

Declarations with *const*

Level 1 – Section 3

Magic Numbers

A magic number is a literal value without a clear meaning.

```
function loadProfiles(userNames){  
  if(userNames.length > 3){  
    //...  
  }else{  
    //...  
  }  
}
```



The number 3 by itself
doesn't tell us much

Issues With Magic Numbers

When used multiple times, magic numbers introduce **unnecessary duplication**, which can lead to **bad code!**

```
function loadProfiles(userNames){  
  if(userNames.length > 3){  
    //...  
  }else{  
    //...  
  }  
  
  if(someValue > 3){  
    //...  
  }  
}
```

Hard to tell whether both numbers
serve the same purpose



Replacing Magic Numbers With Constants

ES2015

The *const* keyword creates **read-only** named constants.

```
function loadProfiles(userNames){
```

```
    const MAX_USERS = 3;
```

```
    if(userNames.length > MAX_USERS){
```

```
        //...
```

```
    }else{
```

```
        //...
```

```
    }
```

```
    const MAX_REPLIES = 3;
```

```
    if(someElement > MAX_REPLIES){
```

```
        //...
```

```
    }
```


```
}
```




Conveys intent and facilitates maintenance

Constants Are Read-only

Once assigned, constants **cannot** be assigned a new value.

```
function loadProfiles(userNames){  
  
  const MAX_USERS = 3;  
  MAX_USERS = 10;  Value does not change  
  
  if(userNames.length > MAX_USERS){  
    //...  
  }else{  
    //...  
  }  
  
  const MAX_REPLIES = 3;  
  if(someElement > MAX_REPLIES){  
    //...  
  }  
}
```

 Still holds the value 3

Constants Require an Initial Value

Variables declared with *const* must be assigned an initial value.

```
function loadProfiles(userNames){
```

```
  const MAX_USERS;  
  MAX_USERS = 10;
```

> SyntaxError: Unexpected token



```
    if(userNames.length > MAX_USERS){  
      //...  
    }else{  
      //...  
    }  
  }
```

```
  const MAX_REPLIES = 3;  
  if(someElement > MAX_REPLIES){  
  
  }  
}
```

Constants Are Block Scoped

Variables declared with *const* are scoped to the nearest block.

```
function loadProfiles(userNames){
```

```
  const MAX_USERS = 3;
```

```
  if(userNames.length > MAX_USERS){
```

```
    const MAX_REPLIES = 15;
```

```
    //...
```

```
  }else{
```

```
    //...
```

```
  }
```

```
  console.log(MAX_REPLIES);
```

```
}
```

← Not visible outside the if block

→ ReferenceError, MAX_REPLIES is not defined.



let vs. const

In most cases, *let* and *const* will behave very similarly. Consider the semantics when choosing one over the other.

```
let loadingMessage = "This might take a while...";  
let currentAge = 50;  
let totalCost = cost + tax;
```

Use *let* when variables could be reassigned new values

```
const MAX_USERS = 3;  
const SEPARATOR = "%%";  
const ALLOW_EDIT = false;
```

Use *const* when new variables are not expected to be reassigned new values