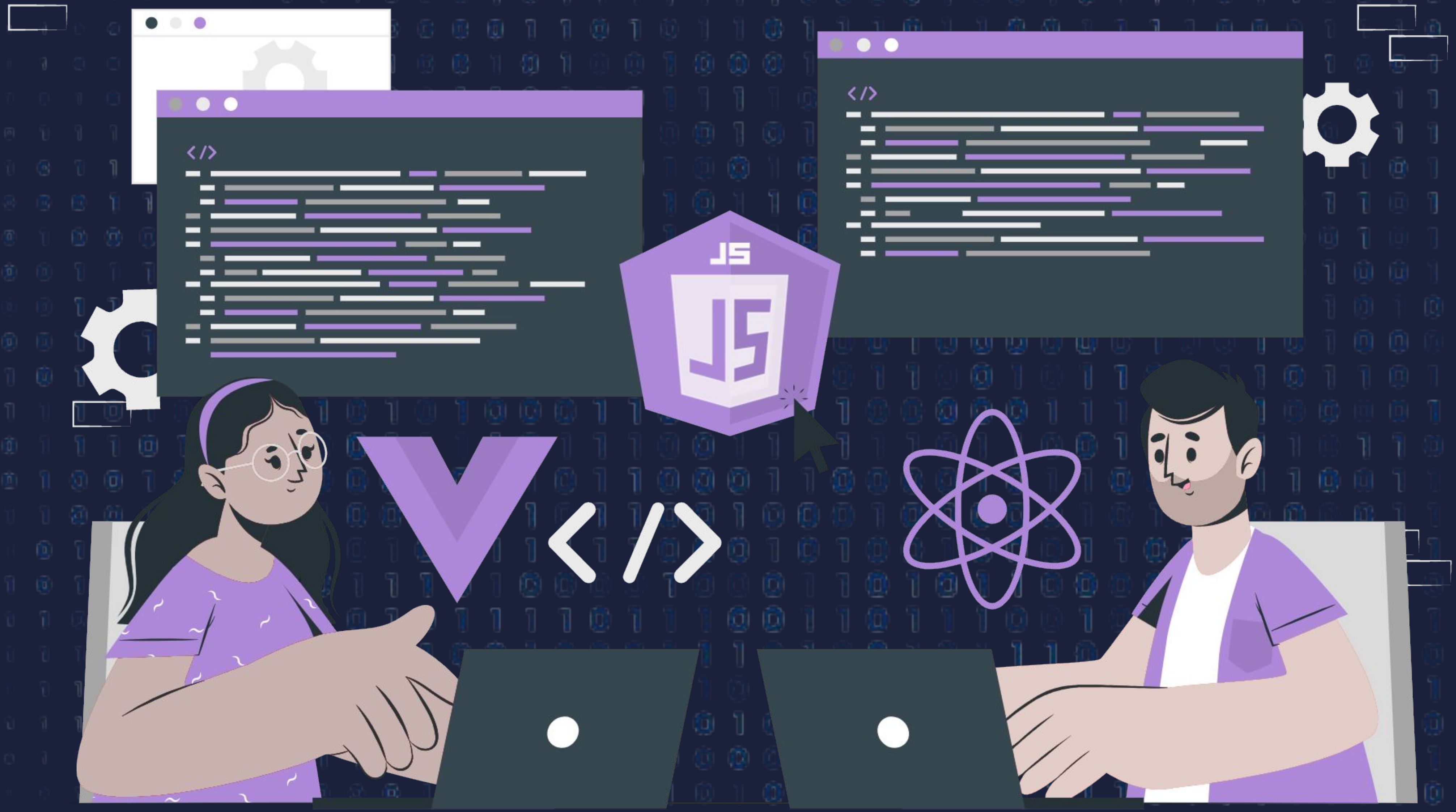
PW SKILLS

Single Thread



Lecture CheckList

1. Introduction to threads.
2. Classification of programming languages based on threads.
3. Javascript is a single-threaded programming language.
4. Is being single-threaded a limitation in Javascript?
5. Is Javascript slow?

Introduction to threads

Imagine a company that needs to process customer orders. One person could be responsible for receiving the order and entering it into the system, while another could prepare the order for shipping. These tasks could be performed independently, and each person would be a separate "thread" of work.

In the same way in programming, a thread refers to a sequence of instructions that a computer can execute independently from other sequences of instructions in the same program. Think of a thread as a single unit of work or a single task that a computer can perform.

Introduction to threads

Threads are important because they allow programs to perform multiple tasks at the same time, which can improve the program's performance and responsiveness. For example, a web browser might use multiple threads to download web pages, render graphics, and respond to user input, all at the same time.

In the above-mentioned example, if the company only had one person to handle all the orders, the process would be slower and less efficient. But by having multiple people working on different aspects of the same task simultaneously, the company can process more orders in less time.

Similarly, in programming, using multiple threads to perform different tasks can improve the efficiency and speed of a program. However, it's important to manage these threads carefully to avoid issues.

Classification of programming languages

Programming languages can be categorized as single-threaded or multi-threaded, depending on how they handle concurrent execution.

Javascript is a single-threaded programming language

JavaScript is primarily a single-threaded programming language, meaning that it can only execute one sequence of instructions at a time.

This is because the JavaScript engine, which is responsible for interpreting and executing JavaScript code, uses a single call stack to manage the sequence of instructions that need to be executed. We will look into call stack in upcoming lectures.

Is being single-threaded a limitation in Javascript?

Being single-threaded is both a limitation and a benefit for JavaScript, depending on the situation and the alternatives available.

As a limitation, a single-threaded language can only execute one piece of code at a time. To avoid this, JavaScript supports asynchronous programming through features like callbacks, promises, and async/await. These features allow for non-blocking I/O operations, which means that the JavaScript engine can continue executing other instructions while waiting for I/O operations to complete. This can help to avoid blocking the main thread and keep the application responsive.

The benefit of being single-threaded is that it simplifies the programming model, making it easier to reason about program behavior and preventing complex errors that can occur in multi-threaded applications. JavaScript event loop, which manages the execution of code, provides a simple and efficient way to handle asynchronous operations, such as I/O, without the need for explicit threading.

Is Javascript slow?

No, being single-threaded does not necessarily mean that JavaScript is slow. Single-threaded simply means that JavaScript can only execute one task at a time, but it does not decide the speed of the language.

In fact, JavaScript can be quite fast due to modern JavaScript engines, such as V8 used in Chrome and Node.js, which have advanced optimization techniques. These techniques allow JavaScript code to be executed at decent speeds, which is why JavaScript has become a popular language for both client-side and server-side applications.

SKILLS

THANK YOU