

IDV460

INTERACTIVE DATA VIZ SPRING 16

JAVASCRIPT EVENTS

EVENTS

When you use the Web, the browser registers various types of events. It's the browser's way of noticing that something just happened. Your script can then respond to those events — by changing a color, perhaps, or showing some previously hidden text.

EVENTS

Events are based on user **interaction**. They occur when the user hovers over or clicks something, or types something on the keyboard, or perhaps when the page finishes loading.

*Interaction example:
User clicks on button*

When an event occurs, or **fires**, it can be used to trigger a specific function in the Javascript.

*Trigger example:
Function "showText" runs*

Scripts often respond to events by updating the content of the web page, which enhances the page's interactivity.

*Update example:
Text on page changes*

TYPES OF EVENTS

There are several different events that can occur in the browser. Any of them can trigger a function in your Javascript code.

UI EVENTS: load, unload, resize, scroll

KEYBOARD EVENTS: keydown, keyup, keypress

MOUSE EVENTS: click, dblclick, mousedown, mouseup, mousemove, mouseover, mouseout

FOCUS EVENTS: focus/focusin, blur/focusout

EVENT HANDLING

There are three steps involved to getting the HTML on your page to trigger Javascript code.

1. SELECT ELEMENT

Select the element node (or nodes) you want the event to respond to.

How? Use a **DOM query** to do this – like

- getElementById
- getElementsByClassName
- getElementsByTagName
- querySelectorAll

2. SPECIFY EVENT

Indicate which event on the selected node will trigger the code. This is called **binding** and event to a DOM node.

How? Create a variable using the DOM query, then follow it the name of the event, which is a keyword.

3. CALL CODE

State the code you want to run when the event occurs. When an event occurs on the specified element, it will trigger a **function**. The function can be named or anonymous.

BINDING

There are three ways to bind an event to an element.

1. HTML EVENT HANDLER ATTRIBUTES

Write the event handler into the HTML code as an attribute.

```
<button onclick="function()">Click me!</button>
```

2. TRADITIONAL DOM EVENT HANDLERS

In Javascript, name the element, then bind the event (adding "on") and state the function.

```
myButton.onclick = functionName;
```

3. USING EVENT LISTENERS

In Javascript, bind an event to an element by using a method that states the event and code.

```
myButton.addEventListener('click', functionName, boolean);
```

BINDING

There are three ways to bind an event to an element.

1. HTML EVENT HANDLER ATTRIBUTES

```
<button onclick="function()">Click me!</button>
```

POOR: This is bad practice, because it is better to separate JS and HTML.

2. TRADITIONAL DOM EVENT HANDLERS

```
myButton.onclick = functionName;
```

BETTER: Keeps JS separate, but you can only attach one function to any event.

3. USING EVENT LISTENERS

```
myButton.addEventListener('click', functionName, boolean);
```

BEST: More flexible because one event can trigger multiple events.

USING DIFFERENT EVENT HANDLERS

1. Open our **color.html** document, and remove the HTML event handler attribute from the button tag.

```
<button onclick="function()">Click me!</button>
```

```
myButton.onclick = functionName;
```

```
myButton.addEventListener('click', functionName, boolean);
```

USING DIFFERENT EVENT HANDLERS

1. Open our **color.html** document, and remove the HTML event handler attribute from the button tag.

```
<button class='myButton'>Click me!</button>
```

2. Open **color.js** and add, after the function, a line creating a variable for the button element.

```
var myButton = document.querySelector('.myButton');
```

3. Try adding each of the other event handlers – a traditional DOM event handler, and an event listener. Try changing the event, too – perhaps a double click or mouseover.

```
<button onclick="function()">Click me!</button>
```

```
myButton.onclick = functionName;
```

```
myButton.addEventListener('click', functionName, boolean);
```

USING DIFFERENT EVENT HANDLERS

1. Open our **color.html** document, and remove the HTML event handler attribute from the button tag.
2. Open **color.js** and add, after the function, a line creating a variable for the button element.
3. Try adding each of the other event handlers — a traditional DOM event handler, and an event listener. Try changing the event, too — perhaps a double click or mouseover.
4. Update your site with the new color documents.

With project one now complete, we'll look over what your live site should have right now.

YOUR SITE RIGHT NOW

IDV460

INTERACTIVE DATA VIZ SPRING 16

- **index.html**
- **class.html**
- cong.html
- deluna.html
- quiz.html
- primary.html
- deathchart.html
- timeline.html
- delegate.html
- demprimary.html
- time.html
- temps.html
- color.html
- income.html
- medianage.html
- **project.html**
- project.html

STEVE LAYTON

HOME

CLASS

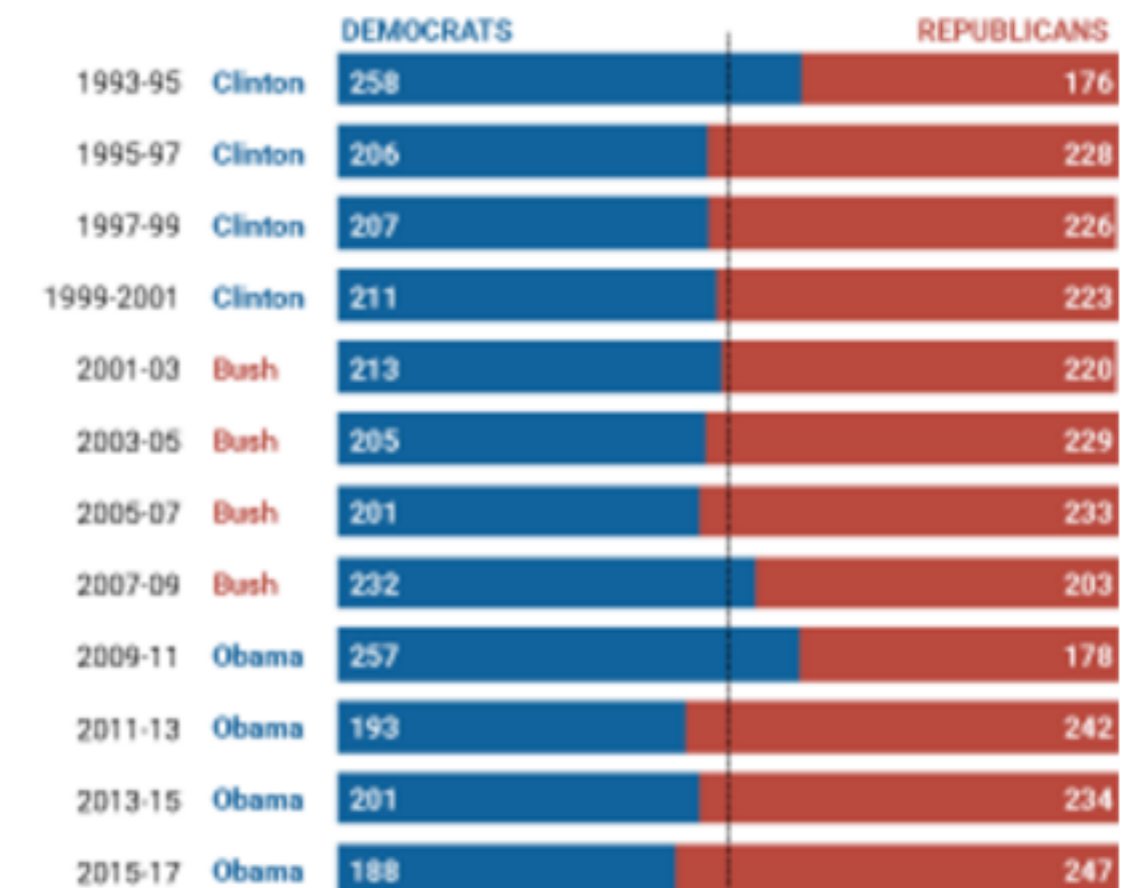
PROJECTS

In-class exercises

This page features the in-class exercises we have worked on in IDV460. Just click on the headline for each to view my work.

The balance of power in the U.S. House

This chart shows how the balance of power tends to ebb away from the president's party over the course of his two terms. We used this as a means of using an SVG graphic.



SOURCE: U.S. House of Representatives

YOUR SITE RIGHT NOW

- **index.html**
 - **class.html**
 - cong.html
 - deluna.html
 - quiz.html
 - primary.html
 - deathchart.html
 - timeline.html
 - delegate.html
 - demprimary.html
 - time.html
 - temps.html
 - color.html
 - income.html
 - medianage.html
 - **project.html**
 - project.html
- **style.css**
 - **class.css**
 - deluna.css
 - time.css
 - color.css
 - (project.css)

IDV460

INTERACTIVE DATA VIZ SPRING

STEVE LAYTON

HOME

CLASS

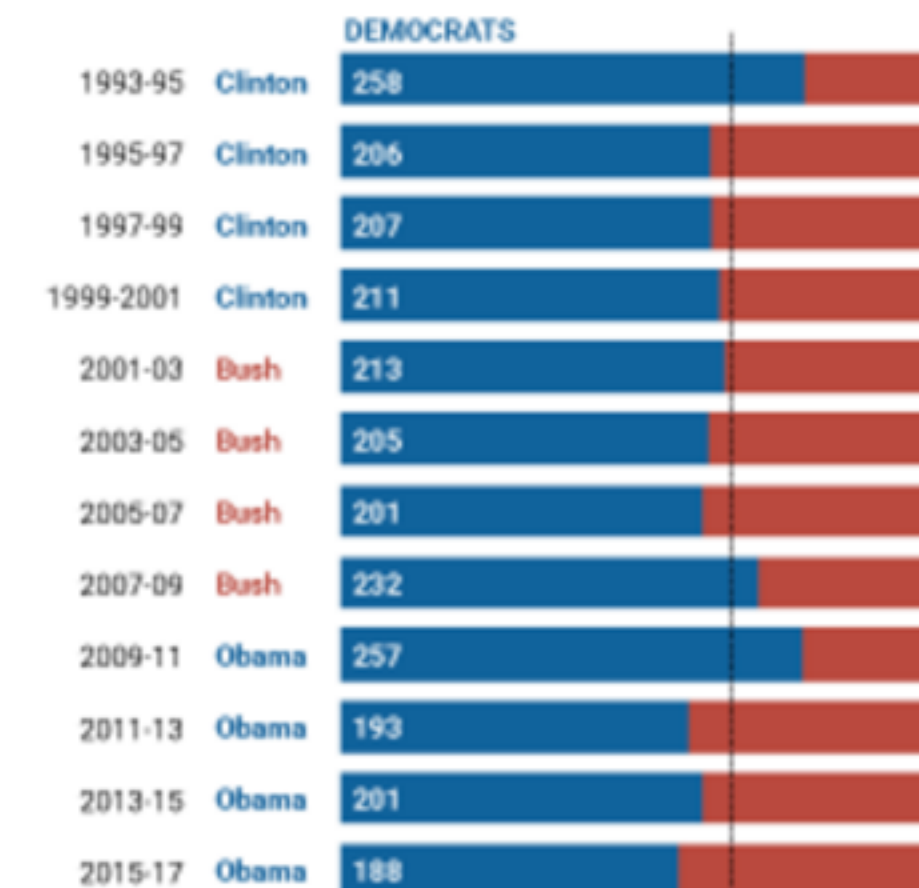
PROJECTS

In-class exercises

This page features the in-class exercises we have worked on in IDV460. Just click on the headline for each to view my work.

The balance of power in the U.S. House

This chart shows how the balance of power tends to ebb away from the president's party over the course of his two terms. We used this as a means of using an SVG graphic.



SOURCE: U.S. House of Representatives

YOUR SITE RIGHT NOW

- **index.html**
 - **class.html**
 - cong.html
 - deluna.html
 - quiz.html
 - primary.html
 - deathchart.html
 - timeline.html
 - delegate.html
 - demprimary.html
 - time.html
 - temps.html
 - color.html
 - income.html
 - medianage.html
 - **project.html**
 - project.html
- **style.css**
 - **class.css**
 - deluna.css
 - time.css
 - color.css
 - (project.css)

- primary.js
- death chart.js
- delegate.js
- demprimary.js
- greeting.js
- temps.js
- color.js
- income.js
- worldmap.js
- project.js

IDV

INTERACTIVE DATA

STEVE L

HOME

CLA

In-class exercises

This page features the in-class exercises with a link on the headline for each to view my work.

The balance of power in the U.S. Ho

This chart shows how the balance of power tends to ebb away from the president's party over the course of his two terms. We used this as a means of using an SVG graphic.

YOUR SITE RIGHT NOW

• index.html	• style.css		
• class.html	• class.css		
cong.html			
deluna.html	deluna.css		
quiz.html			
primary.html		primary.js	(Google)
deathchart.html		death chart.js	(Google)
timeline.html			(timeline.js)
delegate.html		delegate.js	highcharts.js
demprimary.html		demprimary.js	highcharts.js
time.html	time.css	greeting.js	
temps.html		temps.js	highcharts.js
color.html	color.css	color.js	
income.html		income.js	highmaps.js
medianage.html		worldmap.js	highmaps.js
• project.html			
project.html	(project.css)	project.js	

YOUR SITE RIGHT NOW

- **index.html**
 - **class.html**
 - cong.html
 - deluna.html
 - quiz.html
 - primary.html
 - deathchart.html
 - timeline.html
 - delegate.html
 - demprimary.html
 - time.html
 - temps.html
 - color.html
 - income.html
 - medianage.html
 - **project.html**
 - project.html
- **style.css**
 - **class.css**
 - deluna.css
 - time.css
 - color.css
 - (project.css)

- primary.js
- death chart.js
- delegate.js
- demprimary.js
- greeting.js
- temps.js
- color.js
- income.js
- worldmap.js
- project.js

jQuery

- (Google)
- (Google)
- (timeline.js)
- highcharts.js*
- highcharts.js*
- highcharts.js*
- highmaps.js*
- highmaps.js*

