**Browser**

**Node JS**

**We Need Environment to Execute Javascript Codes**

**Javascript Engine**

**Parser**

y0

**Convert To Machine Code**

y0

**Finally Output**

**Hi Md. Al-Amin**

**var info = ‘Finally Output Hi Md. Al-Amin’**

**console.log(info);**

y0

**RUN**

**Javascript**

* **Execution Context**
* **Creational Phase**
* **Execution Phase**
* **Hoisting**
* **Execution Context Stack**
* **Scope**
* **Scope Chain**
* **Lexical scoping**
* **“This” Variable**
* **Closures**
* **Function Declaration**
* **Function Expression**
* **First Class Function**
* **Higher Order Function**
* **Inner Function**
* **Outer Function**
* **Return Function**
* **Callback Function**
* **ForEach Function**
* **Filter Function**
* **Map Function**
* **Reduce Function**
* **Sort Array**
* **Anonymous Function**
* **Object**
* **Array**
* **Execution Context object:**

1. Variable object
2. Scope chain
3. “This” variable

* **When a function get call there create a new execution context and it happens in tow step.**

1. Creational Phase
2. Execution Phase

* **Creational Phase :**

1. Create Variable Object
2. Create Scope Chain

* **Execution Phase :**

1. Determine Value of “this”
2. Read & Run line by line
3. Define All Variables

**What does it mean By Hoisting ?**

Hoisting is a JavaScript mechanism where variables and function declarations are moved to the top of their scope before code execution.

**What is a Function Declaration?**

A Function Declaration defines a named function variable without requiring variable assignment. Function Declarations occur as standalone constructs and cannot be nested within non-function blocks. It’s helpful to think of them as siblings of Variable Declarations. Just as Variable Declarations must start with “var”, Function Declarations must begin with “function”.

**What is a Function Expression?**

A Function Expression defines a function as a part of a larger expression syntax (typically a variable assignment). Functions defined via Functions Expressions can be named or anonymous. Function Expressions must not start with “function” (hence the parentheses around the self invoking example below)

Reff: <https://javascriptweblog.wordpress.com/2010/07/06/function-declarations-vs-function-expressions/>

**What is a scope in JavaScript?**

JavaScript has two scopes – global and local. Any variable declared outside of a function belongs to the global scope, and is therefore accessible from anywhere in your code. Each function has its own scope, and any variable declared within that function is only accessible from that function and any nested functions