JavaScript Time & Canvas

CISC-2350-R01 | Fall 2017 | Week 14-2

Ruta Kruliauskaite

Today’s Agenda

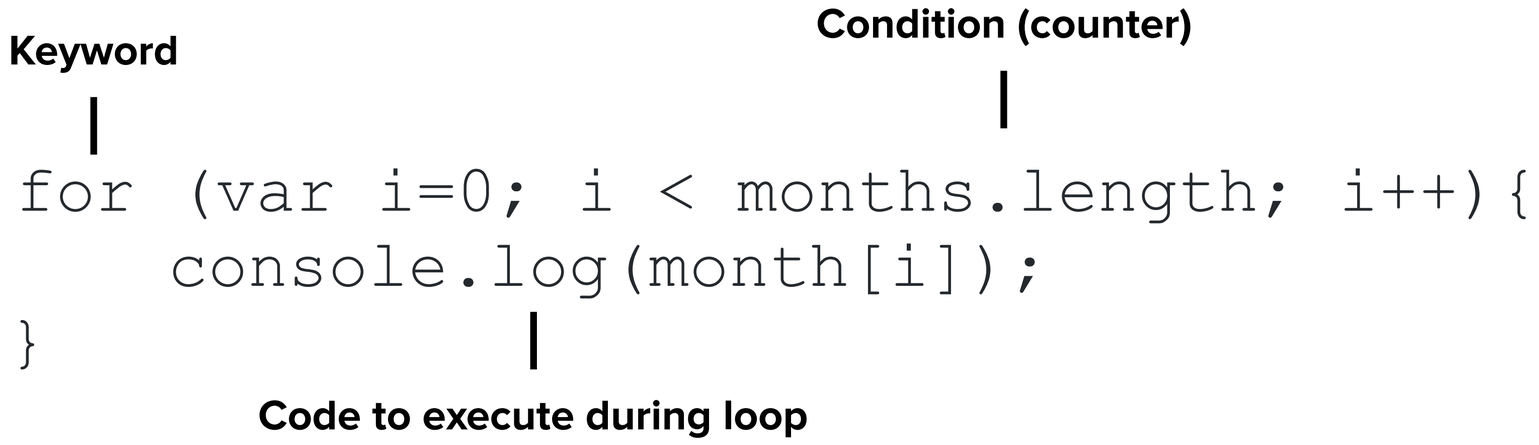
- Attendance
- JavaScript logic
- Time
- Canvas
- Homework Assignment

Loops

For loop

For loop uses a counter as a condition.
It instructs code to run a specified number of times.
Good to use when the number of repetitions is known, or can be supplied by the user.

```
var months = [“January”, “February”, “March”, “April”, “May”,
“June”, “July”, “August”, “September”, “October”, “November”,
“December”];
```



Example: for loop

While loop

Good to use in applications with numeric situations and when we don’t know how many times the code should run.
In other words: the loop repeats until a certain “condition” is met.
If the condition is false at the beginning of the loop, the loop is never executed.

```
var i=1;

while (i < number){
      console.log(i);
      i++;
}
```

Example: while loop

Do while loop

Same concept as the while loop.
Except that this loop will always execute the loop at least once (even if the condition is false).
Good to use when you are asking a question, whose answer will determine if the loop is repeated.

```
var i=1;

do {
  console.log(i);
  i++;
} while (i < number);
```

Example: do...while loop

More examples

Example: loop through data in external file

Example: event listener

Example: name randomizer

Time

When do we need JavaScript Time

- When we want to know what today (or any other day) is
- When we want to execute something at a certain time
- When we want to execute something repeatedly

For this we can use JavaScript Date object that has a lot of different built in methods and parameters.

All date functionality references January 1st, 1970, known as Unix time.

Methods you can use

- `Date();` - get current date
- `Date.now();` - the number of milliseconds since January 1, 1970
- `var today = new Date(Date());` - creates a new date object in a variable
- `today.getDate();` - returns the day of the month (1-31)
- `today.getDay();` - returns the day of the week (0-6, where Sunday - 0)
- `today.getFullYear();` - returns the year
- `today.getHours();` - returns the hour (0-23)
- `today.getMilliseconds();` - returns the milliseconds (0-999)
- `today.getMonth();` - returns the month (0-10)
- `today.getSeconds();` - returns the seconds (0-59)
- `today.getTime();` - returns the number of milliseconds since midnight January 1, 1970, and a specified date

Full Date and Time documentation

- https://www.w3schools.com/jsref/jsref_obj_date.asp
- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date

How to count elapsed time

```
// Get current time in milliseconds
var start = Date.now();

// Something happens for a long time…

// Get new current time in milliseconds
var end = Date.now();
var elapsed = end - start; // elapsed time in
milliseconds
```

setTimeout, setInterval

- We use setTimeout to execute function at a certain time (e.g. in 2 seconds)
- We use setInterval when we want something to happen repeatedly

```
function alertMe() {
  alert(“It’s me!”);
}

setTimeout(alertMe, 2000);
```

```
function alertMe() {
  alert(“It’s me!”);
}

setInterval(alertMe, 2000);
```

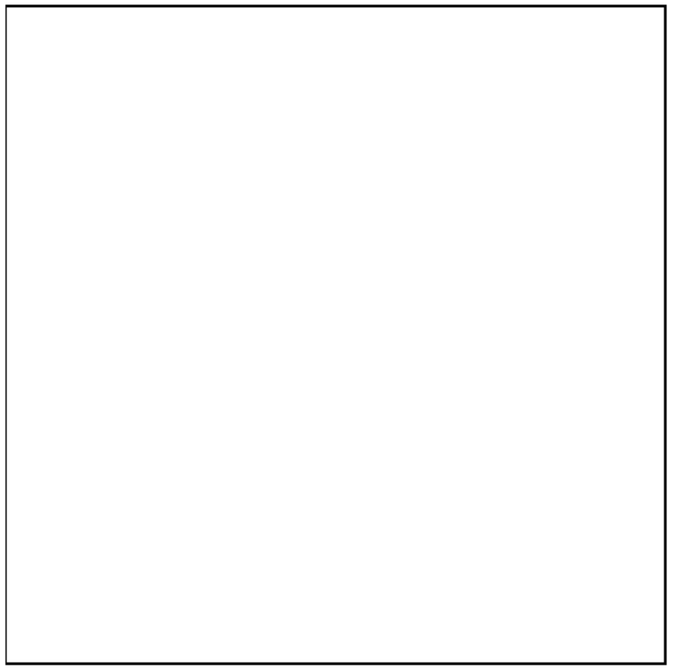
Clearing timeout or interval

- We assign setTimeout, setInterval to a variable
- And use clearTimeout or clearInterval

```
function alertMe() {
   alert(“It’s me!”);
}

var alertIs = setInterval(alertMe, 2000);

function stopAlert () {
   clearInterval(alertIs);
}

setTimeout(stopAlert, 10000); // will stop after 5 alerting 5
times
```

The Canvas

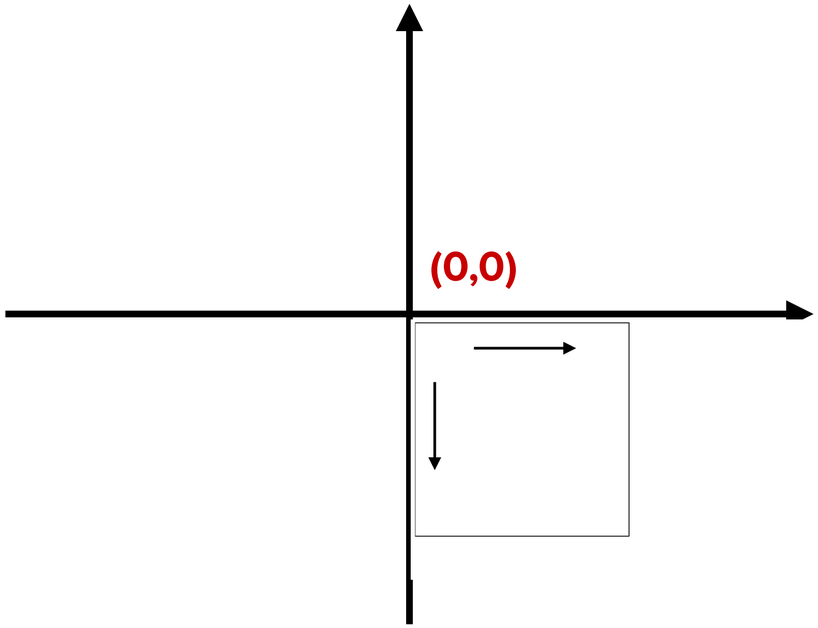
<canvas> is used to draw graphics on the webpage



```
<canvas id="myCanvas" width="300px" height="300px" style="border: 1px solid black"></canvas>
```

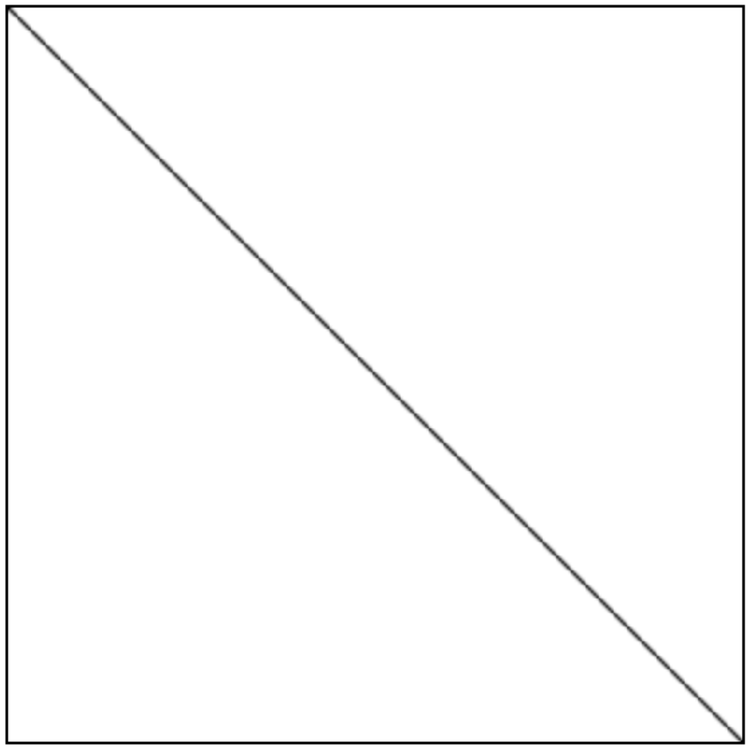
To make Canvas work

- Always define width and height
- And give id (because you'll have to call it in JavaScript)
- Canvas coordinates start from 0



Drawing a line

```
1. var c =
   document.getElementById(“myCanvas”);
2. var ctx = c.getContext(“2d");
3. ctx.moveTo(0,0);
4. ctx.lineTo(300,300);
5. ctx.stroke();
```



Built-in Canvas Methods

- rect() - creates a rectangle
- stroke() - draws a path we have defined
- beginPath() - begins a path
- arc() - creates an arc / curve (used to create circles)
- scale() - scale a current drawing bigger or smaller
- drawImage() - draws an image, canvas or video onto canvas
- And many other

Canvas Tutorial

- https://www.w3schools.com/tags/ref_canvas.asp
- https://code.tutsplus.com/articles/21-ridiculously-impressive-html5-canvas-experiments--net-14210
- http://natureofcode.com/ - more creative approach to code

Example: fill color

Example: draw loop

Example: mousemove

Example: pixel manipulation

Final Project Assignment

Final Project Assignment Timeline

Monday, November 20th:
- Concepts ready to present
- In-class 1 min presentations (1 slide ready)
- Office hours after class available

Thursday, November 23rd:
- NO CLASS, Thanksgiving

Monday, November 27th:
- Content ready
- Wireframes ready
- Visual Design ideas ready: options for use of color, iconography and fonts
- In-class workshop for feedback (3min presentations ready)
- Office hours after class available

Thursday, November 30th:
- First website drafts ready (HTML structure & CSS)
- Office hours after class available if necessary

Monday, December 4th:
- More functional websites ready (improving CSS, adding JavaScript functionality)
- Office hours after class available

Thursday, December 7th:
- Final in-class presentations
- 5min presentations ready
- Double class w/ guest critics

Final Project Grading

Student Name: _____ Section: _____

CISC-2350-R01-2017 – Spring 2017 – Midterm Project Rubric – Instructor: Ruta Kruliauskaite

	Exceeded Criteria (100)	Met Criteria (90)	Approached Criteria (80)	Did Not Meet Criteria (60)	Points (weight)
Site Concept and Originality	Something novel or a great mashup	Interesting idea or a classic take on standards	The idea is not new but you have added many nice embellishments	Little thought or effort put into the site concept	10
Visual Design	Individual and overall design are exceptional	The overall visual design is cohesive and interesting	Some of the individual elements are interesting but overall design not cohesive	Little thought went into the visual design	20
Site Execution	Meets all of the web requirements plus includes features beyond the basics. Exceeds requirements.	Meets all of the web requirements	Most of the web requirements have been met.	Less than 60% of the web requirements have been met.	50
In Class Presentation	Able to explain in detail some aspect of your site that is novel or interesting. You must be able to break down and explain how that piece of code works.	Good presentation of what the site is, how to interact with it and what your greatest programming challenge was. Why you chose to organize your code the way you did.	Can explain why you chose your project. Can walk someone through some of the code logic.	Not able to explain site and how it works beyond very basic level.	10
Submitted on Time. Includes formatting and comments	All files submitted by deadline. Excellent comments, and formatting	All files submitted by deadline. Well formatted and includes comments	Submitted on time with some comments	Did not submit on time. Little or no comments.	10

https://github.com/rutaitp/CISC-2350-R01-2017/blob/master/week6/grading_rubric.pdf

Homework assignment

Homework

By Sunday, December 3, 6pm

1. **Web review presentations: Danielle, Julian**
2. **Review the class slides**
3. **Keep working on your final project assignment:**
    1. You should have more functional websites ready
        - Final HTML structure
        - More improved CSS
        - Added JavaScript functionality
4. Office hours will be available after class for final questions before presentations on Thursday, December 7