## **Class Plan ( JS - Functions (Arrow Functions, Default Parameters, Callback Functions, Higher-Order Functions, Closures))**

### **Digital Excellence Program – Sample Class Plan**

**Topic: JS - Functions part 2**

| **Time Slot** | **Activity** | **Description** |
| --- | --- | --- |
| **First 5 minutes** | **Warm-up & Reflection** | * Quick recap of previous class. |
| **30 minutes** | **Core Concept Explanation** | **Arrow functions and defaults**  * Full syntax and behavior of **arrow functions**   + One-liners vs block-body   + Implicit return   + Difference in this binding * **Default parameters**: how to set fallback values * Example comparisons:   + Regular function vs arrow function   + Function with missing arguments vs using default values * Gotchas to highlight:   + No this in arrow functions   + Can't use arguments object in arrow functions   **Call backs, higher order functions and closures**  * What is a **callback** function? (Pass function as an argument) * Common example: setTimeout, forEach * What is a **higher-order function**? * Takes a function or returns one * Examples: map, filter, reduce * Simple visual explanation of how functions are treated as values in JS  **Code Walkthroughs:**    + sayHello(callback   + [1, 2, 3].map(n => n \* 2)   + createMultiplier(factor) returns a function * What is a **closure**?   + What is lexical scope?   + A closure is when a function “remembers” its lexical scope even when the outer function has finished execution. A closure happens when a function "closes over" variables in its **lexical scope** — even after the outer function has returned.   + Demonstrate with examples |
| **30 minutes** | **Activity** | Guess the Output activity attached in micro projects document. |
| **10 minutes** | **Class Wrap-up** | Recap of key concepts. Questions to check their understanding:   * What’s the main difference between a regular and arrow function? * Can you return a function from another function? When might that be useful? * Can you think of a real-life use case where a function remembering a value would be helpful? |

### **References & Notes for JS - Comparison, Logical and Bitwise Operators**

#### 

#### **Core Concept Explanation**

**References:**

* [**Arrow functions**](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Functions/Arrow_functions)
* [**Closures**](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Closures)