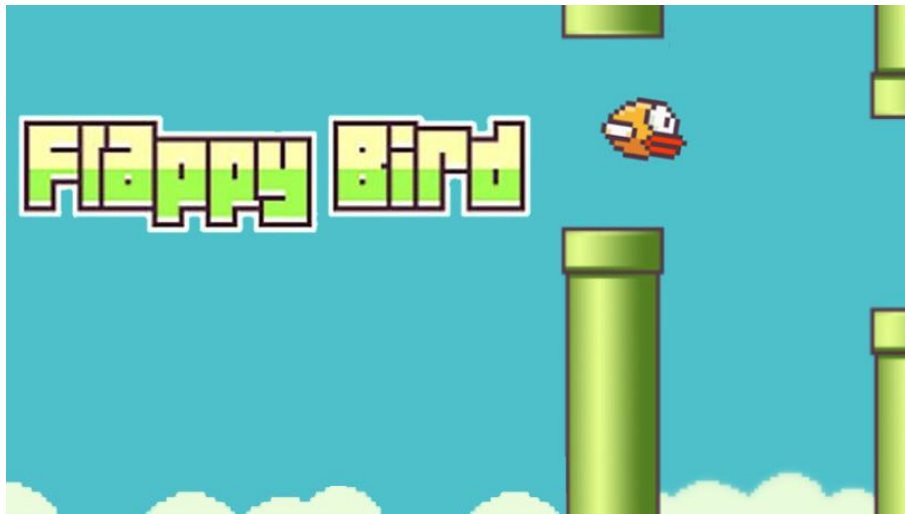

Web Animation

Week1

What Will We do in this Module?

We will learn the basics of animation

We will create a simple HTML canvas game based on Flappy Bird



Plan of Action

- WEEK1
 - Intro to animation and canvas: Setup our constructors and other game functions
- WEEK2
 - Set up our Bird character and gravity
- WEEK3
 - Obstacles and Collision
- WEEK4
 - Score and End Screen
- WEEK5
 - Wrap up

Animations



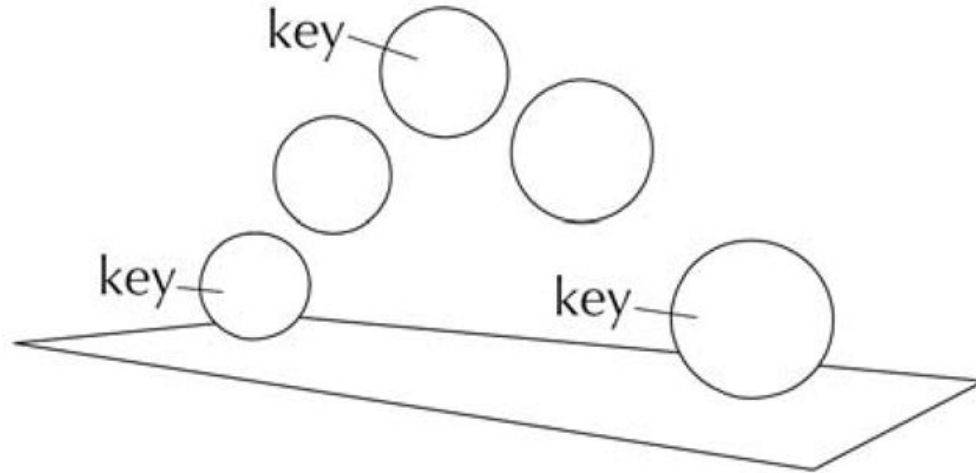
What is an Animation?

- A series of static images with slight modifications; when played together it gives the impression of movement.
- Like a flipbook!



Frames and KeyFrames

- Each of these static images is called a Frame
- Key Frames are moments of change at a certain time in the animation
- Key Frames are usually denoted by diamond symbol in animation software ◀ ◀◀ ▶▶ ▶▶▶



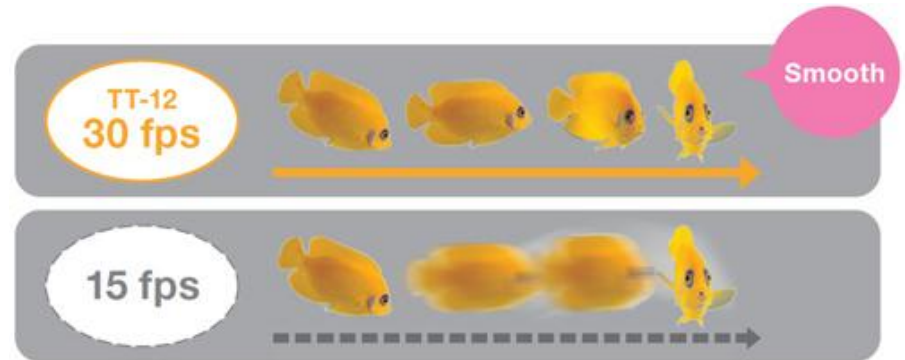
Frames Per Second FPS

FPS refers to how many static images there are in 1 second to give the appearance of movement.

Usually about 24/25 FPS for a standard animation

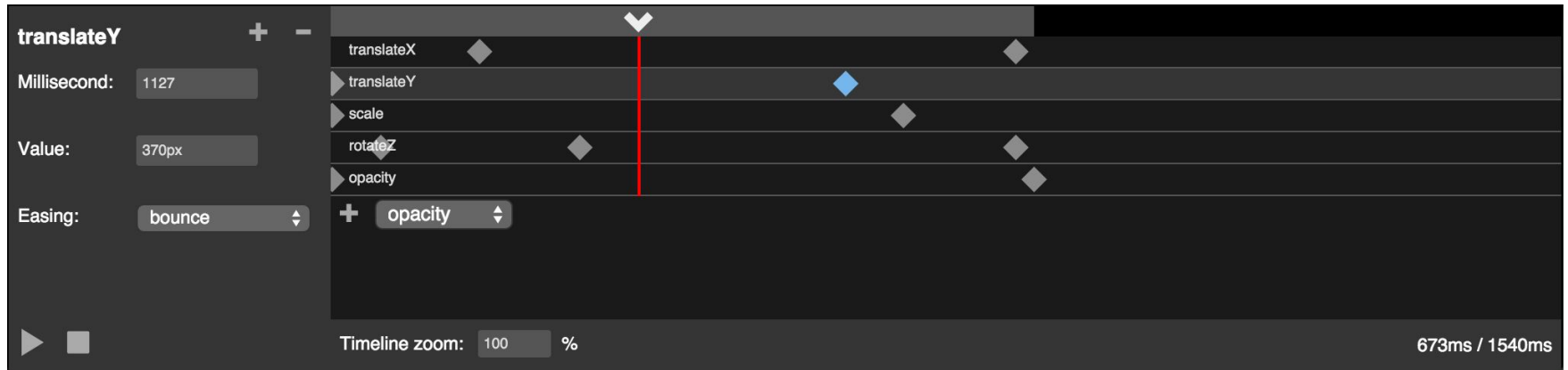
More Frames = Smoother Animation.

Less Frames = Judder/jitter



TimeLine Example

- Reference to time
- Key Frames



Compound/Complex Animations

We are not restricted on what can be animated at any one time

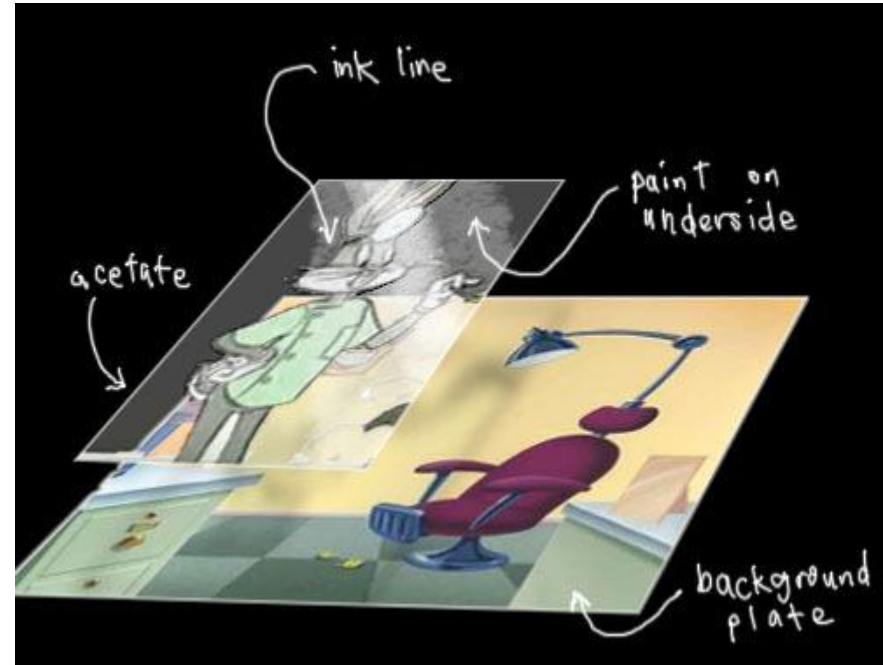
We can animate multiple attributes at once, and multiple objects at once.



Layers

To separate the parts of an animation we use layers

- In animation and graphics software, a "layer" refers to differing levels on which you can place your drawings and objects, stacked above and below each other.
- Top layers will obscure bottom layers, just as when stacking things in real life.



Recap

- Time
- Frames + KeyFrames (changes - scale, position, rotation)
- FPS
- Can have multiple elements animating
- We separated using layers

Where else can animations be used?

Games!



HTML CANVAS



<Canvas>

- Is a HTML Element just like div, section, article etc.
- The tags come in pairs <canvas></canvas>
- We can setup some attributes such as id, height and width
- Canvas lets us draw or create images on the fly
- **NOTE** the canvas is just a container
- Javascript is necessary for drawing and animations

```
<canvas id="myCanvas" width="200" height="100"></canvas>
```

<Canvas> Context

Canvas can have two contexts: 2D and 3D

Depending on the context, there will be different drawing properties

We will use the 2D context in our code

Re-Cap on JavaScript

What is the syntax?
What can we do with it?



Object Oriented Programming

- To use the canvas for drawing or animating we need to understand Object Oriented Programming (OOP)
 - Objects
 - Classes
 - Constructors

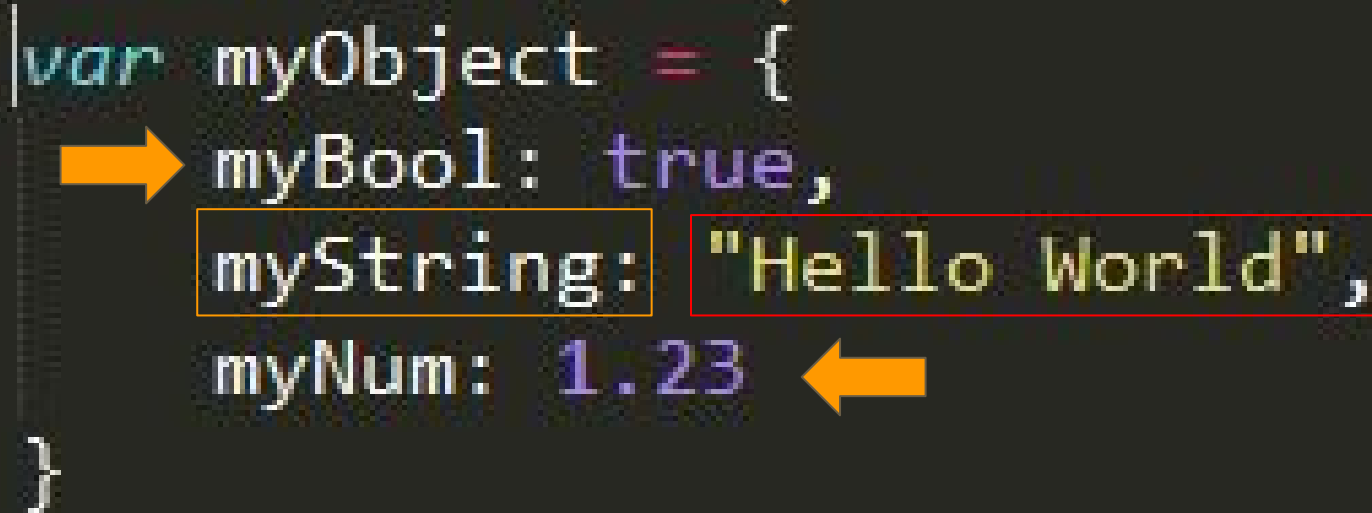
What is an Object? and Why would we use it?



JavaScript Object

- We can create our own objects in JavaScript.
- Observe Syntax
- Think of the body of the object as name: value pairs

```
|var myObject = {  
    myBool: true,  
    myString: "Hello World",  
    myNum: 1.23  
}
```




```
var myObject = {  
  myBool: true,  
  myString: "Hello World",  
  myNum: 1.23  
}
```

The diagram illustrates the syntax of a variable declaration. An orange arrow points down to the opening curly brace '{'. Another orange arrow points right to the first property 'myBool: true'. A yellow box highlights the property 'myString:', and a red box highlights the value '"Hello World"'. A third orange arrow points left to the value '1.23'.

Similar declaration for variable except now we enclose values in { }
Name Value pairs follow and are separated by a comma ,
name: value,

```
var person = {  
    firstName: "John",  
    lastName: "Doe",  
    age: 50,  
    eyeColor: "blue",  
    sayHello: function(){  
        alert("Hello");  
    }  
};
```



We can also add functions to our JavaScript objects. These are called methods

How do we access the object values/methods?

Syntax:

```
objectName["propertyName"]  
objectName.propertyName  
objectName.methodName()
```

Our person
object example:

```
person.firstName  
person["firstName"]  
person.sayHello()
```

Game Area Object

- We will create a game area object that will:
 - Create a canvas
 - Setup width height of canvas
 - Setup context
 - Add to HTML document
 - Set Frame number
 - Set interval for redraw
 - Have a clear method
 - Clear everything in canvas

```
var myGameArea = {  
  // new JS tag -> we are creating an element = canvas  
  canvas : document.createElement("canvas"),  
  start : function() {  
    //setting up width and height of canvas  
    this.canvas.width = 480;  
    this.canvas.height = 270;  
  },  
  // this is so that we can redraw every frame  
  clear : function() {  
  }  
}
```

```
var myGameArea = {  
  // new JS tag -> we are creating an element = canvas  
  canvas : document.createElement("canvas"),  
  start : function() {  
    //setting up width and height of canvas  
    this.canvas.width = 480;  
    this.canvas.height = 270;  
  },  
  // this is so that we can redraw every frame  
  clear : function() {  
  }  
}
```

What's this?

this refers to the object

this in the above code is referring to myGameArea

What if I want to Create Multiple Objects?

- To create multiple objects we will need to use a constructor.
- Once you have an object constructor, you can create new objects of the same type.
- Think of a constructor like a blueprint or a recipe for the objects you want to make.

```
function person(first, last, age, eye) {  
    this.firstName = first;  
    this.lastName = last;  
    this.age = age;  
    this.eyeColor = eye;  
}  
var myFather = new person("John", "Doe", 50, "blue");  
var myMother = new person("Sally", "Rally", 48, "green");
```

- The above function (person) is an object constructor.
- myFather and myMother are person objects created by the person constructor
- Make note of the syntax and the word new



Why would we need a constructor to create objects?

Component Constructor


- We will create a constructor which will create:
 - The Bird
 - The obstacles
 - The background
 - Update method to redraw for every frame

Time, Frames, Interval

- To give the appearance of movement the canvas needs to draw our background, obstacles and bird as one picture or frame
- Then clear the canvas so it is blank
- And draw everything again in its new positions
- To keep track of time we will set the initial Frame Number in our GameArea object and its interval using the following code

```
setInterval(functionToBeCalled, 20);
```

- We will then use another function to update these values



20 = means
every 20th
millsec so it
is called 50
times per
second

Handy Links

cs1.ucc.ie/~tv3/mpt