FAQ

Module-3	
Question 1.	**What are the primary differences between var, let, and const inJavaScript?**
Answer	The variable "var" is scoped to the function and may be redeclared. The keyword "let" in JavaScript is block-scoped, meaning it is only accessible inside the block of code it is defined in. While it may be reassigned to a newvalue, it cannot be redeclared within the same scope. The 'const' keyword is similarly block-scoped, meaning that it is only accessible inside the block of code where it is defined. However, once a value is set to a 'const' variable, itcannot be modified or reassigned.
Question 2.	**How does a loop differ from a conditional in JavaScript?**
Answer	A loop is a control structure that iteratively runs a designated piece of codewhile a certain condition remains true. In contrast, a conditional statement, such as an if statement, is designed to run a certain piece of code alone whena particular condition is

	satisfied.
Question 3.	**Why are data types important in JavaScript?**
Answer	Data types are a fundamental aspect of computing that dictate the permissibleoperations that may be executed on a given set of data. Developers may achieve efficient and error-free code by acquiring a comprehensive grasp ofdata types. This knowledge enables them to ensure that operations such as arithmetic or comparisons function as intended.
Question 4.	**What are arrow functions, and how do they differ from traditionalfunctions?**
Answer	Arrow functions provide a succinct syntax for defining functions in ECMAScript 6 (ES6). The syntax used by the aforementioned entity is of a more concise nature, and it lacks the presence of an independent binding for the "this" keyword. Arrow functions in JavaScript inherit the value of "this" from the encompassing non-arrow function.
Question 5.	**How does scope influence variable access in JavaScript functions?**

Answer	The accessibility and longevity of variables in JavaScript are determined by their scope. In programming, variables that are declared inside a function are considered to have a local scope, meaning that they cannot be accessed or used outside of that specific function. On the other hand, variables that are declared outside of any functions, also known as having a global scope, are accessible and may be used across the whole programme. A comprehensive comprehension of scope aids in the prevention of inadvertent alterations and conflicts among variables.