/\*

  To Understand Ajax, Fetch, Promises

  Call Stack || Stack Trace

  - JavaScript Engine uses a call Stack To Manage Execution Contexts

  - Mechanism to make the interpreter Track Your Calls

  - When Function called it Added To the Stack

  - Whne function executed it removed From the Stack

  - after Function Is finished executing the interpereter Continue From the Last point

  - Work Using LIFO principle => last in First Out

  - Code Exection Is Synchronous.

  - Call Stack Detect Web API Methods And Leave It To The Browser To Handle It

  Web API

  - Mehtods Available From the Environment => Browser

\*/

setTimeout(() => {

  console.log("Web API");

}, 0);

function one() {

  console.log("One");

}

function two() {

  one();

  console.log("two");

}

function three() {

  two();

  console.log("three");

}

console.log("#####");

console.log("One");

console.log("Two");

console.log("Three");