

**1. Explain what each level of the OSI model does.**

- (from bottom to top):
  1. Physical
    - In this layer, the raw bit stream is transferred over the physical medium.
  2. Data Link
    - Defines the format of data on the network.
  3. Network
    - Determines the physical path that the data will take to its destination(s)/recipient(s).
  4. Transport
    - Data is transmitted using a transmission protocol such as TCP and UDP.
  5. Session
    - Maintains connections between the browser and the server and is responsible for sockets and the end-users' sessions.
  6. Presentation
    - Checks to make sure data being input by the end-user is in the correct format and valid. This is also the layer where encryption takes place.
  7. Application
    - The layer that the end-user sees. It is where applications access the network services.

**2. What is the difference between a compiled language, a scripting language, and a JIT (Just-In-Time compiler)?**

- A compiled language (Java, C/C++, C#, etc.) has a separate phase *before* runtime, known as *compilation*. During this phase, the compiler goes line by line through the source code, and initializes constants, designates declared types, function headers/parameters, etc. This is done in order to allocate memory to be used at runtime.
- A scripting language (Python, Lua, Bash, etc.) does not have a separate phase for compilation, but rather allocates memory (designates types, function headers/parameters, etc.) at runtime. Scripting is generally slower to run, but the syntax is typically easier to code as opposed to compiled languages which are very fast to run, but whose syntax require much more careful coding.
- Just-In-Time uses a mix of both scripting and compilation, allocating some things in memory before runtime and some during runtime. An example of a language that uses JIT is the JavaScript language.

**3. HTML5 encourages semantic markup. What does that mean?**

- Semantic markup involves using HTML code to convey information about the meaning of each element in a webpage, beyond just how they appear visually. It helps browsers, search engines, and assistive technologies understand the structure and content of a webpage, improving accessibility and SEO.

**4. What is the DOM ([https://developer.mozilla.org/en-US/docs/Web/API/Document\\_Object\\_Model](https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model))**

- The DOM, or Document Object Model, is an interface between the web page and external programs, which allows those external programs to access specific components of that web page. This is typically done using a `.json` (JavaScript Object Notation) file, wherein the components of the web page are broken down into a tree-structure, making it conveniently accessible to be manipulated by programs.