DWA_02.8 Knowledge Check_DWA2

_____

1. What do ES5, ES6, and ES2015 mean - and what are the differences between them?

ES5, ES6, and ES2015 are different versions of the ECMAScript (ES) specification, which is a standard for scripting languages like JavaScript. ES5 was released in 2009 and brought significant updates to JavaScript, including new array methods and improved object manipulation. ES6, also known as ES2015, was released in 2015 and introduced many new features like arrow functions, block-scoped variables (let and const), classes, and modules. ES2015 is used interchangeably with ES6 to refer to the same version. ES6/ES2015 built upon ES5 and enhanced JavaScript with powerful syntax and productivity improvements. Subsequent versions like ES7, ES8, and so on continued to introduce more features and enhancements to the language.

_____

2. What are JScript, ActionScript, and ECMAScript - and how do they relate to JavaScript?

JScript, ActionScript, and ECMAScript are all related to JavaScript in different ways.
- JScript is Microsoft's implementation of the ECMAScript specification and was primarily used in Internet Explorer.
- ActionScript is a scripting language derived from ECMAScript, specifically for Adobe Flash and Adobe AIR platforms. ECMAScript is the standardized specification that defines the syntax, semantics, and behavior of scripting languages like JavaScript.

JavaScript is the most widely used implementation of ECMAScript, but other languages like JScript and ActionScript also implement the ECMAScript specification with their own additions and differences.

_____

3. What is an example of a JavaScript specification - and where can you find it?

The ECMAScript specification is an illustration of a JavaScript specification. The syntax, semantics, and behavior of the JavaScript language are outlined in the ECMAScript specification. ECMAScript 2022 (ES2022) is the most recent version of the ECMAScript specification, therefore it is important to keep in mind that the specification is updated frequently, thus there may be more recent versions accessible. The organization in charge of standardizing JavaScript is Ecma International, which also maintains the ECMAScript website, where you may obtain the ECMAScript specification. The most recent version of the standard, as well as earlier versions and drafts, are accessible on their website. On this ECMAScript website, you'll find the specification in the form of a PDF document that is downloadable and it has detailed information about the JavaScript language and its features.
https://www.ecma-international.org/publications-and-standards/standards/ecma-262/

_____

4. What are v8, SpiderMonkey, Chakra, and Tamarin? Do they run JavaScript differently?

V8, SpiderMonkey, Chakra, and Tamarin are JavaScript engines responsible for executing JavaScript code in different web browsers and environments.

- V8 is developed by Google and used in Chrome and Node.js, known for its high-performance capabilities and efficient just-in-time (JIT) compilation techniques. It has been optimized for speed and is constantly evolving to improve JavaScript execution.
- SpiderMonkey is Mozilla's engine, the first-ever JavaScript engine, and has been in development since the early days of JavaScript. It has played a significant role in the advancement of JavaScript and has introduced many innovative features and optimizations over the years.
- Chakra, also known as ChakraCore formerly used in Internet Explorer and Edge, is now open-source.
- Tamarin is a JavaScript engine developed by Adobe Systems for ActionScript and JavaScript in Adobe Flash Player and Adobe AIR platforms. It utilizes the just-in-time compilation and introduces advanced optimization techniques to improve the performance of ActionScript and JavaScript code.

Each engine may have different performance, optimization strategies, and additional features, as they are developed by different organizations and prioritize different requirements. They also have their own implementations of the ECMAScript standard.

_____

5. Show a practical example using **caniuse.com** and the MDN compatibility table.

- Caniuse.com is a website that provides information on web browser compatibility for various web technologies, including JavaScript features like the `fetch()` function. By searching for a specific feature, such as `fetch()`, developers can see which browsers and versions support it through a visual representation and detailed information.

- The MDN compatibility table, available on the Mozilla Developer Network website, offers similar information specifically focused on Mozilla Firefox. Developers can search for the `fetch()` function to view a table showing its browser support across different versions of Firefox.

By using both caniuse.com and the MDN compatibility table, as a developer you can gain insights into the compatibility of features like `fetch()` across different browsers and make informed decisions in their web development projects.

Taking the above-mentioned example where we want to check the compatibility of the `fetch()` function, which is commonly used for making HTTP requests in JavaScript.

- To use caniuse.com:

1. Go to caniuse.com in your web browser.

2. In the search bar, type "fetch" and press Enter.

3. The page will display information about the `fetch()` function, including browser support and usage statistics.

There will be a table that shows which versions of different web browsers support the `fetch()` function, along with a visual representation of support in the form of colored cells. Green cells indicate full support, while yellow and red cells indicate partial or limited support, respectively. Additionally, when clicking on individual cells that will allow you to view more detailed information about the specific browser version.

- To use the MDN compatibility table:

1. Go to the MDN web docs at developer.mozilla.org.

2. In the search bar, type "fetch()" and press Enter.

3. The search results will display the documentation for the `fetch()` function.

4. Scroll down to the "Browser compatibility" section to view a table showing the browser support for `fetch()` across different versions of Firefox.

According to the MDN compatibility table, Firefox's 'fetch()' function is supported, along with its specific version numbers and any known problems or inconsistencies. Developers can learn more about the browser support for particular features and make educated decisions about adopting those features in their web development projects by using both caniuse.com and the MDN compatibility table.

_____