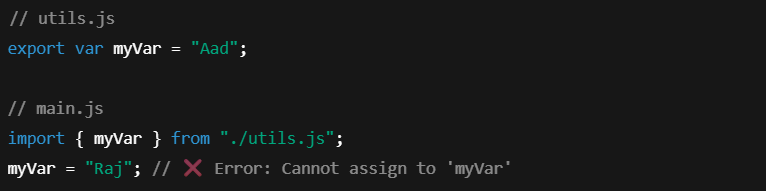
**Assignment – 6**

**Question 1: Output if var is changed in the module, imported in the main file**

* In ES Modules, you cannot reassign a variable imported from another file.
* The import is a read-only live binding.
* Reassigning it will throw a runtime error.

**Code:**

****

**Question 2: Difference between Asynchronous and Synchronous, Strict and Non-Strict in .mjs**

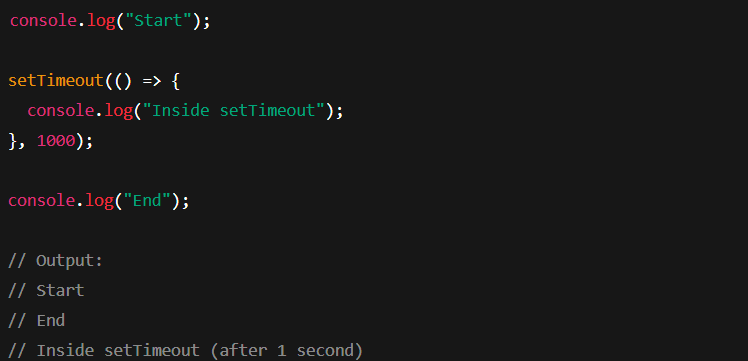
**Synchronous:**

* Code runs step by step.
* Each line waits for the previous one to finish.



**Asynchronous:**

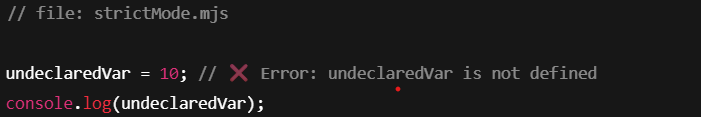
* Code doesn’t wait.
* Tasks like setTimeout, fetch, or promises run in the background.



setTimeout() runs in background — doesn't block the next line.

**Strict Mode (.mjs):**

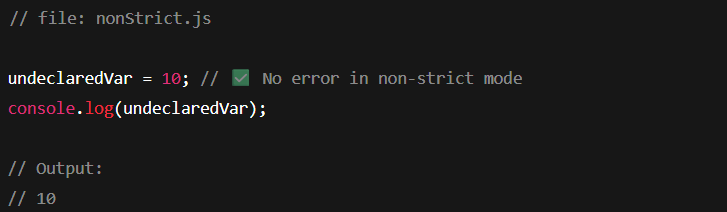
* All .mjs files run in strict mode by default.
* It disallows undeclared variables and throws more errors.
* .mjs files use strict mode **automatically** — no need to write "use strict".



Throws ReferenceError because variables must be declared (e.g., using let or var).

**Non-Strict Mode:**

* In regular .js files, strict mode is optional.
* It allows sloppy behaviors like undeclared variables.



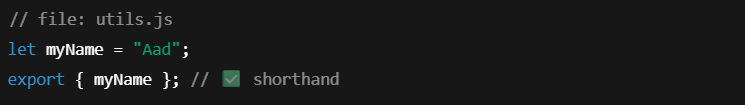
In non-strict mode, JavaScript allows undeclared variables — but it's risky and not recommended.

**Question 3: Difference between myName and myName: myName in Module.**

**🔹 myName**

* This is the **shorthand syntax** in ES6 when the **variable name and the export key are the same**.
* Cleaner and shorter.

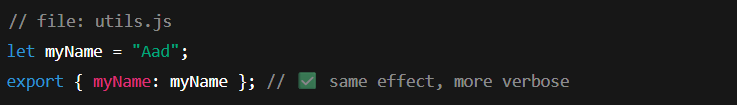
**Code:**



**🔹 myName: myName**

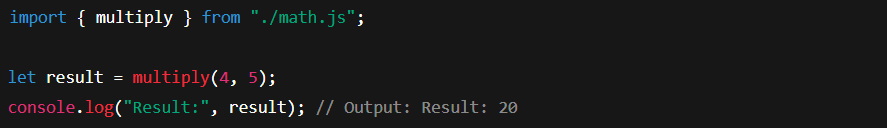
* This is the **explicit syntax**, where the left is the **exported name**, and the right is the **variable name**.
* Useful if you want to **rename** the export.

**Code:**



**Question 4: Do some logic in index.js**

**Code:**

****