Lecture 1

Variables

Source: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/Variables

Let’s start from the building blocks of any language - variables. In JS initially all variables were declared like this:

var count;

In modern JS we use let for variables:

let count;

Rule of thumb - do not use var. Use let.

Variables hold some value.

The value can be:

- String

let myName = ‘Andrei’;

- Number

let myAge = 30;

- Boolean

let isOpen = true;

- Array

let colors = [‘red’, ‘white’];

- Object

let person = { name: ‘Andrei’, age: 18, title: ‘faculty’ };

```
}
```

If variable is supposed to hold a value which will not change, we use const:

```
const pi = 3.14;
```

Note, that constant must have a value once it is declared. This will through an error:

```
const pi;
```

If we declare a variable but do not assign any value to it, it will have value undefined.

Example

For example, if I want to have a counter and store its value, I will create: const count = 1;

Text can be in single quotes (‘), double quotes (“) or back quotes (`).

Events

Source: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Building_blocks/Events

Events happen on the page because of user interactions. We call it ‘fires’. For example, if user clicks on a button, ‘click’ event fires.

Other possible user events: keyup, keydown, hover, select.

There are also events related to global window object. For example, load. That’s why we have:

```
window.onload = () => {}
```

One more example of an event - is submission of the form:

```
form.onsubmit = () => {}
```

Events matter because we can listen to them and perform some specific actions in response to these events. This is called event handler. Event handler is a function which we can create and attach to the event like this:

```
function buttonClickHandler() {
        // some logic inside
}
```

```
button.addEventListener(‘click’, buttonClickhandler);
```

Classes

Before learning classes, let’s recap what is Object.

Source: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/Basics

Objects are collections of related values. For example, a person has a name, age, location where he lives. We can create variables for these:

```
Let name = ‘Andrei’;
Let age = 18;
Let location = ‘Toronto’;
```

Let’s assume we need to create the same values for other person:

```
Let name = ‘Michael’;
Let age = 4;
Let location = ‘Quebec’;
```

Now the problem is that we have lost all data for Andrei - we just reassigned new values to existing variables.

Another problem is that the same data may be used for other things from the real world - like cars, for example. A car can have a name, age and location. How do we differentiate between person data and car data?

Location

Source: https://developer.mozilla.org/en-US/docs/Web/API/Location

There is a location object available for us in JS.

Let’s examine of which elements a url consists. We can access all of them in location object available for us.

Forms

We can access form value programmatically

First we locate form on the page:

```
Const myForm = document.forms.form_id
```

Next we can access form data from form fields by field name:

```
<input name=”age”>
```

myForm.age.value => will have value from input

Adapted from Sean Doyle class.