

# # Scope: Who Dunit

## ### Learning Objectives

- Understand function scope
- Know the difference in between the let and const keywords

## ## Brief

Using your knowledge about scope and variable declarations in JavaScript, look at the following code snippets and predict what the output or error will be and why.

## ### MVP

### #### Episode 1

```
``js
const scenario = {
  murderer: 'Miss Scarlet',
  room: 'Library',
  weapon: 'Rope'
};

const declareMurderer = function() {
  return `The murderer is ${scenario.murderer}.`;
}

const verdict = declareMurderer();
console.log(verdict);
``
```

const scenario declared in block scope  
Remains constant  
Verdict = Miss Scarlet

### #### Episode 2

```
``js
const murderer = 'Professor Plum';

const changeMurderer = function() {
  murderer = 'Mrs. Peacock';
}

const declareMurderer = function() {
  return `The murderer is ${murderer}.`;
}

changeMurderer();
const verdict = declareMurderer();
console.log(verdict);
``
```

const murderer and const changeMurderer  
are two different declarations that clash?

### #### Episode 3

```
``js
let murderer = 'Professor Plum';

const declareMurderer = function() {
  let murderer = 'Mrs. Peacock';
  return `The murderer is ${murderer}.`;
}

const firstVerdict = declareMurderer();
console.log('First Verdict: ', firstVerdict);

const secondVerdict = `The murderer is ${murderer}.`;
```

```
console.log('Second Verdict: ', secondVerdict);
```

'Mrs. Peacock'; is declared in block scope and will be picked up as first verdict via "const declareMurderer function"

'Professor Plum' is 'let' declared globally and will be picked up as second verdict via "\${murderer}"

#### #### Episode 4

```
``js
let suspectOne = 'Miss Scarlet';
let suspectTwo = 'Professor Plum';
let suspectThree = 'Mrs. Peacock';

const declareAllSuspects = function() {
  let suspectThree = 'Colonel Mustard';
  return `The suspects are ${suspectOne}, ${suspectTwo}, ${suspectThree}.`;
}

const suspects = declareAllSuspects();
console.log(suspects);
console.log(`Suspect three is ${suspectThree}.`);
```

let' suspects are all declared globally

let suspect three is declared in const 'declareAllSuspects' scope block

suspects are then called via 'declareAllSuspects' which would call suspect three as 'Colonel Mustard'

suspect called outside the scope block would be 'Miss Scarlet'

#### #### Episode 5

```
``js
const scenario = {
  murderer: 'Miss Scarlet',
  room: 'Kitchen',
  weapon: 'Candle Stick'
};

const changeWeapon = function(newWeapon) {
  scenario.weapon = newWeapon;
}

const declareWeapon = function() {
  return `The weapon is the ${scenario.weapon}.`;
}

changeWeapon('Revolver');
const verdict = declareWeapon();
console.log(verdict);
```

somewhat tired, I pondered this one  
and then ran it in atom to follow the logic

I now see how it works

but I was starting to make mistakes with this so I stopped for the evening.

#### #### Episode 6

```
``js
```

```

let murderer = 'Colonel Mustard';

const changeMurderer = function() {
  murderer = 'Mr. Green';

  const plotTwist = function() {
    murderer = 'Mrs. White';
  }

  plotTwist();
}

const declareMurderer = function () {
  return `The murderer is ${murderer}.`;
}

changeMurderer();
const verdict = declareMurderer();
console.log(verdict);
```

```

#### #### Episode 7

```

```js
let murderer = 'Professor Plum';

const changeMurderer = function() {
  murderer = 'Mr. Green';

  const plotTwist = function() {
    let murderer = 'Colonel Mustard';

    const unexpectedOutcome = function() {
      murderer = 'Miss Scarlet';
    }

    unexpectedOutcome();
  }

  plotTwist();
}

const declareMurderer = function() {
  return `The murderer is ${murderer}.`;
}

changeMurderer();
const verdict = declareMurderer();
console.log(verdict);
```

```

#### #### Episode 8

```

```js
const scenario = {
  murderer: 'Mrs. Peacock',
  room: 'Conservatory',
  weapon: 'Lead Pipe'
};

const changeScenario = function() {
  scenario.murderer = 'Mrs. Peacock';
  scenario.room = 'Dining Room';

  const plotTwist = function(room) {
    if (scenario.room === room) {
      scenario.murderer = 'Colonel Mustard';
    }
  }

  const unexpectedOutcome = function(murderer) {
    if (scenario.murderer === murderer) {
      scenario.weapon = 'Candle Stick';
    }
  }
}
```

```

```
    }  
  }  
  unexpectedOutcome('Colonel Mustard');  
}  
  
plotTwist('Dining Room');  
}  
  
const declareWeapon = function() {  
  return `The weapon is ${scenario.weapon}.`  
}  
  
changeScenario();  
const verdict = declareWeapon();  
console.log(verdict);  
``
```

#### #### Episode 9

```
``js  
let murderer = 'Professor Plum';  
  
if (murderer === 'Professor Plum') {  
  let murderer = 'Mrs. Peacock';  
}  
  
const declareMurderer = function() {  
  return `The murderer is ${murderer}.`;   
}  
  
const verdict = declareMurderer();  
console.log(verdict);  
``
```

#### ### Extensions

Make up your own episode!