PART 1

LAB SESSION

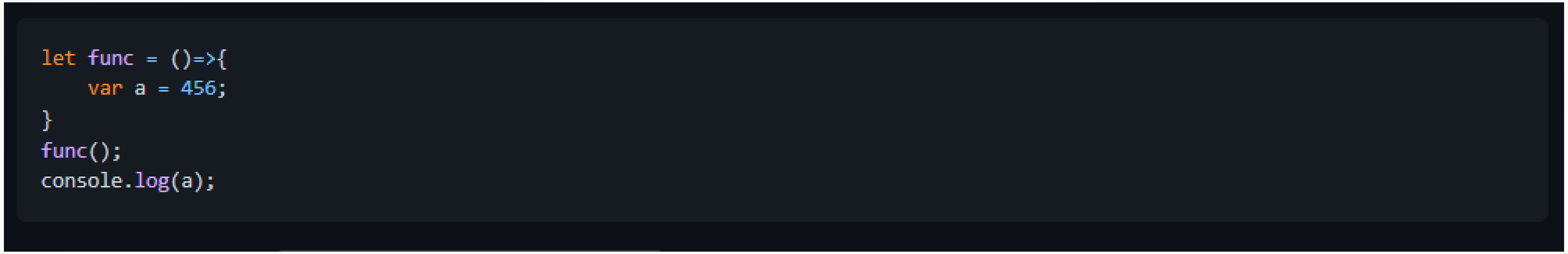
✓ Variables and Scopes

Lab 2 - Variables and Scopes

- The purpose for this lab is to illustrate the unintented consequences of using *var* or variables without keywords. This can create very elusive errors that is difficult to troubleshoot. Since Javascript compiler excepts variables declared using *var* and without keywords as valid, therefore the developer must make it a habit to delcare variables explicitly using *let* and *const* .

Lab 2a: Using var to declare local variable

1. Create the following in *2a-local-variable.js* then run using node *2a-local-variable.js*

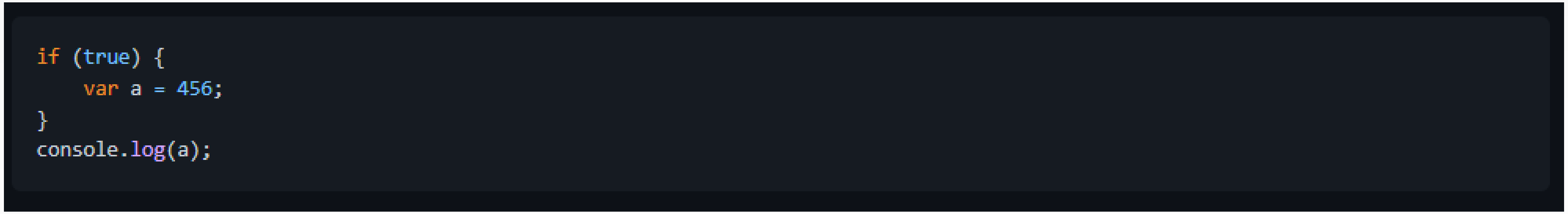


```
let func = ()=>{
    var a = 456;
}
func();
console.log(a);
```

This will throw an error *ReferenceError: a is not defined.* Why?

Lab 2b: Using var to unintentionally declare global variable

1. Create the following in *2b-global-variable.js* then run using *node 2b-global-variable.js*

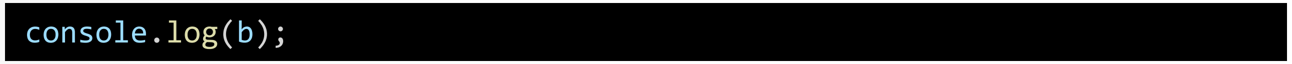


```
if (true) {
    var a = 456;
}
console.log(a);
```

This returns *456.* Why not an error?

Lab 2c: Uncaught errors when using undefined global variable

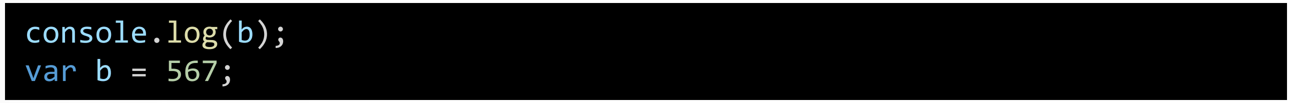
1. Create the following in *2c-hoisting.js* then run using *node 2c-hoisting.js*



```
console.log(b);
```

This should throw an exception *ReferenceError: b is not defined*

Change the file to the following then save and run again.



```
console.log(b);
var b = 567;
```

This did not throw an exception even though b is only declared after console.log(). Why?