Get ready for an exciting adventure into the world of JavaScript closures! By mastering closures through simple exercises, you’ll level up your JavaScript skills and discover their true potential. So, let’s dive in and explore closures together!

**If you don not have understanding of Closures, I highly recommend to go through below articles first**.

- [Part 11: Closures in JS 🤷🏻‍♀️](https://medium.com/@swati.developer17/part-11-closures-in-js-%EF%B8%8F-e0874a86b2de)

- [Part 12: setTimeout + Closures Interview Questions 🤓](https://medium.com/@swati.developer17/part-12-settimeout-closures-interview-questions-cfa0d402831f)

[-Part 13: Closure JS Interview Questions 🤯](https://medium.com/@swati.developer17/part-13-closure-js-interview-questions-f31c1be77d3e)

Let’s get started.

function outerFunction() {  
 var outerVariable = "Hi, ";  
 function innerFunction(name) {  
 console.log(outerVariable + name); // What is logged?  
 }  
 return innerFunction;  
}  
  
var inner = outerFunction();  
inner("Closure");

=================================================================

function outer() {  
 var x = 10;  
 function inner() {  
 console.log(x); // What is logged?  
 }  
 return inner;  
}  
var innerFunc = outer();  
innerFunc();

=========================================================================

function outer() {  
 var x = 10;  
 function inner() {  
 console.log(x); // What is logged?  
 }  
 var x = 20;  
 return inner;  
}  
var innerFunc = outer();  
innerFunc();

==========================================================================

function outer() {  
 var x = 10;  
 function inner() {  
 var y = 12;  
 console.log(x+y); // What is logged?  
 }  
 return inner;  
}  
var innerFunc = outer();  
innerFunc();

=========================================================================

var a = 1;   
function b() {   
 a = 10;   
 return;   
 function a() {}   
}   
b();   
console.log(a); // What is logged?  
======================================================

function test() {  
 console.log(a); // What is logged?  
 console.log(foo()); // What is logged?  
   
 var a = 1;  
 function foo() {  
 return 2;  
 }  
}  
  
test();

==========================================================================

var a = 1;   
  
function someFunction(number) {  
 function otherFunction(input) {  
 return a;  
 }  
   
 a = 5;  
   
 return otherFunction;  
}  
  
var firstResult = someFunction(19); // What is firstResult?  
var result = firstResult(12); // What is result?

==================================================================

var fullname = 'Daisy Doe';  
var obj = {  
 fullname: 'Vera Singh',  
 prop: {  
 fullname: 'Amar Akbar Anthony',  
 getFullname: function() {  
 return this.fullname;  
 }  
 }  
};  
  
console.log(obj.prop.getFullname());  
  
var test = obj.prop.getFullname;  
  
console.log(test()); // What is logged?