

EXAMPLE 23: BAD THINGS HAPPEN IF YOU
CALL A CONSTRUCTOR WITHOUT USING **new**

AN OBJECT CONSTRUCTOR

**IS A FUNCTION THAT CREATES OBJECTS, ALL OF
WHICH HAVE THE SAME PROPERTIES**

**THIS FUNCTION 'LOOKS' LIKE ANY OTHER
FUNCTION, BUT IT IS SPECIAL IN A FEW WAYS**

RECAP

AN OBJECT CONSTRUCTOR

**'LOOKS' LIKE ANY OTHER FUNCTION, BUT IT IS
SPECIAL IN A FEW WAYS**

**IT MAGICALLY 'KNOWS' WHAT OBJECT IT IS CREATING,
AND ACCESS IT VIA A VARIABLE CALLED `this`**

**IT MUST BE CALLED IN A SPECIFIC WAY, USING THE
KEYWORD `new`**

RECAP

USING AN OBJECT CONSTRUCTOR

```
var rectangle = new Rectangle(3.3,  
2.5, "Blue");
```

AN OBJECT CONSTRUCTOR MUST BE CALLED IN A
SPECIFIC WAY, USING THE KEYWORD **new**

THIS IS THE ONLY DIFFERENCE BETWEEN CALLING A
CONSTRUCTOR, AND CALLING ANY OTHER FUNCTION

RECAP

```
var rectangle = new Rectangle(3.3,  
2.5, "Blue");
```



THAT **new** KEYWORD IS A SIGN TO THE JAVASCRIPT
INTERPRETER..

TO CREATE AN EMPTY OBJECT, AND PASS IT INTO
THE FUNCTION (SECRETLY) AS THE **this**..

..IN ADDITION TO THE OTHER FUNCTION
ARGUMENTS

RECAP

```
var rectangle = new Rectangle(3.3,  
2.5,
```



THAT **new** **KEYWORD** IS A SIGN TO THE JAVASCRIPT
INTERPRETER..

TO CREATE AN EMPTY OBJECT, AND PASS IT INTO
THE FUNCTION (SECRETLY) AS THE

..IN ADDITION TO THE OTHER FUNCTION
ARGUMENTS

RECAP

```
var rectangle = new Rectangle(3.3,  
2.5,
```



THAT

```
var this = {};
```

INTERPRETER..

TO CREATE AN EMPTY OBJECT, AND PASS IT INTO
THE FUNCTION (SECRETLY) AS THE `this`..

..IN ADDITION TO THE OTHER FUNCTION
ARGUMENTS

RECAP

```
var rectangle = new Rectangle(3.3,  
2.5,
```



THAT

```
var this = {};
```

TO CREATE AN EMPTY OBJECT, AND **PASS IT INTO**
THE FUNCTION (SECRETLY) AS THE `this`..

```
function Rectangle(length, breadth, color)  
  this.length = length;  
  this.breadth = breadth;  
  this.color = color;  
}
```

RECAP


```
function Rectangle(this, length, breadth, color) {  
  var this = {};  
  this.length = length;  
  this.breadth = breadth;  
  this.color = color;  
  return this;  
}
```

```
var rectangle = Rectangle(this,  
3.3, 2.5, "Blue");
```

TO CREATE AN EMPTY OBJECT, AND PASS IT INTO
THE FUNCTION (SECRETLY) AS THE **this**..

..IN ADDITION TO THE OTHER FUNCTION
ARGUMENTS

RECAP

```
var rectangle = new Rectangle(3.3,  
2.5, "Blue");
```

IS LOGICALLY EQUIVALENT TO

```
function Rectangle(this, length, breadth, color)  
{  
  var this = {};
```

```
    this.length = length;  
    this.breadth = breadth;  
    this.color = color;  
    return this;  
}
```

```
var rectangle = Rectangle(  
3.3, 2.5, "Blue");
```

RECAP

WRITING AN OBJECT CONSTRUCTOR

REMEMBER THE MAGICAL ROLE OF `this`

USING AN OBJECT CONSTRUCTOR

REMEMBER THE MAGICAL ROLE OF `new`

RECAP

EXAMPLE 23: BAD THINGS HAPPEN IF YOU
CALL A CONSTRUCTOR WITHOUT USING **new**

**EXAMPLE 23: BAD THINGS HAPPEN IF YOU
CALL A CONSTRUCTOR WITHOUT USING `new`**

```
var rectangle = new Rectangle(3.3,  
2.5, "Blue");
```

THIS IS THE CORRECT WAY TO USE AN OBJECT

WHAT WOULD HAPPEN IF WE FORGOT THE `new`?

EXAMPLE 23: BAD THINGS HAPPEN IF YOU CALL A CONSTRUCTOR WITHOUT USING **new**

```
var rectangle = new Rectangle(3.3,  
2.5, "Blue");
```



THIS IS THE CORRECT WAY TO USE AN OBJECT

WHAT WOULD HAPPEN IF WE FORGOT THE **new**?

**EXAMPLE 23: BAD THINGS HAPPEN IF YOU
CALL A CONSTRUCTOR WITHOUT USING `new`**

```
var rectangle = new Rectangle(3.5, 2.5, "Blue");  
var rectangle = Rectangle(3.5, 2.5, "Blue");
```

WRONG! WRONG! WRONG!

**THE JAVASCRIPT INTERPRETER WILL NOT
KNOW THAT AN OBJECT CONSTRUCTOR HAS BEEN CALLED.**

**EXAMPLE 23: BAD THINGS HAPPEN IF YOU
CALL A CONSTRUCTOR WITHOUT USING `new`**

```
var rectangle = Rectangle(3.3, 2.5, "Blue")
```

WRONG! WRONG! WRONG!

**THE JAVASCRIPT INTERPRETER WILL NOT
KNOW THAT AN OBJECT CONSTRUCTOR HAS BEEN CALLED.**

**THE `this` POINTER WILL NOT BE PASSED IN AT ALL,
AND WILL DEFAULT TO THE BROWSER `window` OBJECT**

**EXAMPLE 23: BAD THINGS HAPPEN IF YOU
CALL A CONSTRUCTOR WITHOUT USING `new`**

**THE `this` POINTER WILL NOT BE PASSED IN AT ALL,
AND WILL DEFAULT TO THE BROWSER `window` OBJECT**

```
function Rectangle(length,breadth,color) {  
    this.length = length;  
    this.breadth = breadth;  
    this.color = color;  
}
```

EXAMPLE 23: BAD THINGS HAPPEN IF YOU CALL A CONSTRUCTOR WITHOUT USING **new**

THE **this** POINTER WILL NOT BE PASSED IN AT ALL, AND WILL DEFAULT TO THE BROWSER **window** OBJECT

```
function Rectangle(length, breadth, color) {  
    this.length = length;  
    this.breadth = breadth;  
    this.color = color;  
}
```

AND ANY ATTEMPT TO USE IT IN THE CONSTRUCTOR WILL THROW AN ERROR

EXAMPLE 23: CALL A CONSTRUCTOR WITHOUT USING THE AND WILL DEFAULT TO THE BROWSER

value of this = [object Window]

undefined

✖ ▶ Uncaught TypeError: Cannot read property 'length' of undefined

```
this.color = color;
```

AND ANY ATTEMPT TO USE IT IN THE CONSTRUCTOR WILL THROW AN ERROR

EXAMPLE 23: CALL A CONSTRUCTOR WITHOUT USING THE AND WILL DEFAULT TO THE BROWSER

```
value of this = [object Window]
```

```
undefined
```

✖ ▶ Uncaught TypeError: Cannot read property 'length' of undefined

```
this.color = color;
```

AND ANY ATTEMPT TO USE IT IN THE CONSTRUCTOR WILL THROW AN ERROR