Scope:

scope determines the accessibility or visibility of the variable.

Closures:

Closures are inner functions that have access to external variables.
By creating a function inside a function, we create a closure.

In javascript, there are three different types of scope:
1)Local
2)Global
3)Block

In Local: variable, function, object is only available to a local function.
In Global: variable, function, object is available to all.
In Block scope variables, which are declared with let and const.they are available to the innermost block they are surrounded by.

Hoisting:

Hoisting in javascript means that moving all the declaration to the top of its scope.

let's say we have

Example1:

```
name ="shrinivas" // at line1
console.log(name); // "shrinivas"
.
.
.
var name;// at somewhere in the program, may be at line 20.
```

even though we have declared a variable at line 20 but it will be moved to the top of the scope and it will get assigned with value. var variables are hoisted.

Example 2:

```
console.log(name);// "shrinivas"
var name = "shrinivas";
console.log(name);// "shrinivas"
```

Example3:

```
console.log(age); // Uncaught ReferenceError: age is not
                     defined
age = 10; // here value is assigned to variable without var,let, const
console.log(age); // 10
```

Global scope Variables:

global variables are any functions, objects, variables that are declared outside a function.
They are accessible throughout the file.
Two drawbacks:

if we don't declare a variable and if we assign a value to a variable inside of a function then it automatically becomes globally available.
If we use strict mode then they will not become automatically global and it will give errors at the line where the variable is assigned a value.

From practice conclusion:

Arrow functions are not hoisted which can be declared with var, let, const and without var, let, const as well.

Normal functions are hoisted.

Local scope Variables:

local variables can be declared with var, let, const.
variables which are declared inside a function are local variables.
multiple functions can have the same variable name inside their function.

when we declare a variable with var inside a function then also this variable is not available outside the function.
If we try to access it then we get the error variable is not defined.

Block Variables:

block scope variables use let and const keywords.
let is mutable, const is immutable.

Don't get confuse: A function scope variables can be block scope variables whereas
A block scope variable may or may not be a function scope variable.