DWA_02.8 Knowledge Check_DWA2

---

1. What do ES5, ES6 and ES2015 mean - and what are the differences between them?
Different versions of ECMAScript.
ES5, ES6 and ES2015 are terms related to the versions of the ECMAScript standard, which is the standardized specification for JavaScript.

- ES5-stands for ECMAScript 5. It was released in December 2009 and introduced significant enhancements to JavaScript, including strict mode, new array methods(such as 'forEach', 'map', 'filter'), JSON support, getter/setter syntax, and more. ES5 is widely supported by modern browsers and is considered the baseline for JavaScript development.
- ES6-ES6 stands for ECMAScript 6, which is also known as ECMAScript 6, which is also known as ECMAScript 2015. It was released in June 2015 and brought numerous new features and improvements to JavaScript. ES6 introduced let and const for block-scoped variables, arrow functions, classes, template literals, destructuring assignments, modules and many other enhancements. ES6 has greatly improved readability and maintainability of JavaScript code.
- ES2015-is the formal name for ECMAScript2015, which is the same as ES6. The name "ES2015" was adopted to reflect the year it was released.

Differences

- ES5 introduced strict mode, new array methods(forEach, map and filter), JSON support, getter/setter syntax.
- ES6 introduced let and const for block-scoped variables, arrow functions, classes, template literals, destructuring assignments, modules and more.

---

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

- JScript-is a scripting language developed by Microsoft. It shares similarities with JavaScript.
- ActionScript-is a scripting language primarily used for developing applications, animations and interactive content on the Adobe Flash platform. It is an object-oriented programming (OOP) language and is based on ECMAScript.
- ECMAScript-a general purpose, cross-platform programming language which was adopted by Netscape and Microsoft in 1997.

How are they related to JavaScript

- JScript and ActionScript are specific implementations of the ECMAScript standard.
- JScript is associated with Internet Explorer.
- ActionScript is associated with the Adobe Flash platform.

---

3. What is an example of a JavaScript specification - and where can you find it?
   - ECMAScript specification.
   - ECMAScript specification can be found on the official website of ECMA International.

---

4. What are v8, SpiderMonkey, Chakra and Tamarin? Do they run JavaScript differently?
   - V8 is a free and open source JavaScript and WebAssembly engine developed by the Chromium Project for Chromium and Google Chrome web browser.

   - SpiderMonkey is an open-source JavaScript and WebAssembly engine by Mozilla Foundation.
   - Chakra is a proprietary JScript engine developed by Microsoft. It is used in the Internet Explorer Web browser.
   - Tamarin is a discontinued free software virtual machine with just in time compilation(JIT) support intended to implement the 4th edition of the ECMAScript(ES4) language standard.

---

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

- If you want to use ‘fetch’ API to make AJAX requests in your JavaScript code. Before using it, you want to check its browser compatibility using both caniuse.com and MDN compatibility table.
- Start by visiting caniuse.com, a website that provides up-to-date information on web technologies' browser compatibility.
- Search for "fetch" in the search bar on the homepage. The search results will display various versions and variations of the Fetch API.
- Click on the first result, which should be "Fetch API". This will take you to the detailed page for the Fetch API on caniuse.com.
- On this page, you'll see a table that shows browser support for the Fetch API across different versions of popular web browsers. The table will indicate whether a particular browser version fully supports the Fetch API or requires a polyfill or fallback.
- Take note of the browser versions that are relevant to your target audience or project requirements. The table may also show partial support or indicate specific features that are missing in some versions.
- Now, let's complement this information with the MDN compatibility table. Open a new tab and search for "MDN Fetch API" on your preferred search engine.
- The search results should include the Fetch API documentation page on the Mozilla Developer Network (MDN). Click on the link to open the page.
- Scroll down to the "Browser compatibility" section on the MDN page. Here, you'll find a compatibility table that shows support for the Fetch API across different browsers.
- The MDN compatibility table provides similar information to caniuse.com but may offer additional details or explanations about specific browser versions and their compatibility.
- Compare the information from caniuse.com and the MDN compatibility table to get a comprehensive understanding of the Fetch API's browser compatibility. Take note of any discrepancies or additional information provided by the MDN documentation.

By combining the information from caniuse.com and the MDN compatibility table, you can make informed decisions about whether to use the Fetch API directly or if you need to consider polyfills or fallbacks to ensure compatibility with your target audience's browsers.
