JavaScript variables

Variables are containers for storing Information or Data

```
Example:
```

var a =10
var b=20
console.log(a+b)

In the above program
a and b are variables,10 and 20 are integer data

- JavaScript variables can be declared in 3 ways.
  1.var
  2.let
  3.const

1. **var:**
   - `var` was the original way to declare variables in JavaScript.
   - It has function scope, which means it's limited to the function in which it is declared.
   - Variables declared with `var` are hoisted to the top of their function or global context.
   - `var` can be redeclared within the same scope, and its value can be changed.
2. **const:**
   - `const` is short for "constant" and is used to declare variables that should not be reassigned after their initial value is set.
   - It has block scope, meaning it's limited to the block or function where it's declared.
   - A `const` variable must be assigned a value when declared and cannot be reassigned.
   - It's commonly used for values that shouldn't change, like configuration settings or fixed values.
3. **let:**
   - `let` was introduced to provide block-scoped variables in JavaScript.
   - It also has block scope, making it suitable for limiting the variable's visibility to a specific block or function.
   - Unlike `const`, `let` can be reassigned a new value.
   - It's often used for variables that need to change their values within a specific scope.

Using var keyword - function scope or global scope

`Example:-
var x =10;
{var y=20;}
console.log(y)

```
Output:- 20
```

Using let keyword - block scope

- The let Keyword introduced in 2015.
- Let Keyword Must be declared before use.
- Let Keyword can not be Redeclared.
- Let Keyword Have block scope.

`Example:-`
let keyword can't do this
let num = "Let Keyword"
let num = "100"

let keyword has block scope
{
let x = 2;
}
//X cannot used be used here

Using const keyword - block scope

- In JavaScript, the **const** keyword is used to declare a constant variable. A constant variable , once assigned a value, cannot be reassigned or redeclared. It provides a way to create variables that meant to be immutable.

`Example:-`

const keyword cannot do this
const num = 10
num = 20

Exercise 1

1. Create a variable named "Price" and set it to a specific price value.
2. Create a variable named "Product" and set it to a product name.
3. Create a variable named "Tax" and set it to a specific tax value.
4. Print the name of the product.
5. Calculate the total price amount, including tax(total=price+tax), and then print it.

`solution:-`

```
var price=15000
var product="iPhone"
var tax=200
var total=price+tax

console.log(product)
console.log("Total price amount is", total)
```
