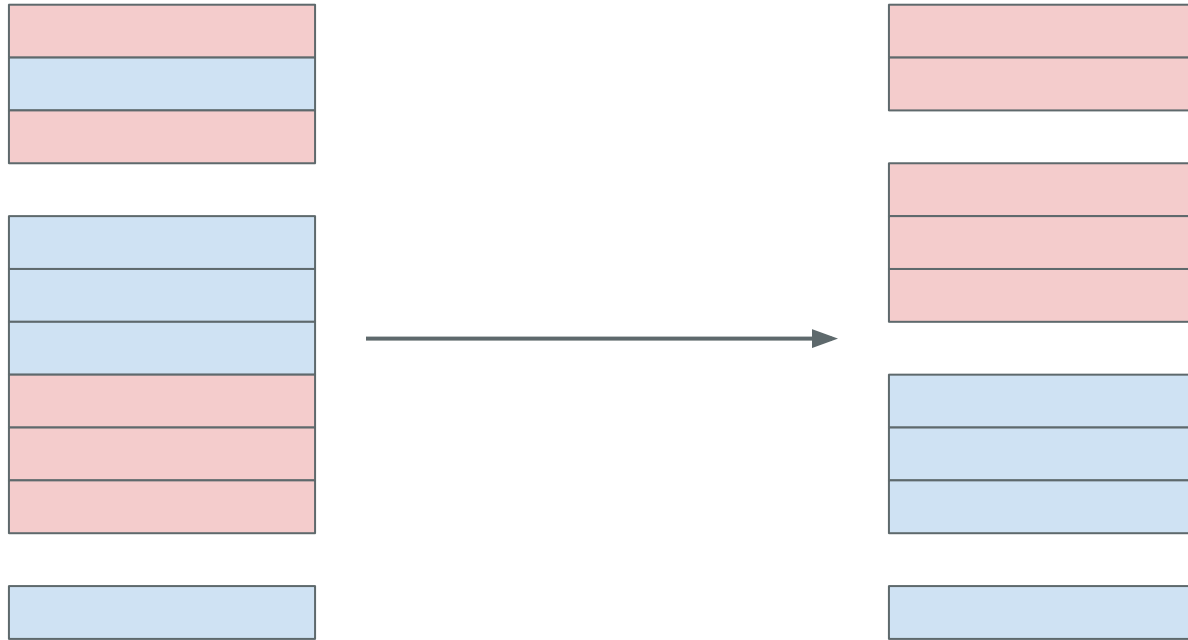




Marmalum

Alex Curtiss
Seth Hovestol
Izaak Weiss

Designating Frontend v Backend



```
frontend javascript
  "(id, str) => { document.getElementById(id).innerHTML = str }"
  as set(id: String, val: String): Void
```

```
frontend pageload(): Void {
  let x: Number = get_and_increment();
  set("counter", x);
}
```

```
backend counter: Number = 0;
```

```
backend get_and_increment(): Number {
  counter = counter + 1;
  print(counter);
  return counter;
}
```

How do I return the response from an asynchronous call?



I have a function `foo` which makes an Ajax request. How can I return the response from `foo`?

3593



I tried to return the value from the `success` callback as well as assigning the response to a local variable inside the function and return that one, but none of those ways actually return the response.

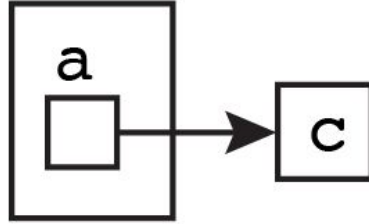


1770

```
function foo() {  
  var result;  
  
  $.ajax({  
    url: '...',  
    success: function(response) {  
      result = response;  
      // return response; // <- I tried that one as well  
    }  
  });  
  
  return result;  
}  
  
var result = foo(); // It always ends up being `undefined`.
```

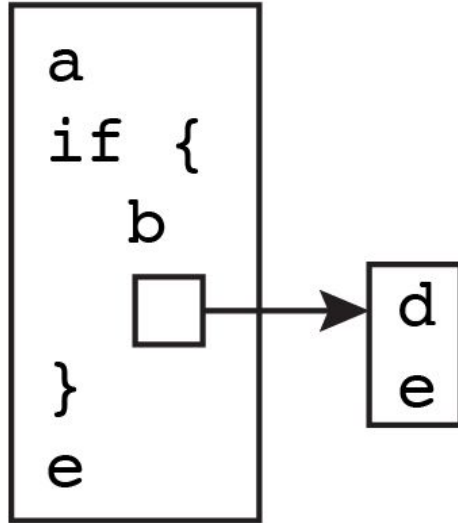
Statement Cutting

a
b ()
c



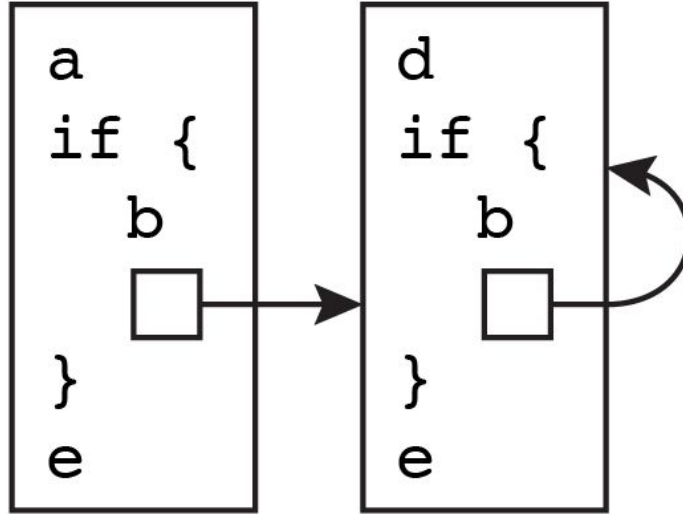
If Statement Cutting

```
a
if {
  b
  c()
  d
}
e
```



While Loop Cutting

```
a
while {
  b
  c()
  d
}
e
```



Generation

```
frontend get_id(user): Number{  
    return get_user_data(user).id;  
}
```



```
function get_id(user) {  
    function get_id_0(result) {  
        return result.id;  
    };  
    return backend("get_user_data", get_id_0)(user);  
}
```


A large red square with a white border, centered on a white background. The word "Demo" is written in white text in the center of the square.

Demo

```
frontend javascript
  "(id, str) => { document.getElementById(id).innerHTML = str }"
  as set(id: String, val: String): Void
```

```
frontend pageload(): Void {
  let x: Number = get_and_increment();
  set("counter", x);
}
```

```
backend counter: Number = 0;
```

```
backend get_and_increment(): Number {
  counter = counter + 1;
  print(counter);
  return counter;
}
```