```
// LECTURE 13: let, const and var

/*
We use the keyword let
- to be able to change its value over the course of our
  program1
- Variables are mutable.
*/
let age = 30;
age = 31;
// Mutability also applies to empty variables
// then assign its value later.
let year;
year = 1991;
/*
We use the keyword const
- to create a variable which cannot be mutated.
*/
const birthYear = 1991;
birthYear = 1990; // <-- Error code!!

const job; // <-- This is not legal!

// Always declare a variable as const by default,
// unless you are really certain that its value will
// have to change some point in the future.
// Its a good practice to minimise the variable mutations
// you have in your code.

/*
We use the keyword var
- to create a mutable variable.
- this is the old way of declaring variables.
We'll learn more about this in Section 7 on variable
  scope, but for now
don't use the keyword var.
*/
var job1 = "programmer";
job1 = "teacher"; // this will work!

var job2;
job2 = programmer; // this will work too!
```

```
41
42  /*
43  In fact, you can even declare variables like this:
44  */
45  myName = 'main';
46  console.log(myName); // ==> "main"
47  // However, this is actually a pretty terrible idea
    •   because
48  // this doesn't create a variable in the current scope.
    •   Instead,
49  // Js will create a variable on the GLOBAL SCOPE.
50
```