Assignment (Session 1)

Things to remember before attempting this assignment:

Important Points for **var :**

- The **var** is either a function-scoped or globally-scoped variable.

- Because **var** declarations are processed before any code is executed, declaring a variable anywhere in the code is equivalent to declaring it at the top. This also means that a variable can appear to be used before it's declared. This behaviour is called **hoisting**, as it appears that the variable declaration is moved to the top of the function or global code.

- It's important to point out that only a variable's declaration is hoisted, not its initialization. The initialization happens only when the assignment statement is reached. Until then the value of that variable remains undefined.

Important Points for **let**

- The let declaration declares a block-scoped local variable.
- Example of let

```
let x = 1;
if (true) {
  let x = 2;
  console.log(x);
  // Expected output: 2
}
console.log(x);
// Expected output: 1
```

Important Example for understanding difference between let and var.

```
function varTest() {
  var x = 1;
  {
    var x = 2; // same variable!
    console.log(x); // 2
  }
  console.log(x); // 2
}

function letTest() {
  let x = 1;
  {
    let x = 2; // different variable
    console.log(x); // 2
  }
  console.log(x); // 1
}
```

>What will be output of the following programs:

1>

```
console.log(x);
var x = 5;
console.log(x);
```

2>

```
var x = 5;
if (true) {
  var x = 10;
  console.log(x);
}
console.log(x);
```

3>

```
console.log(x);
let x = 5;
console.log(x);
```

4>

```
var x = 5;
function foo() {
  console.log(x);
  var x = 10;
}
foo();
```

5>

```
console.log(x);
var x = 5;
function foo() {
  console.log(x);
  var x = 10;
}
foo();
console.log(x);
```

6>

```
function foo() {
  console.log(x);
  let x = 10;
}
foo();
```