OpenSWPlatform 중간고사 정리

Open Source SW

Open Source Software

: A type of computer software in which source code is released under a license in which the copyright holder grants users the rights to the use, study, change, and distribute the software to anyone and for any purpose.

Pros and Cons

Advantages

1. Cheaper than commercially marketed products.
2. Created by skillful and talented people.
3. Highly reliable.
4. Help you become more flexible

Disadvantages

1. Vulnerable to malicious users.
2. Might not be as user-friendly as commercial versions.
3. Don't come with extensice support.

Open Source Initiative

: A non-profit corporation whose goal is to promote the use of open source software in the commercial world. To be approved by the Open Source Initiative, a license must go through the Open Source Initiative's license review process.

Open Source License

: License that comply with the Open Source Definition – in brief, they allow software to be freely used, modified, and shared.

Open Source Definition

: a document published by the Open Source Initiative, to determine whether a software license can be labeled with the open-source certification mark.

1. Free Redistribution

: The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

2. Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost, preferably downloading via the internet without charge. The source code must be preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

3. Derived Works

The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

4. Integrity of the Author's Source Code

The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

5. No Discrimination Against Persons or Groups

The license must not discriminate against any person or group of persons.

6. No Discrimination Against Fields of Endeavor

The license must not restrict anyone from making the use of the program in a specific field of endeavor. For Example, it may not restrict the program from being used in a business, or from being used for genetic research.

7. Distribution of License

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

8. License Must Not Be Specific to a Product

The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

9. License Must Not Restrict Other Software

The License must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

10. License Must Be Technology-Neutral

No provision of the license may be predicated on any individual technology or style of interface.

Open Source License

: Licenses that comply with the Open Source Definition. They allow software to be freely used, modified, and shared.

How to contribute to Open Source Projects
: Use open source SW, Report bugs, request new ideas, fix bugs, review others' code, share your projects

Example of Open-Source Projects
GPL License
Linux Kernel: Similar to Unix, the most representative open source software
Ubuntu: Linux Family
Android: Mobile operating system developed by Google
MySQL: relational DB Management system
WordPress: Content Management System
Java
GIMP: Image Editor

Apache License 2.0
Apache HTTP Server: Opensource web server
OpenOffice: for utility software (word processor, spreadsheet, presentation)
Tensorflow

MPL 2.0 (Mozilla Public License)
Firefox: a web browser by Mozilla Foundation

MIT License
Node.js

License specified at its Github repository
Node.js: an open-source, cross-platform, JavaScript runtime environment that executes Javascript code outside a web browser

PHP License
PHP: Hypertext preprocessing language for web development

Git

Git
: a distributed version-control(record changes) system for tracking changes in source code during software development.

Version Control
: a system that records changes to a file or set of files over time so that you can recall specific versions later.
Revert project back to a previous state/ compare changes over time/ see who last modified something that might be causing a problem/ who introduced an issue
(Local Version Control System/Centralized Version Control System/Distributed Version Control system

Merge VS. Rebase
Merge: non-destructive operation. The existing branches are not changed in any way.
Rebase: re-writes the project history by creating brand new commits for each commit in the original branch.

HTML

-웹: HTTP 를 이용하는 인터페이스

Internet Services example: Telnet(원격조종), FTP(File Transfer), WWW, IRC(Internet Relay chat), HTTP, Email

WWW 의 구성 요소

- Web server: a computer connected to Internet that waits request from a browser
- Web browser: a software application for accessing information on the world web web
- HTTP: HyperText Transfer Protocol
- URL: 웹 브라우저 주소
- HTML(HyperText Markup Language): 웹 페이지에 포함된 정보와 그 정보를 디스플레이하는 방법을 기술하는 표준 언어
- CSS(Cascading Style sheet): style sheet language
- javascript: client-side scripting VS. php: server-side functions

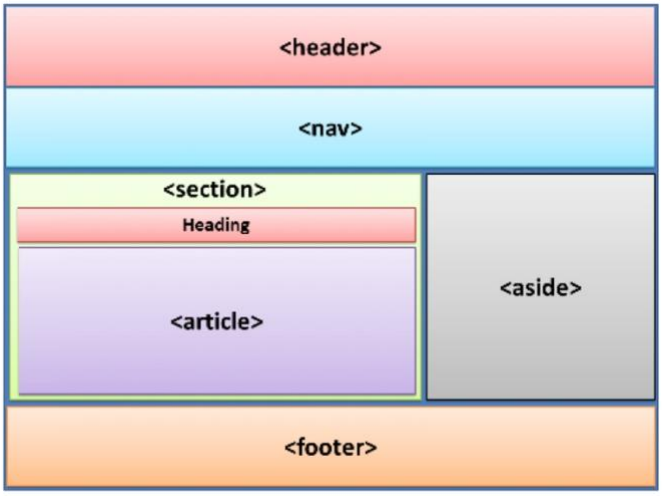
(검색엔진: 부분적인 정보만을 가지고 원하는 정보를 찾을 수 있도록 도와주는 www 시스템 상의 도구)

HTML tags: hidden keywords within a web page that define how your web browser must format and display the content.

HTML VS. HTML5

HTML	HTML5
No standardized process to handle structurally incorrect HTML codes	supports persistent error handling via improvised error handling process
nope.	mobile friendly/audio&video support/support all major web browsers./Javascript in background
vector graphics is possible when used in conjunction with third-party plugins.	Scalable Vector Graphics(SVG) is an integral part of the HTML4 language specification

Standard webpage layout



JavaScript
Object oriented language/interpreted programming language with object-oriented capabilities/ allow client-side script to interact with the user and make dynamic pages.
Pros
1. less server interaction – You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
2. Immediate feedback to the visitor – They don't have to wait for a page reload to see if they have forgotten to enter something.
3. Increased interactivity – You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
4. Richer interfaces – You can use javascript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.
Cons
1. client side javascript does not allow the reading or writing of files. This has been kept for security reason.
2. Javascript cannot be used for networking applications because there is no such support available.
3. Javascript doesn't have any multi-threading or multiprocessor capabilities.

PHP

PHP: Hypertext Preprocessor(Open-source project)
Server-side scripting language to develop

Good performance/extendable/interface with other database systems/libraries/low cost/easy to learn/oop(object oriented programming)/cross-platform
+ no additional component is needed and can be accessed via a web browser
+ Easy to verify forms
- not good for heavy computation, or managing other devices
- Bad security

Node.js

: Open-source, Cross-Platform, back-end, Javascript runtime environment that executes Javascript code outside a web browser / Web Server (alternative to PHP back-end service) / Event-Driven Programming Non-blocking/Very fast/single threaded but highly scalable/no buffering/MIT License

PHP vs Node.js
PHP:
- Mixed use of HTML, CSS, Database
- Easy to code(no need for compilers, interpreters)
- Database Management(ex. SQL)
- Heavy CPU computation, size
Node.js:
- I/O bound Application
- Data Intensive Real-time Application(DIRT) / Data Streaming Application
- JSON APIs based Application
- Single Page Application

*(Event Driven Programming)
There is generally a main loop that listens for events, and then triggers a callback function when one of those events is detected.

Web Application Architecture
Client: makes HTTP requests to the web server
Server: intercepts the requests from clients and pass them the response
Business Layer: interacts with the data layer via database or some external programs (Application server, File System)
Data Layer: contains the databases or any other source of data