sử dụng và phổ biến

+ Xây dựng web cá nhân, blog, web tin tức, báo điện tử, TMĐT,…

+ Viết = ngôn ngữ lập trình PHP sử dụng hệ quản trị CSDL MySQL

+ Các chức năng chính của WordPress:

- Quản lý nội dung: WordPress cho phép bạn tạo, chỉnh sửa, và quản lý các bài viết và trang web. Nó cung cấp một trình soạn thảo văn bản phong phú, hỗ trợ định dạng văn bản, hình ảnh, video, và nhiều loại nội dung khác.

- Theme và plugin: WordPress hỗ trợ hàng ngàn theme và plugin, giúp bạn tùy chỉnh giao diện và mở rộng chức năng của website. Có cả theme và plugin miễn phí lẫn có phí.

- SEO-friendly: WordPress hỗ trợ các tính năng SEO cơ bản, giúp website của bạn tối ưu hóa cho công cụ tìm kiếm. Có nhiều plugin SEO mạnh mẽ cho WordPress, như Yoast SEO và All in One SEO Pack.

- Quản lý người dùng: WordPress cho phép bạn tạo nhiều tài khoản người dùng với các vai trò khác nhau, như quản trị viên, tác giả, biên tập viên, và người đóng góp.

+ Cách thức hoạt động của WordPress:

- Khi một người dùng truy cập một trang web WordPress, trình duyệt của họ gửi một yêu cầu đến server web. Server web sau đó chạy mã PHP của WordPress, lấy dữ liệu từ cơ sở dữ liệu MySQL, và tạo ra trang HTML để gửi trở lại trình duyệt.

- Khi bạn đăng nhập vào trang quản lý của WordPress (còn được gọi là Dashboard), bạn có thể tạo và chỉnh sửa nội dung, cài đặt và tùy chỉnh theme và plugin, quản lý người dùng, và thực hiện nhiều tác vụ quản lý khác. Mọi thay đổi bạn thực hiện trong Dashboard sẽ được lưu trong cơ sở dữ liệu và hiển thị trên website của bạn.

+ Tính năng & ưu điểm:

- Hệ thống Plugin & Theme đa dạng, phong phú,…

- Hỗ trợ đa ngôn ngữ. - Dễ dàng sử dụng, quản lý,…

- Cập nhật phiên bản liên tục, cộng đồng hỗ trợ lớn.

- Tích hợp sẵn Latex.

- Thể hiện các tệp PDF, DOC, Powerpoint ngay trên nội dung bài viết.

+ Nhược điểm:

- Khó cài đặt template & plugin

- Chỉ phù hợp cho các doanh nghiệp vừa và nhỏ

+ So sánh WordPress.com và WordPress.org?

WordPress.com và WordPress.org đều liên quan đến WordPress, một hệ thống quản lý nội dung (CMS) phổ biến.

- Điểm giống nhau:

Cả hai đều sử dụng WordPress, một CMS phổ biến giúp người dùng tạo và quản lý website mà không cần kiến thức chuyên sâu về lập trình.

Cả hai đều cho phép bạn tạo và quản lý nội dung trên website của mình.

- Điểm khác nhau:

Hosting: WordPress.com là một dịch vụ hosting được quản lý, nghĩa là họ cung cấp không chỉ phần mềm WordPress mà còn cả không gian lưu trữ cho website của bạn. Ngược lại, WordPress.org chỉ cung cấp phần mềm WordPress, và bạn cần tự tìm một dịch vụ hosting để lưu trữ website của mình.

Tùy chỉnh: WordPress.org cho phép bạn tùy chỉnh mọi thứ trên website của mình, từ theme, plugin cho đến mã nguồn của WordPress. Ngược lại, WordPress.com giới hạn một số tùy chỉnh để đảm bảo ổn định và bảo mật cho dịch vụ của họ. Ví dụ, bạn không thể cài đặt plugin tùy ý trên WordPress.com trừ khi bạn nâng cấp lên gói Business.

Giá cả: WordPress.org miễn phí, nhưng bạn cần tự chi trả cho dịch vụ hosting và tên miền. Ngược lại, WordPress.com cung cấp một số gói dịch vụ có phí, bao gồm hosting và tên miền, nhưng cũng có gói miễn phí với các tính năng giới hạn.

Bảo dưỡng và bảo mật: Với WordPress.com, họ sẽ chịu trách nhiệm về bảo dưỡng và bảo mật cho website của bạn. Ngược lại, với WordPress.org, bạn sẽ cần tự quản lý những việc này.

- **Version Control System:** hệ thống quản lý phiên bản

* Là hệ thống lưu giữ các phiên bản của mã nguồn của sản phẩm phần mềm.
* Cho phép người quản trị phân chia các tập tin cho từng thành viên, quản lý lịch sử thay đổi file/ mã nguồn.
* Cho phép khôi phục lại phiên bản cũ của các file
* Cho phép các thành viên có thể phát hiện lỗi và sửa lỗi một cách thuận tiện
* Tự động so sánh và cập nhật mã nguồn khi các thành viên sửa đổi cùng lúc

**+ Mô hình hoạt động**

* Centralized version control system(CVCS) Mô hình tập trung
* Điển hình của hệ thống quản lý phiên bản sử dụng mô hình tập trung là SVN(Subversion)
* Mô hình sử dụng kiến trúc Client-Server. Server đóng vai trò là một kho dữ liệu (Repository) duy nhất lưu trữ tất cả các tập tin, mã nguồn, thay đổi,… Các client sẽ kết nối đến server để lấy phiên bản mới nhất của mã nguồn về máy(checkout) và cập nhật tất cả các thay đổi lên server(commit).
* Distributed version control system (DVCS) Mô hình phân tán
* Điển hình của hệ thống quản lý phiên bản sử dụng mô hình phân tán là Git
* Khi client lấy mã nguồn phiên bản mới nhất từ server về máy, hệ thống sẽ sao chép luôn cả kho mã nguồn, máy client lúc này đóng vai trò là kho mã nguồn, mỗi kho con sẽ đồng bộ các thay đổi đến kho trên server chính.
* Sự linh hoạt trong việc cập nhật lại mã nguồn trên server, client không cần lúc nào phải duy trì kết nối đến server.

**+ Lợi ích của hệ thống:**

* Lưu trữ mã nguồn tập trung
* Tạo môi trường làm việc nhóm
* Lưu vết các thay đổi trong mã nguồn

**+ Ưu điểm của VCS:**

* Tăng hiệu suất làm việc: VCS giúp đội ngũ lập trình viên làm việc hiệu quả hơn bằng cách cho phép họ làm việc cùng nhau mà không gây ra xung đột.
* Phục hồi dễ dàng: Nếu có lỗi xảy ra, bạn có thể dễ dàng quay lại phiên bản trước của mã nguồn.
* Dễ dàng theo dõi thay đổi: VCS giúp bạn theo dõi ai đã thực hiện thay đổi nào và khi nào thay đổi đã được thực hiện.

**+ Nhược điểm của VCS:**

* Đòi hỏi thời gian học hỏi: VCS có thể khá phức tạp và đòi hỏi thời gian để học cách sử dụng hiệu quả.
* Cần quản lý cẩn thận: Nếu không được quản lý cẩn thận, VCS có thể dẫn đến các vấn đề như xung đột mã nguồn, lịch sử mã nguồn rối rắm, và vấn đề khác.
* Cần cấu hình phù hợp: Để tận dụng tối đa VCS, bạn cần cấu hình nó phù hợp với quy trình làm việc của đội ngũ.

**+ Phần mềm kiểm soát:**

\* Git: là phần mềm quản lý mã nguồn phân tán (được phát triển bởi Linus Torvalds) mã nguồn mở, đa nền tảng và miễn phí, cung cấp hỗ trợ mạnh mẽ cho phát triển phi tuyến tính, có khả năng xử lý hiệu quả mọi thứ từ các dự án nhỏ đến rất lớn với tốc độ và hiệu quả.

**\* CVS (Concurrent Versions System): hệ thống các phiên bản đồng thời.**

+ Công nghệ hoàn thiện với hỗ trợ đa nền tảng

+ Trợ giúp và hỗ trợ tức thì

+ Dòng lệnh mạnh mẽ và đầy đủ tính năng

+ Có thể làm việc song song trên cùng một dự án.

+ Không kiểm tra tính toàn vẹn cho mã nguồn

+ Hỗ trợ kém cho kiểm soát nguồn phân tán

**\* Apache Subversion (SVN)**

+ Được thiết kế với mục đích thay thế CVS

+ Được dùng để quản lý mã nguồn cho nhiều ứng dụng Phần mềm miễn phí

như: KDE, GNOME, Python, Ruby,…

+ Meta data được đánh dấu phiên bản dạng tự do

+ Có hỗ trợ windows tốt hơn so với Git + Còn bug liên quan đến việc chuẩn hóa tên tập tin và thư mục

\* Bazaar

+ Các công cụ phát triển ứng dụng: Xampp, Php, MySQL

- **PHP**

Mô tả chi tiết: Ai là người dùng, chức năng người dùng, kết quả từng chức năng đó là gì, tính năng, kết quả tổng của ứng dụng ra sao.

+ PHP (viết tắt của "PHP: Hypertext Preprocessor") là một ngôn ngữ lập trình mã nguồn mở phía máy chủ, chủ yếu được sử dụng để phát triển các ứng dụng web. PHP được tích hợp sẵn vào HTML và thường được sử dụng cùng với hệ thống quản lý cơ sở dữ liệu MySQL.

**+ Ưu điểm của PHP:**

- Mã nguồn mở: PHP là ngôn ngữ lập trình mã nguồn mở, do đó nó miễn phí để sử dụng và phát triển.

- Dễ học: PHP có cú pháp đơn giản và dễ hiểu, giúp người mới bắt đầu dễ dàng học hỏi.

- Tích hợp sẵn vào HTML: PHP có thể được nhúng trực tiếp vào mã HTML, giúp tạo ra các trang web động một cách dễ dàng.

- Hỗ trợ cơ sở dữ liệu rộng rãi: PHP hỗ trợ nhiều hệ thống quản lý cơ sở dữ liệu, bao gồm MySQL, PostgreSQL, Oracle, và nhiều hệ thống khác.

- Cộng đồng lớn: PHP có một cộng đồng lập trình viên lớn, nghĩa là có nhiều tài liệu học tập, thư viện mã nguồn mở và hỗ trợ khi gặp vấn đề.

**+ Nhược điểm của PHP:**

- Hiệu suất: Mặc dù PHP đã được cải thiện đáng kể về hiệu suất trong các phiên bản gần đây, nhưng nó vẫn không nhanh bằng một số ngôn ngữ lập trình khác như Node.js hoặc Go.

- Quản lý lỗi: PHP không quản lý lỗi tốt như một số ngôn ngữ lập trình khác. Điều này có thể gây khó khăn khi debug và bảo trì mã nguồn.

- Không thích hợp cho các ứng dụng lớn: Do cấu trúc và tính năng của nó, PHP không phải lựa chọn tốt nhất cho các ứng dụng lớn và phức tạp.

- Bảo mật: Mặc dù PHP có các tính năng bảo mật, nhưng nếu không được sử dụng đúng cách, nó có thể dễ dàng bị tấn công bởi các loại tấn công như SQL Injection, XSS, và CSRF.

**PHP kết nối đến CSDL MySQL bằng cách nào?**

+ PHP kết nối đến cơ sở dữ liệu MySQL thông qua các hàm cung cấp bởi extension MySQLi hoặc PDO (PHP Data Objects).

**+ Ví dụ về cách kết nối đến MySQL sử dụng MySQLi:**

<?php

$servername = "localhost";

$username = "username";

$password = "password";

$dbname = "database";

// Tạo kết nối

$conn = new mysqli($servername, $username, $password, $dbname);

// Kiểm tra kết nối

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

echo "Connected successfully";

?>

**+ Ví dụ về cách kết nối đến MySQL sử dụng PDO:**

<?php

$servername = "localhost";

$username = "username";

$password = "password";

$dbname = "database";

try {

$conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);

// set the PDO error mode to exception

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

echo "Connected successfully";

} catch(PDOException $e) {

echo "Connection failed: " . $e->getMessage();

}

?>

**+ So sánh MySQLi và PDO?**

- MySQLi (MySQL Improved) và PDO (PHP Data Objects) đều là các phần mở rộng của PHP giúp kết nối và tương tác với cơ sở dữ liệu.

- Bảo mật: Cả MySQLi và PDO đều hỗ trợ các truy vấn được chuẩn bị (prepared statements), giúp bảo vệ chống lại các cuộc tấn công SQL Injection.

- Dễ sử dụng: Cả hai phần mở rộng đều dễ sử dụng, nhưng nhiều người cho rằng PDO dễ sử dụng hơn do API hướng đối tượng của nó.

- Hỗ trợ cơ sở dữ liệu: MySQLi chỉ hỗ trợ cơ sở dữ liệu MySQL và MariaDB. Trong khi đó, PDO hỗ trợ nhiều loại cơ sở dữ liệu khác nhau bao gồm MySQL, PostgreSQL, SQLite, Oracle, và nhiều hệ thống khác.

- API: MySQLi cung cấp cả API hướng đối tượng và API thủ tục. Trong khi đó, PDO chỉ cung cấp API hướng đối tượng.

- Tính năng: MySQLi cung cấp một số tính năng chỉ dành riêng cho MySQL như các hàm lưu trữ, hỗ trợ cho các truy vấn nhiều kết quả, v.v. Trong khi đó, PDO không hỗ trợ các tính năng cụ thể của MySQL nhưng lại cung cấp một số tính năng khác như giao dịch, lấy dữ liệu dưới dạng đối tượng, v.v.

**Trình bày các bước truy vấn CSDL MySQL sử dụng thư viện PDO?**

Để truy vấn cơ sở dữ liệu MySQL sử dụng thư viện PDO, ta có thể thực hiện theo các bước sau:

**Bước 1: Kết nối đến cơ sở dữ liệu**

$host = 'localhost';

$db = 'testdb';

$user = 'testuser';

$pass = 'testpass';

$charset = 'utf8mb4';

$dsn = "mysql:host=$host;dbname=$db;charset=$charset";

$opt = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE=>PDO::FETCH\_ASSOC,

PDO::ATTR\_EMULATE\_PREPARES => false,

];

$pdo = new PDO($dsn, $user, $pass, $opt);

**Bước 2: Chuẩn bị truy vấn SQL**

<?php

$stmt = $pdo->prepare('SELECT \* FROM users WHERE email = ? AND status=?');

?>

**Bước 3: Thực thi truy vấn với các tham số cần thiết**

<?php

$stmt->execute([$email, $status]);

?>

**Bước 4: Lấy kết quả truy vấn**

<?php

$user = $stmt->fetch();

?>

**Bước 5: Xử lý kết quả**

<?php

if ($user) {

echo $user['name'];

} else {

echo "No users found";

}

?>

**Nêu ý nghĩa các biến có sẵn trong PHP?**

PHP cung cấp một số biến dựng sẵn (còn được gọi là biến siêu toàn cục hoặc biến tự động) để lưu trữ thông tin về yêu cầu, phiên làm việc, môi trường, v.v.

$\_SERVER: Một mảng chứa thông tin về tiêu đề, đường dẫn và vị trí script. Các thông tin này được tạo bởi máy chủ web.

$\_GET: Một mảng liên kết chứa các biến được gửi đến script hiện tại thông qua URL parameters.

$\_POST: Một mảng liên kết chứa các biến được gửi đến script hiện tại thông qua HTTP POST.

$\_FILES: Một mảng liên kết chứa các item được tải lên trong script hiện tại thông qua phương thức POST.

$\_COOKIE: Một mảng liên kết chứa các biến được gửi đến script hiện tại thông qua HTTP Cookies.

$\_SESSION: Một mảng liên kết chứa các biến phiên làm việc có sẵn cho script hiện tại.

$\_REQUEST: Một mảng liên kết chứa các biến $\_GET, $\_POST và $\_COOKIE.

$\_ENV: Một mảng liên kết chứa các biến môi trường.

$GLOBALS: Một mảng liên kết chứa tất cả các biến toàn cục trong script hiện tại.

Tất cả các biến này đều bắt đầu bằng dấu gạch dưới (\_) và tất cả chữ cái đều viết HOA.

**Viết code PHP thực hiện những bài toán sau:**

|  |  |
| --- | --- |
| **• Kiểm tra một số là chẵn hay lẻ?**  <?php  function checkEvenOrOdd($number) {  if ($number % 2 == 0) {  echo "The number $number is even.";  } else {  echo "The number $number is odd.";  }  }ConcheckEvenOrOdd(7);  ?> | **• Giải phương trình bậc nhất 1 ẩn?**  <?php  function solveLinearEquation($a, $b) {  if ($a == 0) {  echo "No solution.";  } else {  $solution = -$b / $a;  echo "The solution is: $solution";  }  }  solveLinearEquation(2, 3);  ?> |
| **• Giải phương trình bậc 2?**  <?php  function solveQuadraticEquation($a, $b, $c) {  $delta = $b\*$b - 4\*$a\*$c;  if ($delta < 0) {  echo "No solution.";  } elseif ($delta == 0) {  $solution = -$b / (2\*$a);  echo "One solution: $solution";  } else {  $solution1 = (-$b - sqrt($delta)) / (2\*$a);  $solution2 = (-$b + sqrt($delta)) / (2\*$a);  echo "Two solutions: $solution1, $solution2";  }  }  solveQuadraticEquation(1, -3, 2);  ?> | **• Kiểm tra một số có phải là nguyên tố hay không?**  <?php  function isPrime($number) {  if ($number < 2) {  return false;  }  for ($i = 2; $i \* $i <= $number; $i++) {  if ($number % $i == 0) {  return false;  }  }  return true;  }  $number = 7;  if (isPrime($number)) {  echo "$number is a prime number.";  } else {  echo "$number is not a prime number.";  }  ?> |
| **• In ra màn hình bảng cửu chương?**  <?php  for ($i = 1; $i <= 10; $i++) {  for ($j = 1; $j <= 10; $j++) {  $product = $i \* $j;  echo "$i \* $j = $product\n";  }  echo "\n";  }  ?> | |