

The variable `name` is now restricted to its block. `let` has been introduced to replace `var` in the long run, so you can pretty much drop the good old `var` keyword and start using `let` instead. The cool thing is, it should be painless to use `let`, and if you can't, you have probably spotted something wrong with your code!

3.3. Constants

Since we are on the topic of new keywords and variables, there is another one that can be of interest. ES6 introduces `const` to declare... constants! When you declare a variable with `const`, it has to be initialized and you can't assign another value later.

```
const poniesInRace = 6;
```

```
poniesInRace = 7; // SyntaxError
```

As for variables declared with `let`, constants are not hoisted and are only declared at the block level.

One small thing might surprise you: you can initialize a constant with an object and later modify the object content.

```
const PONY = {};  
PONY.color = 'blue'; // works
```

But you can't assign another object:

```
const PONY = {};
```

```
PONY = {color: 'blue'}; // SyntaxError
```

Same thing with arrays:

```
const PONIES = [];  
PONIES.push({ color: 'blue' }); // works
```

```
PONIES = []; // SyntaxError
```

3.4. Creating objects

Not a new keyword, but it can also catch your attention when reading ES6 code. There is now a