```
  function inner() {
    console.log(outerVar); // inner can access outerVar even after outer function finished
  }

  return inner;
}

var closureFunction = outer();
closureFunction(); // Prints: "I am from outer function"
```

In this example, the `inner` function is returned from the `outer` function. The `inner` function forms a closure, retaining access to the `outerVar` variable even though `outer` has finished executing.

Python Closure Example:

```
def outer():
    outer_var = "I am from outer function"

    def inner():
        print(outer_var) # inner can access outer_var even after outer function finished

    return inner

closure_function = outer()
closure_function() # Prints: "I am from outer function"
```