**🔷 What is a Polyfill in JavaScript?**

A **polyfill** is **a piece of code (usually JavaScript)** that **adds support for a feature that doesn't exist in older browsers**.

In simple terms:  
**Polyfill is a fallback.** It mimics a newer browser feature so your code still works in older environments.

## 🔹 Why Do We Need Polyfills?

Modern JavaScript (ES6 and beyond) introduces new features like:

* Array.prototype.includes
* Promise
* fetch
* Object.assign
* Map, Set, etc.

Older browsers (like Internet Explorer or old mobile browsers) don’t support these.

To make your code compatible with those browsers, you use **polyfills**.

**✅ Example: Polyfill for Array.prototype.includes**

Modern JavaScript:

js

CopyEdit

let fruits = ["apple", "banana", "cherry"];

console.log(fruits.includes("banana")); // true

Older browsers like IE don’t support includes. Here's a polyfill:

js

CopyEdit

if (!Array.prototype.includes) {

Array.prototype.includes = function(searchElement) {

return this.indexOf(searchElement) !== -1;

};

}

Now even in old browsers, you can use .includes().

**🔹 How Polyfills Work:**

* They check if a feature exists.
* If it doesn't, they define it using older supported syntax.

## 🔍 Real-Life Analogy

Imagine an old mobile charger that doesn't support fast charging.  
You buy an adapter (polyfill) to make it work with modern fast-charging phones.

## ✅ Summary

| **Term** | **Meaning** |
| --- | --- |
| **Polyfill** | JavaScript code that adds missing browser functionality |
| **Purpose** | Makes modern features work in older environments |
| **Example** | Adding Array.prototype.includes support in IE |

A **polyfill in JavaScript** is a script that adds modern features to older browsers that do not natively support them. To use it, include the polyfill script in your HTML or install it via a package manager, ensuring compatibility with older environments.

In older browsers, the following features require polyfill support by explicitly defining the functions:

Promises, Array.from, Array.includes, Array.of, Map, Set, Symbol, object.values, etc

Polyfills are essential for maintaining cross-browser compatibility when using modern JavaScript features. They help ensure that your code works seamlessly across both modern and older environments, providing a consistent user experience