

# 授業レポート

女子美術大学短期大学部造形学科デザインコース  
竹田大純

## はじめに

2014年の女子美短大造形学科デザインコースのメディアデザイン（Web）では、Raspberry Pi + Coder で簡単なウェブアプリケーションを制作する授業を行いました。

デザイン手法や技術を学ぶことはあっても、普段日常的に利用している仕組みについて学ぶ機会はそう多くはありません。学生にとっては例えば制作のためのソフトウェアや、制作環境についてなどがそれにあたるかもしれません。

学校にあるコンピュータで単にウェブページを作るのではなく、普段使わないハードウェアを活用するカリキュラムにしたのは、デザインを学ぶ学生にはなかなか挑戦する機会が少ないのでと思ったことと、サーバーとクライアントコンピュータを目に見える形で手元に置いた上で制作させるのがネットワークを用いたデザインを考えるきっかけとして最適だと思ったためです。

## 授業について

Coder (<http://goo.gl/coder>) とは、Google 社の有志によって開発された IDE で、クライアントサイドとサーバーサイドのひと通りを学ぶのに適した Raspberry Pi 用のソフトウェアです。





授業ではセットアップの前に、まずインターネットとWWWの仕組み～ブラウザへのレンダリングまでを簡単に説明し、Raspberry Pi と Coder についてそれぞれ公式の動画を見ながら学習しました。

**What is a Raspberry Pi?**

<http://www.youtube.com/watch?v=e0wkVVVLvR8>

**Coder: A simple way to make web stuff on Raspberry Pi.**

<https://www.youtube.com/watch?v=wH24YwdayFg>



ひと通り説明した後の学生の反応で多かったのは、Raspberry Pi と普段使っているコンピュータ（授業では iMac を使用）が同じコンピュータであり、iMac と Raspberry Pi が同一の LAN 内にあってそれぞれのクライアントマシン（iMac）からサーバ（Raspberry Pi）にアクセスし、レスポンスをクライアントサイドで描画してそれを操作しているという状況がわからないというものでした。

これは正に普段閲覧しているウェブページ（アプリケーション）が、どこかのサーバーに置いてあり、そのサーバに自分の端末からリクエストした結果を表示しているということと同じなのですが、その仕組みを実際に手の届く範囲で用意し、ぼんやりとでもイメージしてみると少なからず理解が深まったように感じました。

```
1 | // ...
2- $(document).ready( function() {
3 |
4     var centered = document.getElementsByClassName('centered');
5     var eyes = document.getElementsByClassName('eye');
6     var eyeRadius = 100;
7     var mouseX = 0.0;
8     var mouseY = 0.0;
9 |
10    // Returns the pixel value of a number, e.g. '42px'
11    function toPixels( number ) {
12        return Math.round( number ) + 'px';
13    }
14 |
15    // Applies the CSS3 blink animation to the eyes
16    function blink() {
17        for ( var i = 0; i < eyes.length; i++ ) {
18            var eye = eyes[i];
19            var container = eye.getElementsByClassName('container')[0];
20            container.style.webkitAnimation = '';
21 |
22            // Since the animation property was already
23            // set, we need to wait until the next frame to
24            // register as a change
25            setTimeout( function() {
26                container.style.webkitAnimation = 'blink 1s';
27            }, 0 );
28        }
29        setTimeout( blink, 4000 + Math.random() * 500 );
30    }
31 |
32    function updateEyes() {
33        for ( var i = 0; i < eyes.length; i++ ) {
34            var eye = eyes[i];
35            var container = eye.getElementsByClassName('container')[0];
36            var mouseX = eye.offsetLeft + eye.offsetWidth / 2;
37            var mouseY = eye.offsetTop + eye.offsetHeight / 2;
```

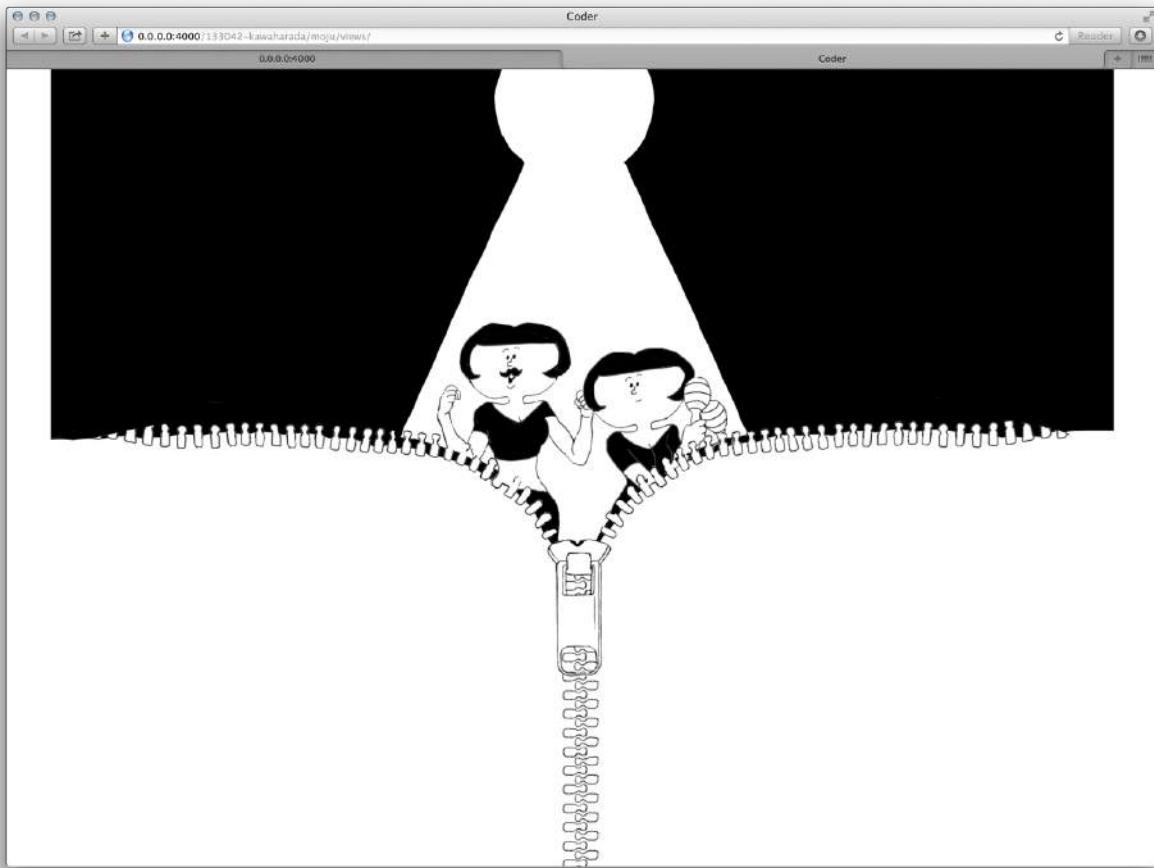
次に Coder の「Hello Coder」や「Eyeball」というアプリケーションを用いてクライアントサイドとサーバーサイドとの間でどういったことが行われているのかを実際に IDE を操作しながら学びました。ここでは意味と表示、振る舞いを HTML / CSS / JavaScript でそれぞれ分担しているということを学習し、自分でページやアプリケーションを作る際に注意すべき点は何かという点を確認しました。



ひとり通りのオリエンテーションを終えたのち、まず表示するだけの静的なページを作成しました。内容は任意の料理のレシピページで、主に論理マークアップとレイアウトのトレーニングのためでした。WYSIWYGでのデザイン制作ではないので、慣れるまで苦労していましたようでしたが、普段目にする見慣れた内容のものを意味 / 表示 / 振る舞いに分解し、コードを書いて組み上げていくという作業はそれなりに新鮮なものであったようです。

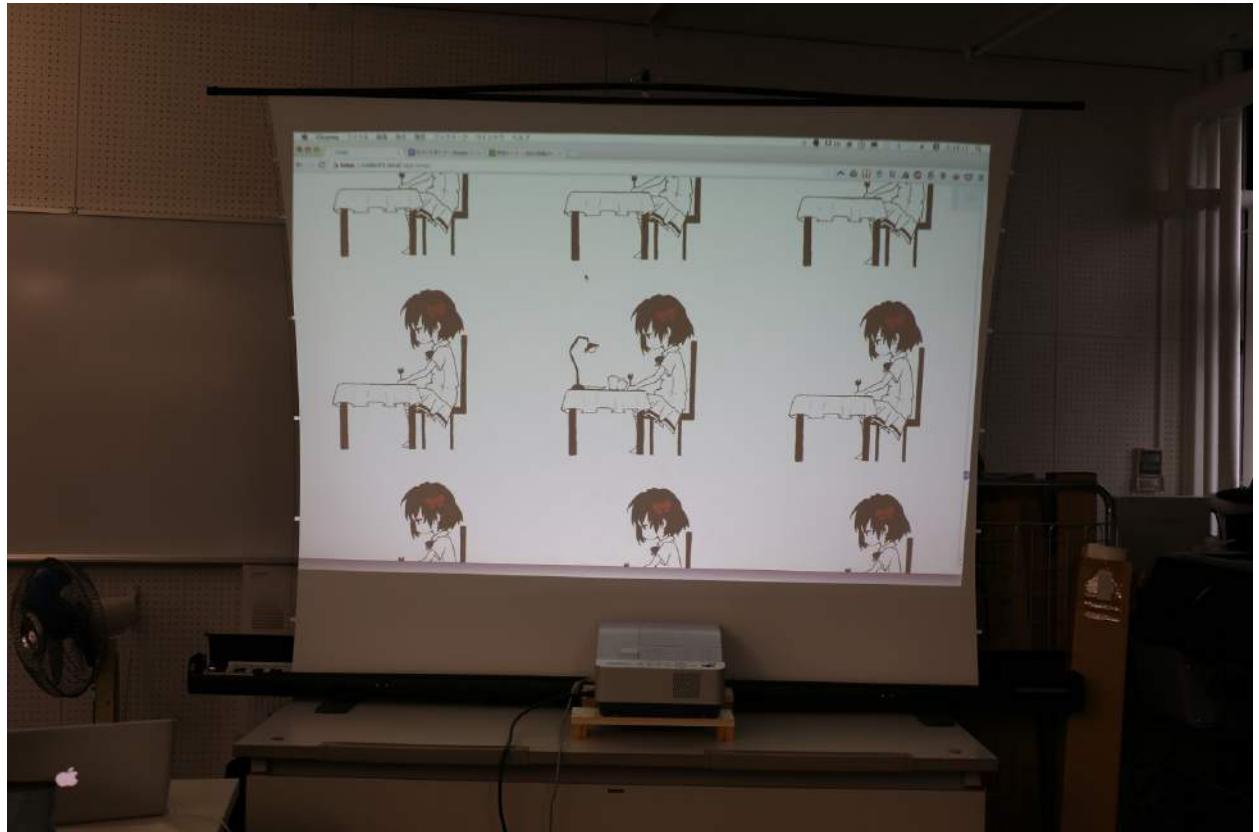


2つ目の課題として、Coder の「Pop-Up Penguin Game([http://googlecreativelab.github.io/coder-projects/projects/pop\\_up\\_penguins/](http://googlecreativelab.github.io/coder-projects/projects/pop_up_penguins/))」、「Night and Day ([http://googlecreativelab.github.io/coder-projects/projects/night\\_and\\_day/](http://googlecreativelab.github.io/coder-projects/projects/night_and_day/))」などでクリックやドラッグ、スクロールなどのアクションにイベントを用意する、というところから学習し、こうしたインタラクションを用いた動きのあるウェブアプリケーションを作成することにしました。



ここではトラックパッド / マウスホイールのスクロールでフレーム操作ができるような内容のものを制作することにしました。CSSプロパティを動的に操作するというよりは、わかりやすさから画像をフレーム毎に切り替える手法を使う学生が多いようでした。また、スクロールバーが最下部に達すると上部に戻し無限ループするようなスクリプトを用意し、ループアニメーションのようなものを作る学生もいました。

## 講評会を終えて



今回 Raspberry Pi + Coder を使用したメリットとして実感できたのは、

- サーバーとクライアントコンピュータ、ネットワークの最小構成を触りながら学習できたこと
- Coder が IDE として優秀だったこと
- Raspberry Pi がウェブサーバーとして使用できることが使い方の一部にすぎない（もっと色々なことにも使用できる）こと

の3点です。

主に3点目のコンピュータに対してソフトウェア次第でどんな役割を与えることもできるという点や、ネットワークを通じてできることは考え方次第で様々な可能性があることを学べたという点が本年度の授業での成果だったのではと感じました。

情報デザインというアプローチからもこういった領域に踏み込んでいく機会は今後もっと増えていくでしょうし、早期からこうした理解があればもっと柔軟に色々なことに取り組んでいけるのではと思います。

そういう意味では Scratch などをきっかけとしてプログラミングに馴染んだ学生が次のステップとしてスクリプト言語を学ぶとすれば、環境として Raspberry Pi と Coder は良い組み合わせなのかも知れません。