



Web Apps Development



Part II - Advanced Features
Device Orientation API - Part I

Homework Feedback

- Code Formatting. (<http://jsbeautifier.org>)



- Event Handlers, Handler Functions, & Back Holes (*e*, *e.target*, and *this* inside handler funct.)
 - Classes vs Instances (Do not call classes straight - or lose all your instances)

Homework Feedback

```
htmlElement.addEventListener("click", disappear);
```

```
function(e) {  
  this.style.display = "none"  
  e.target.style.display = "none"  
}
```

your black hole!

Increasing Flexibility

The problem:

So far, all of our objects are created within the width and height of the document. How do I make my object flexible enough to allow for other different dimensions?

Increasing Flexibility

Generic Functions

```
xPos: function() {  
    return this.generateRandomPositionForBody(document.documentElement.clientWidth);  
},  
yPos: function() {  
    return this.generateRandomPositionForBody(document.documentElement.clientHeight);  
},
```

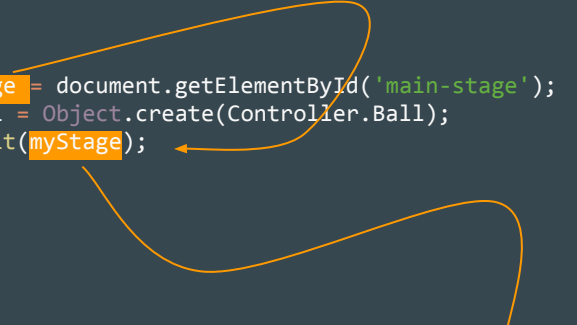
must go!



Increasing Flexibility

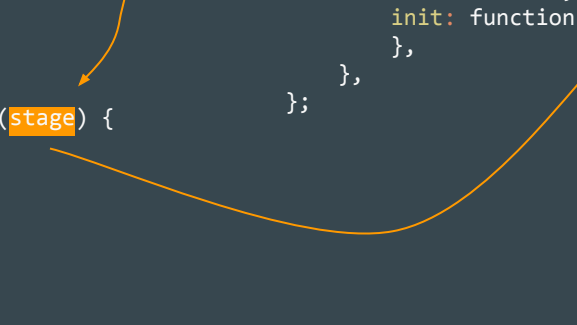
Generic Functions

```
(function() {  
  var myStage = document.getElementById('main-stage');  
  var myBall = Object.create(Controller.Ball);  
  myBall.init(myStage);  
})();
```



A yellow arrow originates from the `myStage` variable in the first code block and points to the `stage` parameter in the `init` function of the `myBall` object in the second code block.

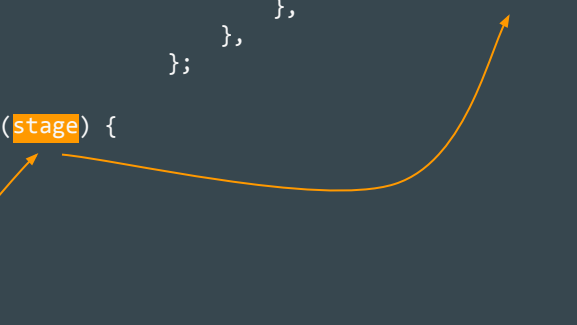
```
var Model = {  
  Ball: {  
    view: null,  
    init: function(stage) {  
    },  
  },  
};
```



A yellow arrow originates from the `stage` parameter in the `init` function of the `Ball` object in the second code block and points to the `stage` parameter in the `init` function of the `Controller.Ball` object in the third code block.

```
var View = {  
  Ball: {  
    element: null,  
    init: function(stage) {  
    },  
  },  
};
```

```
var Controller = {  
  Ball: {  
    model: null,  
    init: function(stage) {  
    },  
  },  
};
```



A yellow arrow originates from the `stage` parameter in the `init` function of the `View.Ball` object in the fourth code block and points to the `stage` parameter in the `init` function of the `Controller.Ball` object in the fifth code block.

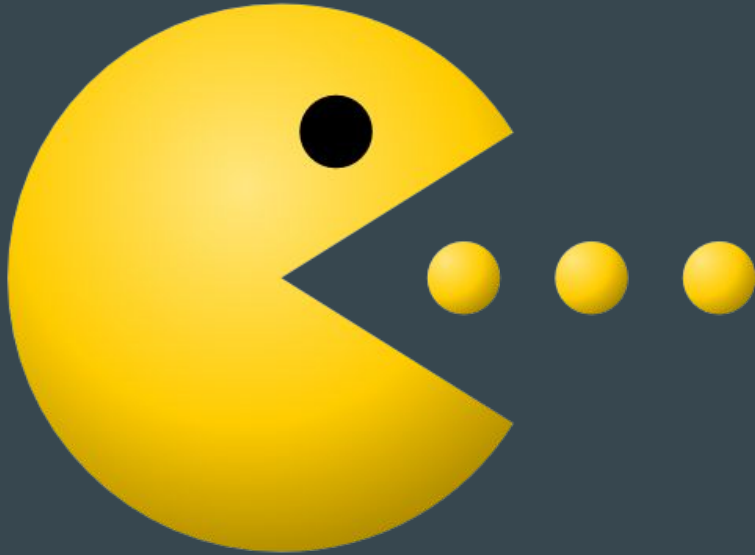
Exercise

Add flexibility

Homework 1

Implement a new widget called screensaver in the dashboard application using our newly adapted “ball object”.

Homework 2



Create our Pacman

Device Motion & Orientation

Device Motion & Orientation

Tap into the device's sensor (MEMs: Micro-electro-mechanical systems)

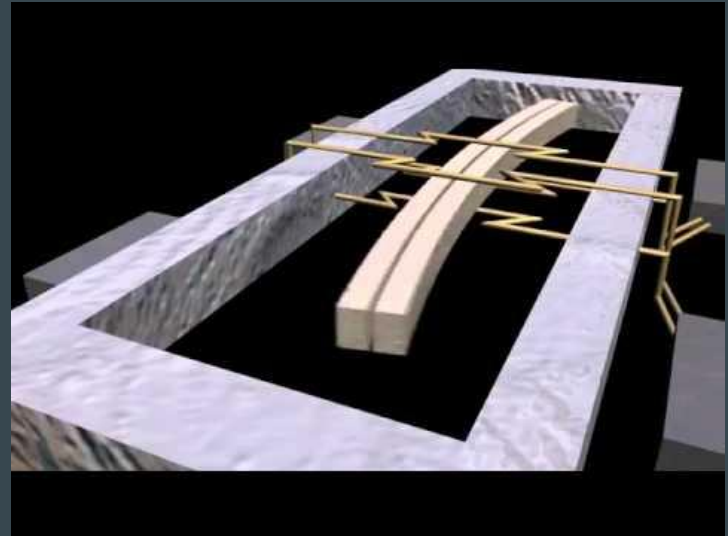
Accelerometer

Measures linear accelerations.



Gyroscope

Measures rotation changes.



Processing Motion Events

Event Handler should be added
to the window element



Device Motion Event has 4
different properties



```
window.addEventListener("devicemotion", handleMotion);
```

`DeviceMotionEvent.acceleration`

`DeviceMotionEvent.accelerationIncludingGravity`

`DeviceMotionEvent.rotationRate`

`DeviceMotionEvent.interval`

Demo

Possibilities



<https://itunes.apple.com/us/app/motion-synth/id945266624?mt=8>

Final Project

<https://github.com/leonardomra/web-apps-dev-course/tree/master/PE03-DeviceMotion/FinalProject>