**JavaScript**

1. **Introduction to JavaScript**

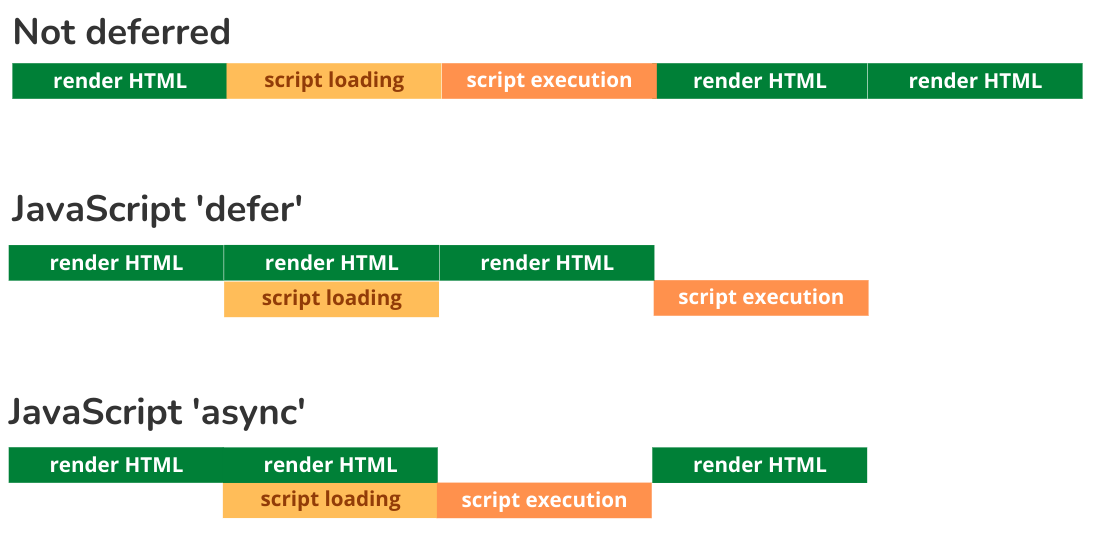
JavaScript is a programming language used to create dynamic content for websites. It is a lightweight, cross-platform, and single-threaded programming language. It's an interpreted language that executes code line by line, providing more flexibility**.** The data type of the variable is decided at run-time in JavaScript that’s why it is called dynamically typed.

* JavaScript is definitely a programming language because it has all the features of one - it supports variables, control structures (loops, conditionals), functions, objects, and can be used to write complex applications.
* JavaScript is **JIT-compiled** - a mix of interpretation and compilation done dynamically.

1. **Use of JavaScript**
2. Use in web development (form validation, dynamic content)
3. Use in back-end development (Handle routes, process form data, and manage databases)
4. API and Microservices (Create APIs that other services or apps can talk to)
5. Real-Time Apps (Enable chat apps, gaming platforms, or live data dashboards)
6. Mobile Apps (Build native mobile apps using frameworks like React Native)
7. **Ways to include JavaScript**
8. Inline JavaScript
9. Internal JavaScript (Within <script> Tag)
10. External JavaScript (Using External File)

* Best practice is to put your <script> in the <head> with defer, or at the end of the <body>.

**Loading strategies in JavaScript:**



1. **Syntax of JavaScript**
2. Variables: In JavaScript, variables are used to store data values. JavaScript uses the keywords var, let and const to declare variables. An equal sign is used to assign values to variables.

Example:

let x, y;

x = 5 + 6;

y = x \* 10;

var a, b;

a = 5 + 6;

b = a \* 10;

const PI = 3.14;

1. Operators: JavaScript uses arithmetic operators ( + - \* / ) to compute values. JavaScript uses an assignment operator ( = ) to assign values to variables.
2. Expressions: An expression is a combination of values, variables, and operators, which computes to a value.
3. Comments: Code after double slashes // or between /\* and \*/ is treated as a comment.
4. Identifiers: Identifiers are used to name variables and keywords, and functions.

A JavaScript name must begin with:

* A letter (A-Z or a-z)
* A dollar sign ($)
* Or an underscore (\_)

1. **Basic Event in JavaScript**

JavaScript Events are actions or occurrences that happen in the browser. They can be triggered by various user interactions or by the browser itself.

JavaScript supports a variety of event. Common events are:

1. onclick: Triggered when an element is clicked.

<button onclick="alert('Hello World')">Click</button>

1. onmouseover: Fired when the mouse pointer moves over an element.

<button onmouseover="this.style.background = 'red'">Hover here</button>

1. onmouseout: Occurs when the mouse pointer leaves an element.

<button onmouseout="this.style.background = 'green'">Mouse out</button>

1. onkeydown: Fired when a key is pressed down.

<input type="text" onkeydown="this.style.background = 'yellow'" placeholder="Enter your name">

1. onkeyup: Fired when a key is released.

<input type="text" onkeyup="this.style.background = 'gray'" placeholder="Enter your email">

1. onchange: Triggered when the value of an input element changes.

<input type="radio" onchange="alert('Selected Male')" id="male" name="gender" value="male">

<label for="male">Male</label>

<input type="radio" onchange="alert('Selected Female')" id="female" name="gender" value="female">

<label for="female">Female</label>

1. onload: Occurs when a page has finished loading.

<body onload="alert('Page loaded')">

1. onsubmit: Fired when a form is submitted.

<form action="#" onsubmit="alert('Form submitted')">

<button type="submit">Submit</button>

</form>

1. onfocus: Occurs when an element gets focus.

<input type="text" onfocus="this.style.background = 'aqua'" placeholder="Focus here">

1. onblur: Fired when an element loses focus.

<input type="text" onblur="this.style.background = 'black'" placeholder="Focus and go">

1. **Basic Validation in JavaScript**

JavaScript form validation checks user input before submitting the form to ensure its correct. It helps catch errors and improves the user experience.

function validateForm(){

let name = document.forms["myForm"]["name"].value;

let age = document.forms["myForm"]["age"].value;

if(name==""){

alert("Name must be filled");

return false;

}

if(age<=0 || age>150){

alert("Enter valid age");

return false;

}

}