DWA_02.8 Knowledge Check_DWA2

---

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

i) ES5 means ECMAScript 5 and is the fifth edition of this specification. ECMAScript is the standardized scripting language specification that JavaScript is based on. It focuses on the syntax, semantics, and behavior of the language.

ii) ES6 stands for ECMAScript 6. It is the sixth edition of the ECMAScript specification, which is the standardized scripting language that JavaScript is based on. ES6 was released in June 2015 and introduced many significant enhancements to the JavaScript language.

iii) ES2015 is known as ECMAScript 2015, refers to the version of the ECMAScript specification that was released in June 2015. It is the same version that is commonly referred to as ES6. The name "ES2015" is used to indicate that it was released in the year 2015.

THE DIFFERENCE:
ES5 is an earlier version of the ECMAScript specification with important features and improvements. ES6 or ES2015 represents a major update to the language, introducing a wide range of new features and syntax enhancements. While ES5 is widely supported by browsers, not all ES6 features are natively supported in older browsers, so additional tools like transpilers or polyfills may be required to use ES6 features in older environments.

---

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

i) JScripis Microsoft's legacy dialect of the ECMAScript standard that is used in Microsoft’s Internet Explorer. It was designed to provide scripting capabilities within Microsoft's products and platforms, primarily focusing on web development in Internet Explorer.

ii)ActionScript is a scripting language which was originally created for authoring animations and interactivity in Macromedia Flash, which was later acquired by Adobe. It became the primary language used to create Flash content, including games, interactive websites, and multimedia applications.

iii) ECMAScript is a standardized scripting language specification that defines the syntax, semantics, and behavior of a programming language. It serves as the foundation for several popular programming languages, with JavaScript being the most well-known and widely used implementation of ECMAScript.

RELATABILITY:
They are all relatable
JavaScript, JScript, and ActionScript are different implementations or variations of ECMAScript. JavaScript is the most widely used implementation, primarily running in web browsers, while JScript is Microsoft's implementation and ActionScript is used within Adobe Flash applications. Despite their slight differences in syntax and additional features, they are all rooted in the ECMAScript standard and share a common foundation.

---

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

---

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

They are all JavaScript compilers.
i) V8 is a free and open source, JavaScript and WebAssembly engine developed by the Chromium Project for Chromium and Google Chrome web browsers.

ii) SpiderMonkey is an open source Javascript and WebAssembly engine developed by Mozilla for the Firefox web browser. It is an open-source JavaScript engine written in C++. SpiderMonkey is responsible for executing JavaScript code in the browser and providing the necessary runtime environment for JavaScript applications to run.

iii) Chakra is a proprietary JScript engine developed by Microsoft, It is primarily used in Microsoft Edge web browser, but it has also been utilized in other Microsoft products and projects.

IV) Tamarin is an open-source virtual machine and runtime environment for executing programs written in the ActionScript and ECMAScript (JavaScript) programming languages. It was initially developed by Adobe Systems as a part of the Adobe Flash Player project.
Yes, the V8, SpiderMonkey, Chakra, and Tamarin JavaScript engines do have differences in how they execute JavaScript code. While they all aim to adhere to the ECMAScript specification and provide the expected behavior.

---

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