Variables, Data Types & Template Literals

AiCE Software Engineering Foundations — JavaScript & Node.js

Node.js is installed and your first file ran successfully. This lesson covers how JavaScript actually stores information — declaring variables, the core data types, and template literals, the modern way to build text with variables embedded inside it.

1. Declaring Variables — let, const & var

From Python, you already know what a variable is and how variables work — JavaScript works almost identically. The only difference is how you declare one. In Python:

```
name = "Adesina"
```

In JavaScript, add a keyword called let before the variable name:

```
let name = "Adesina"
```

let is a powerful keyword used to declare a variable. There are two other keywords used to declare variables in JavaScript as well:

```
let name = "Adesina"
var isStudent = true
const age = 26
```

> **Forget var**
> *You'll see var in older code, but modern JavaScript avoids it — it lets you accidentally redeclare the same variable with no error. This course only uses let and const.*

Create a file named variables.js, and declare a few more:

```
let name = "Adesina"
let isStudent = true
const age = 26
let friendName = "Goodnews"
const friendCity = "Lagos"
let friendAge = 25
```

Keyword	When to Use It
const	For values that never change.
let	For values that will be reassigned later.

Save and run this file, and nothing displays — the variables were declared, but nothing was written to display them yet. Add console.log() for each:

```
console.log(name)
console.log(isStudent)
console.log(age)
console.log(friendName)
console.log(friendCity)
console.log(friendAge)
```

Save and run again — every value now prints.

2. Data Types & typeof

Basic data types in JavaScript are: String, Number, Boolean, Undefined, and null. Create a file named datatype.js:

```
const firstName = "Adesina";
const age = 26;
const score = 98.5;
const isEnrolled = true;
let phoneNumber;
let address = null;
```

```
console.log(typeof firstName);
console.log(typeof age);
console.log(typeof score);
console.log(typeof isEnrolled);
console.log(typeof phoneNumber);
console.log(typeof address);
```

typeof is JavaScript's version of Python's type() — it tells you the data type of a value. Save and run:

Variable	typeof Result	Note
firstName	string	Same as Python — text is a string.
age, score	number	In Python, age is an int and score is a float. JavaScript has only one type, Number, for both.
isEnrolled	boolean	true and false are written entirely in lowercase in JavaScript, unlike Python's True and False.
phoneNumber	undefined	Declared but never given a value — its data type is undefined.
address	object	Set to null, a special value meaning intentionally empty; JavaScript reports it as an object.

3. Template Literals — JavaScript's f-Strings

Template literals in JavaScript do the same job as f-strings in Python. Python uses f-strings to embed variables in text; JavaScript uses template literals — strings wrapped in backticks ` with variables inserted using ${}.

Create a file named template_literals.js:

```
const name = "Kwame";
const course = "JavaScript";

console.log(`Hello, ${name}! You are learning ${course}.`);
console.log(`Next year you will know ${course} and
TypeScript.`);
```

Save and run — each ${} is replaced with the value of the variable inside it.

4. Practice: A Student Progress Printer

Let's build a small project combining everything from this lesson. Create a file named goodnews.js:

```
const studentId = 1;
const firstName = "Goodnews";
const country = "Nigeria";
let progressPercentage = 45;

console.log("========================================");
console.log(`ID: ${studentId} | Name: ${firstName}`);
console.log(`Country: ${country} | Progress:
${progressPercentage}%`);

progressPercentage = 60; // this is why we used let
console.log(`Updated progress: ${progressPercentage}%`);
console.log("========================================");
```

Save and run — this prints a formatted profile block, then updates and reprints the progress percentage, exactly the reason progressPercentage was declared with let instead of const. Push everything built so far to GitHub.

Key Takeaways

Variables are declared with let for values that change and const for values that don't — var is left behind entirely. JavaScript's five basic data types are String, Number, Boolean,

Undefined, and null, checked with typeof, and JavaScript merges Python's int and float into a single Number type. Template literals — backticks with ${} — build text with embedded variables the same way f-strings do in Python. With the basics in place, the next lesson looks at something Python simply doesn't do: type coercion, JavaScript's automatic conversion between types when they mix together, along with a deeper look at strings and numbers.