



RICE®

Web Development

COMP 431 / COMP 531

Lecture 6: Storage and Canvas

Instructor: Mack Joyner

Department of Computer Science, Rice University

mjoyner@rice.edu

<http://www.clear.rice.edu/comp431>

Recap

- HTML
- JavaScript and Scope
- Forms
- CSS
https://www.clear.rice.edu/comp431/pdfs/lec_css.pdf
- Events

Homework Assignment 2
(Dynamic Page)
Due **TODAY** 9/14

Homework Assignment 3
(JS Game)
Due Thursday 9/28

A Brief History of HTML5



- Dec 1997 – HTML 4.0 (*strict/transitional/frameset*)
- 2008 – First Working Draft
- 2012 – Candidate Version (*browsers start your engines*)
- Oct 2014 – HTML5

Overview

- Quick Tour (with audio & video)
- LocalStorage
- SVG
- Canvas
- WebGL

<https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5>

HTML5 Elements

- **Semantic tags**

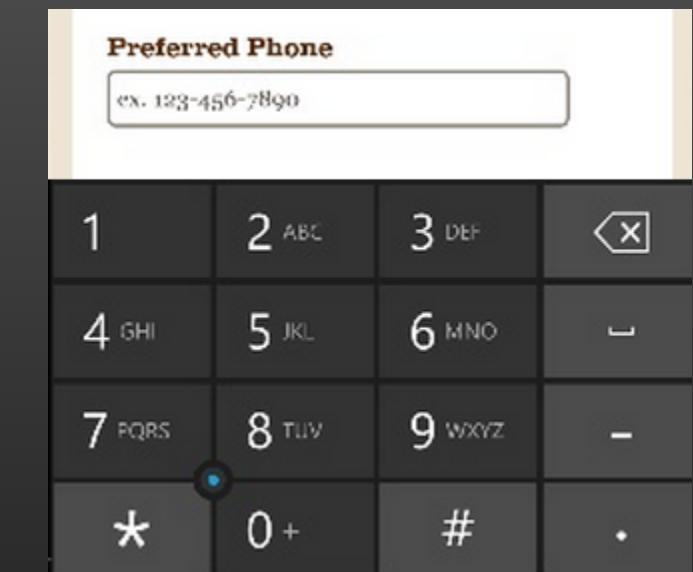
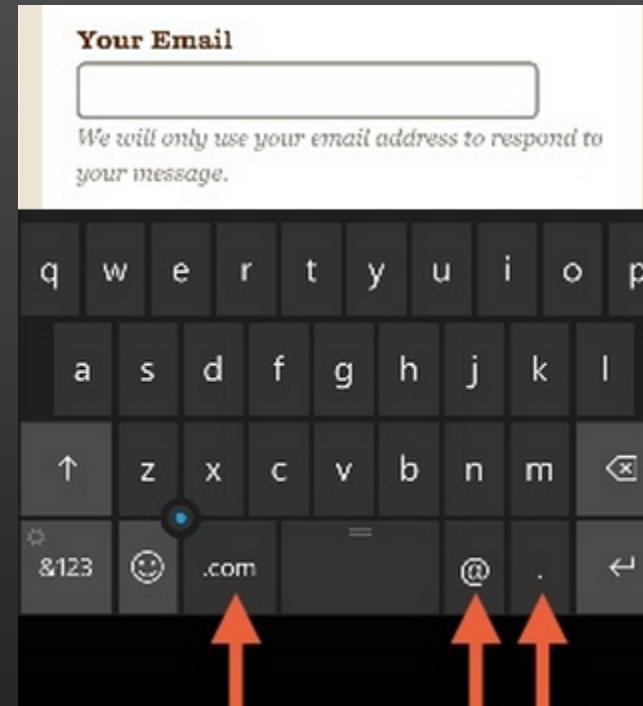
section, aside, header, footer
figure, fig caption, main, etc...

- **Embedded Media**

video, audio

- **Form improvements**

- Input types like email, tel, date, etc...
- Data validation



Preferred Date to Visit*

mm/dd/yyyy

September 2014 ▾

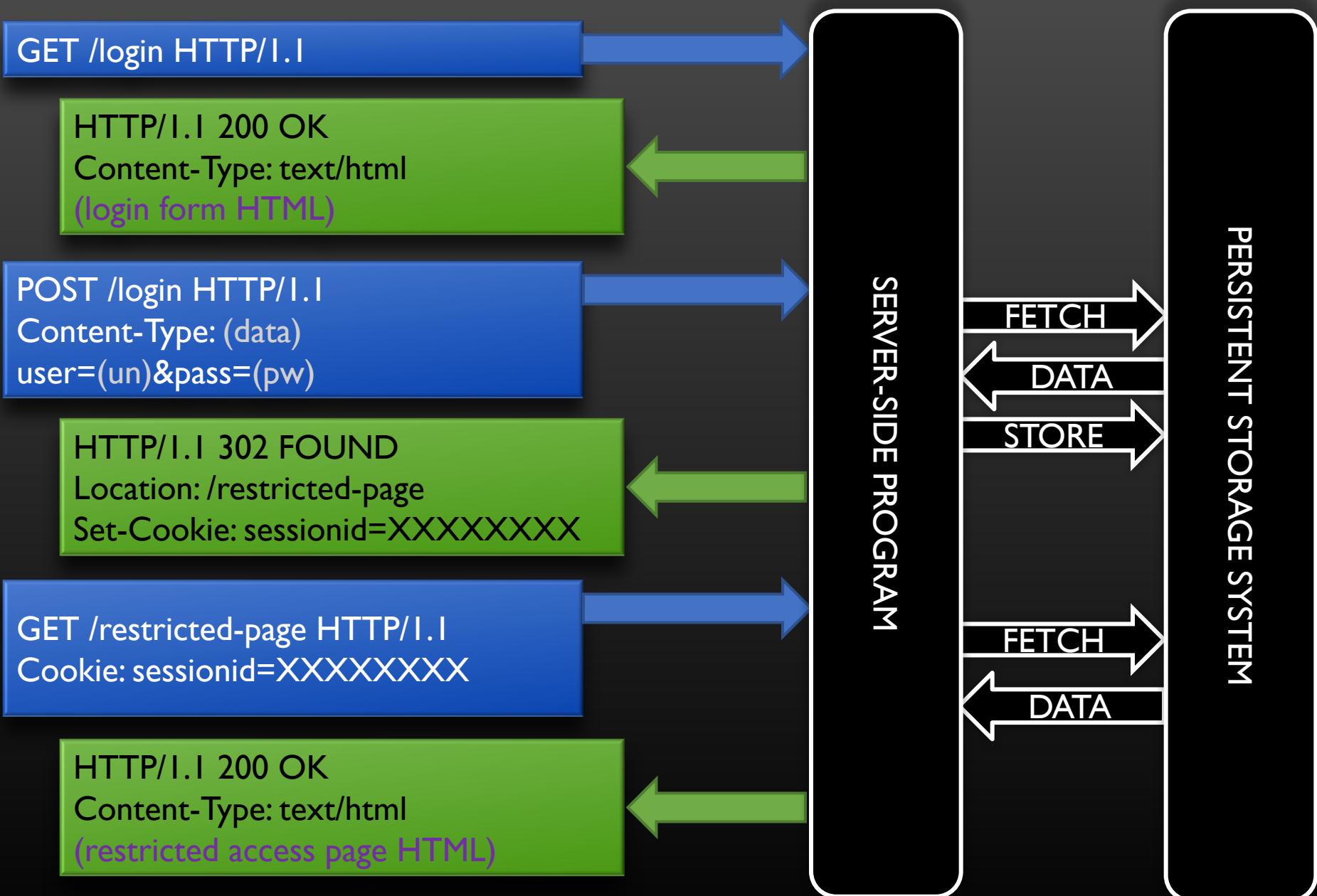
Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4

Form Example

<http://codepen.io/aarongustafson/pen/KppxxY>

Stateless Servers

- HTTP is a *stateless* protocol
- Each transaction is independent of previous ones
- There is no implementation in the protocol to retain state or session
- If you want state then you have to do it, i.e., by sending data back and forth in the payload



Note: "sessionid" will be attached to all future requests.

Courtesy of Matthew Schurr

Cookies

- It's a *magic cookie* (received and sent back unchanged)
- June 1994 – first introduced [Netscape]
- They are key-value pairs with zero or more attributes
- The server creates them
- Browser retains for the session or until some server declared expiration



Wikipedia's Example

```
GET /index.html HTTP/1.1  
Host: www.example.org
```

...

HTTP/1.0 200 OK

Content-type: text/html

Set-Cookie: theme=light

Set-Cookie: sessionToken=abc123; Expires=Wed, 09 Jun 2021 10:18:14 GMT

...

```
GET /spec.html HTTP/1.1
```

Host: www.example.org

Cookie: theme=light; sessionToken=abc123

...

Cookies are scoped

- Default is the domain and path for the page requested
- Can expand path as well as to entire domain, i.e., * . **foo . com**

```
HTTP/1.0 200 OK
Set-Cookie: lu=Rg3vHJZnehYLjVg7qi3bzjzg; Expires=Tue, 15-
Jan-2013 21:47:38 GMT; Path=/; Domain=.example.com; HttpOnly
Set-Cookie: made_write_conn=1295214458; Path=/;
Domain=.example.com
Set-Cookie: reg_fb_gate=deleted; Expires=Thu, 01-Jan-1970
00:00:01 GMT; Path=/; Domain=.example.com; HttpOnly
```

Cookies in JavaScript (*i.e., not HttpOnly*)

- In Chrome, file:// will not store cookies

```
> document.cookie  
< ""  
  
> document.cookie = "secret=message"  
< "secret=message"  
  
> document.cookie  
< ""
```

Cookies in JavaScript (*i.e., not HttpOnly*)

- In Chrome, file:// will not store cookies, so we'd move to Firefox

```
▶ document.cookie  
▶ "..."  
▶ ▶ document.cookie = "secret=message"  
▶ "secret=message"  
▶ ▶ document.cookie  
▶ "secret=message"
```

Cookies append

```
◀ document.cookie = "another=secret"  
▶ "another=secret"  
◀ document.cookie  
▶ "secret=message; another=secret"
```

Adding expiration

```
↳ expiry = new Date()
↳ Date 2015-08-24T02:36:14.482Z
↳ expiry.setDate(1)
↳ 1438482974482
↳ expiry
↳ Date 2015-08-02T02:36:14.482Z
↳ document.cookie = "stale=crumb; expires=" +
expiry.toUTCString()
↳ "stale=crumb; expires=Sun, 02 Aug 2015 02:36:14
GMT"
↳ document.cookie
↳ "secret=message; another=secret"
```

Cookies

```
document.cookie  
"secret=message; another=secret"
```

- To “use” a cookie in JavaScript you have to parse it out



- As you can see.... they really are crumby

Run a Local Web Server using Python [optional]

As just mentioned, `file:///` is not great for JavaScript, instead host locally:

```
python -m SimpleHTTPServer 8080
```

-OR-

```
python3 -m http.server 8080
```

Then navigate to <http://localhost:8080>

Storage in HTML5

- Cookies (IE2+)
 - Limited to **4095 bytes**
 - No expiration = session cookie
 - Expiration per cookie
- Local Storage (IE8+)
 - **5MB** per origin (browser dependent)
 - No expiration
 - ... UTF-16 (!) so really it's **2.5MB**
 - Compression? => **50 MB!**
- Indexed Database API (IE10+)
 - Sounds promising...
- Session Storage (IE8+)
 - Like local storage but released when session terminates
 - per origin per window

LocalStorage Example

```
if (window.localStorage) {
    document.getElementById('save').onclick = function() {
        localStorage.setItem('magic', 'This is my content')
    }

    document.getElementById('load').onclick = function() {
        var content = localStorage.getItem('magic')
        document.getElementById('content').innerHTML = content
    }
} else {
    window.alert('No localStorage support!')
}
```

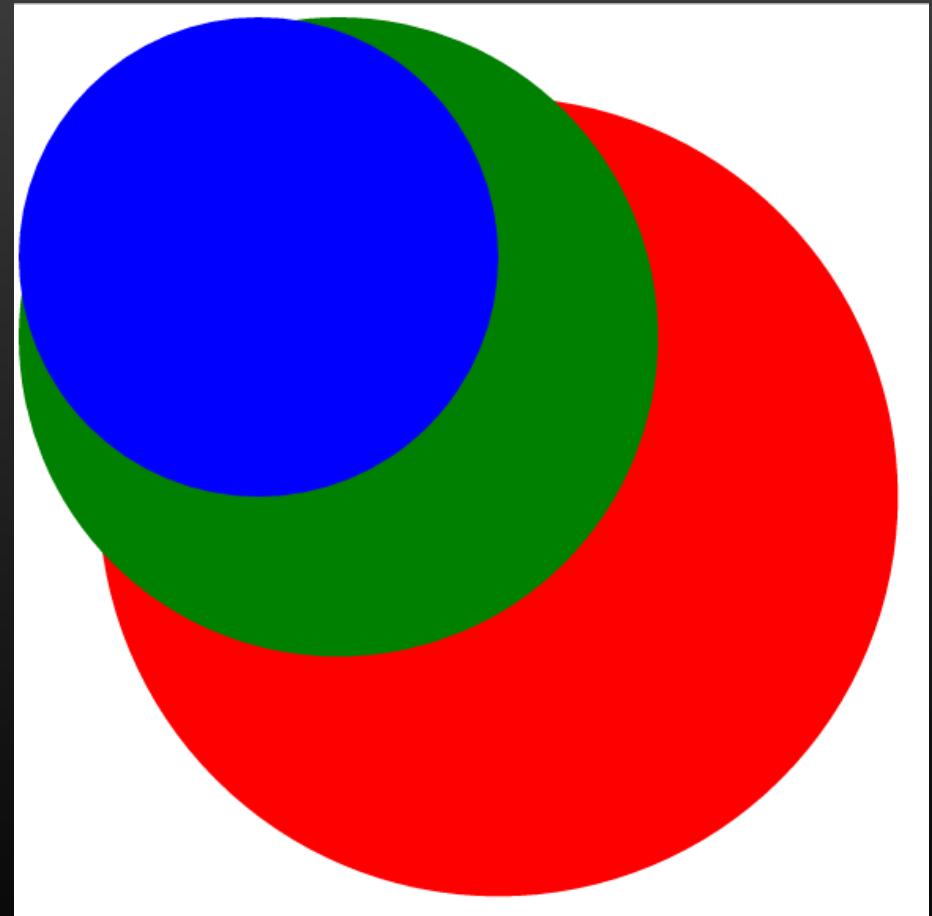
demo

Scalable Vector Graphics

XML-based vector image
format for 2D graphics.

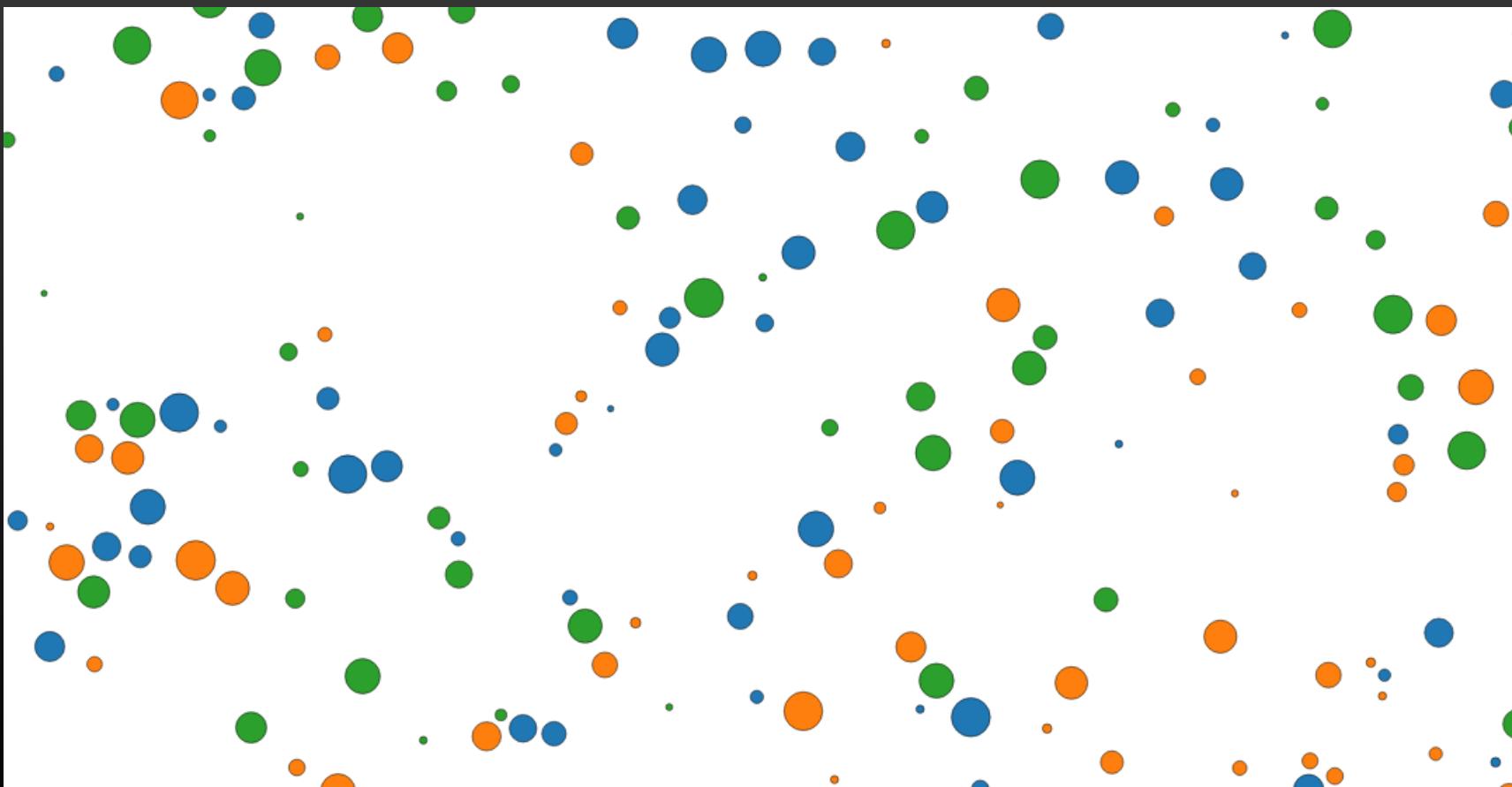
```
<body>

<svg viewBox="0 0 120 120">
  <circle cx="60" cy="60" r="50" fill="red" />
  <circle cx="40" cy="40" r="40" fill="green"/>
  <circle cx="30" cy="30" r="30" fill="blue"/>
</svg>
```



SVG Example

<http://mbostock.github.io/d3/talk/20111018/collision.html>

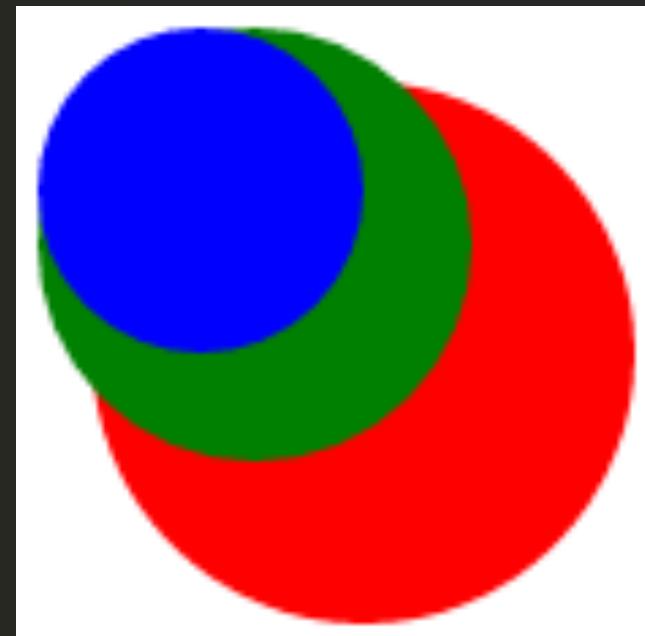


Canvas

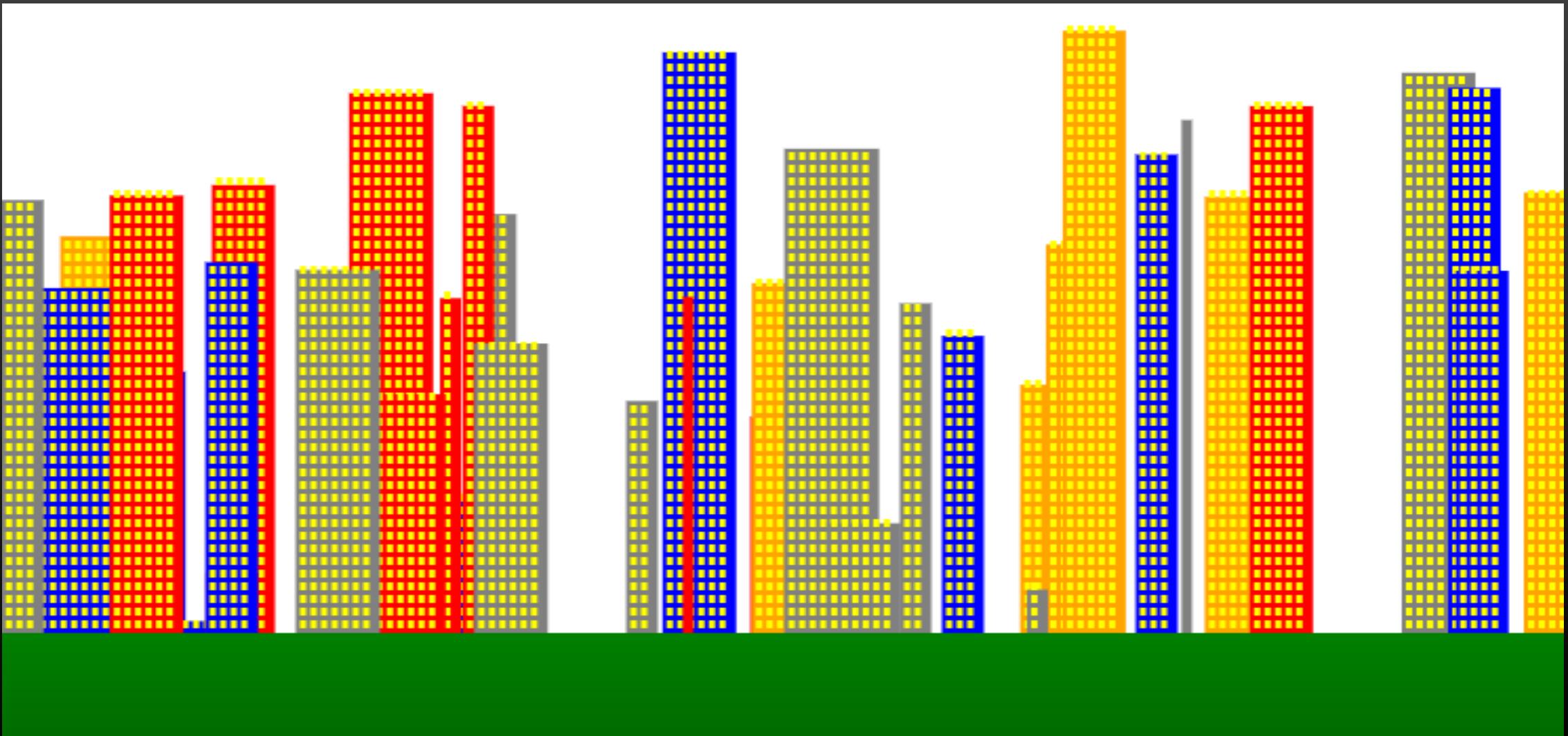
```
<canvas></canvas>
<script>
var c = document.querySelector("canvas").getContext("2d");

function fillCircle(x, y, r, color) {
    c.fillStyle = color;
    c.beginPath()
    c.arc(x, y, r, 0, 2 * Math.PI, false)
    c.closePath()
    c.fill()
}

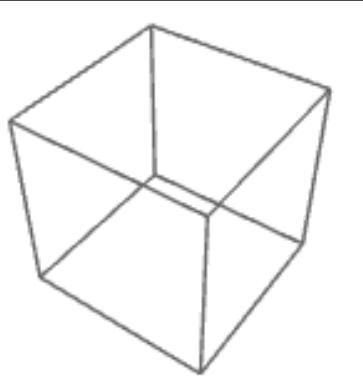
fillCircle(60, 60, 50, 'red')
fillCircle(40, 40, 40, 'green')
fillCircle(30, 30, 30, 'blue')
</script>
```



Canvas Example



WebGL



<https://get.webgl.org/>

<http://inear.se/beanstalk/>



<http://dinosaurpictures.org/ancient-earth/>