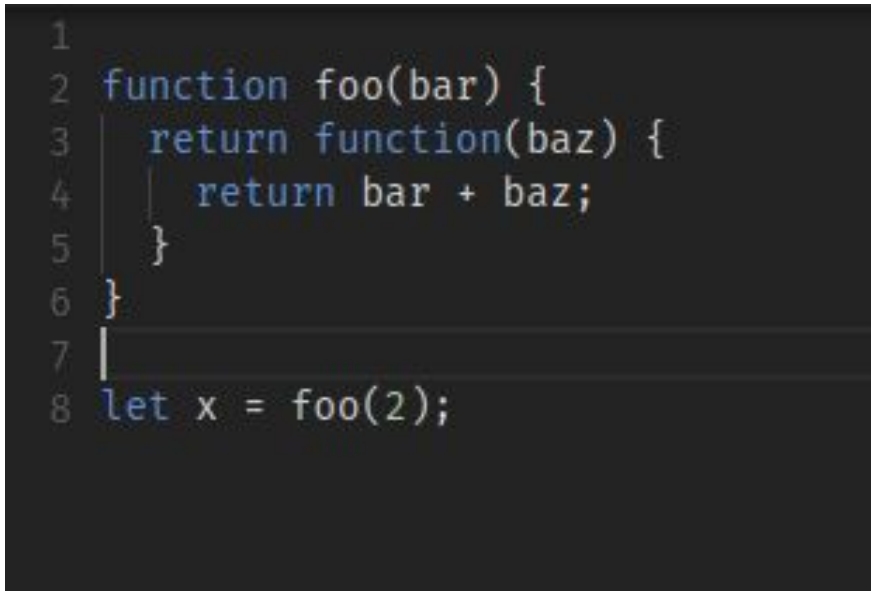
Discussion Section 7

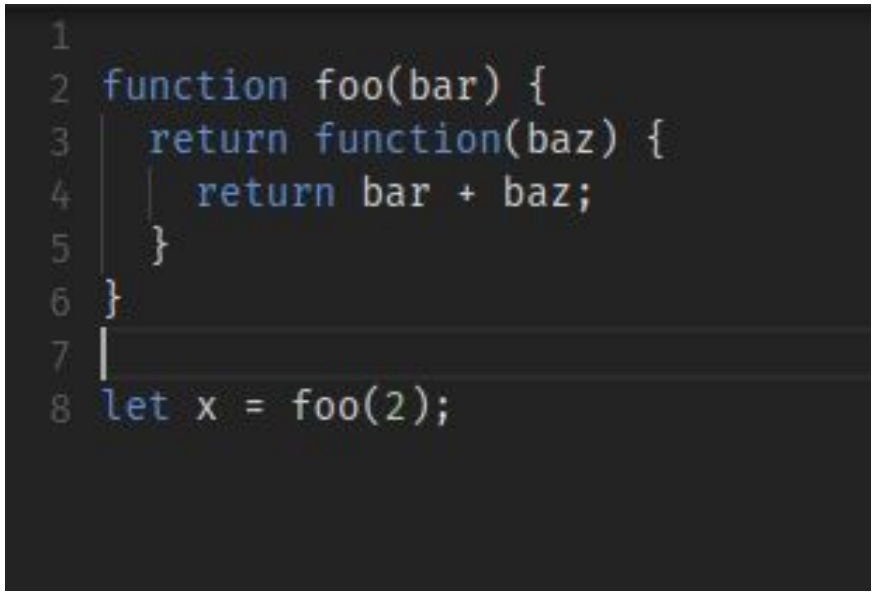
10/23 and 10/25

What is a Closure?

Find the Closure!



Find the Closure!



What is the type of f?

```
function foo(bar) {
  function baz(x) {
    return function(y){return bar + x};
  }

  let f = function(x){return 0;};
  for (let i = bar; i >= 0; --i) {
    let g = baz(f(i));
    f = g;
  }

  return f;
}

let f = foo(10);
console.log(f(10));
```

What is printed to the console?

```
function foo(bar) {
  function baz(x) {
    return function(y){return bar + x};
  }

  let f = function(x){return 0;};
  for (let i = bar; i >= 0; --i) {
    let g = baz(f(i));
    f = g;
  }

  return f;
}

let f = foo(10);
console.log(f(10));
```

What is printed to the console?

```
3 function foo() {
4   let z = {a: 3};
5   let f = function() { return 10 * z.a; }
6   z.a = 7;
7   return f;
8 }
9 console.log(foo()());
```

What is printed to the console?

```
3 function foo() {
4   let z = {a: 3};
5   let f = function() { return 10 * z.a; }
6   z.a = 7;
7   return f;
8 }
9 console.log(foo()());
```

What is printed to the console?

```
3 function foo() {
4   let z = 3;
5   let f = function() { return 10 * z; }
6   z = 7;
7   return f;
8 }
9 console.log(foo()());
```

What is printed to the console?

```
3 function foo() {
4   let z = 3;
5   let f = function() { return 10 * z; }
6   z = 7;
7   return f;
8 }
9 console.log(foo()());
```

Also 70!

Application of Closures: “Classes”

```
3  function rectangle(x, y, width, height) {
4    return {
5      x: x,
6      y: y,
7      width: width,
8      height: height
9    };
10 }
11
12 console.log(rectangle(2, 10, 100, 200).y);
```

Problem: The implementation of the “class” is not hidden: users can modify the returned rectangle at will. Maybe you don’t want this.

Application of Closures: “Classes”

```
3  function rectangle(x, y, width, height) {
4    return {
5      getX: function() { return x; },
6      getY: function() { return y; },
7      getWidth: function() { return width; },
8      getHeight: function() { return height; },
9    };
10 }
11
12 console.log(rectangle(2, 10, 100, 200).getY());
```

Fixed! x, y, width, and height are captured in the closure, are hidden from direct access by the user, but can still be returned from the accessor functions.

But we can do more than accessor functions!

Application of Closures: “Classes”

```
3  function rectangle(x, y, width, height) {
4    function getX() { return x; }
5    function getY() { return y; }
6    function setX(newX) { x = newX; };
7    function setY(newY) { y = newY; };
8    return {
9      getX: getX,
10     getY: getY,
11     getWidth: function() { return width; },
12     getHeight: function() { return height; },
13     moveBy: function(dx,dy) { setX(getX()+dx); setY(getY()+dy); }
14   };
15 }
16
17 let r = rectangle(2, 10, 100, 200);
18 r.moveBy(5, 8);
19 console.log(r.getX());
```