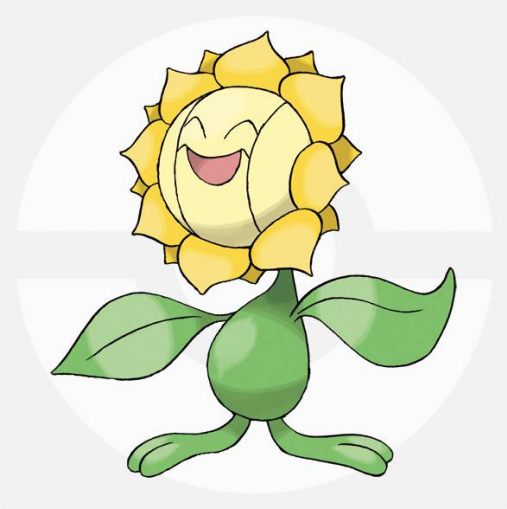
Webデザイン実習3B

2019/4/15

Kazuma Sekiguchi
class@cieds.jp

担当



- 関口　和真（Kazuma Sekiguchi）
  そのうち見飽きるでしょうけど、顔はなかなか変わらない
  ←手足はもう少ししっかりしている

- 弱点
  いつでも眠い

  - Comcent,Inc（今年も誰か採用したい）
  - Webプログラマ兼コーダーとかいろいろ
  - デザインもごく希にやります
  - たまにWebに記事を書いていたり本を書いていたりします
  - Adobeの多少なりとも関係者
  - 絵は全く描けないので聞かないように

Agenda

- インタラクションの付与
  - 先にサイトデザインを作成して、それを元に良く利用されいるパーツやUIについてJSでどう実現するかを学習していく

そのために

- JSの基本知識
- JSでのインタラクション付与の仕方
- インタラクション性のあるWebサイトの作成
- スマートフォンでのインタラクティブ性

JSとか

- 年々JSの重要性は増している
  - JSでインタラクションを付与する
  - JSでSPA（SinglePageApplication）を作成する
  - JSでスマートフォンアプリを作成する
  - JSでデスクトップアプリを作成する

- JSは取っつきにくい言語
  - あまり効率的では無い言語
  - 全部を1から書ける必要は無い（今のところは）
  - 何ができるのか、どう書いていくのか、どういうときにインタラクションを付与すれば良いかを考える

JSとか

- JSはHTMLとCSSができていることが前提
  - HTMLとCSSがまだ苦手な人は復習しておく

- JSはプログラム＝一度では上手くいかない
  - トライアンドエラーを繰り返す必要はある
  - ある程度諦めない、辛抱強さは必要

UX

- ユーザーに利便性を感じてもらって、使いやすいサイト、サービスを構築していくことの重要性
  - 作成者の独りよがりにならないことが重要になってきている
  - システム上の制約、考え不足、そぎ落としの足りなさ
- できるだけ考えて、最適解を見つける
  - それでも実際には最適解では無い
  - フィードバックが重要
  - フィードバックを受けて更に改善するサイクルを高速にまわす

やって欲しいこと

You should see any web sites and do more.

やって欲しいこと（1）

- 様々なウェブサイトを見る
  - 趣味のサイトは意識せずに見るはずなので、趣味以外のサイトを見る
  - インタラクション、インターフェイスを良く観察する
  - Dribble、Behance辺りはオススメ
- スマートデバイスを意識したUI/UXの作成
  - PC向けサイトの重要性は低い
  - スマートフォン向けにどういうメリットを出せるかを考える

やって欲しいこと（2）

- Webは進化が早い
  - 技術的なことは直ぐに陳腐化していく
    - 貪欲に調べる、やってみる
  - 基本原理は継続しているため、基本原理をきちんと把握する
    - 何かを解消するために新しい技術が出てくる
    - 何を解消しているのかを把握する
  - 技術が伴うことで表現できるものも多い
    - 最近のWeb表現はWeb技術と上手く融合している例が多い

やって欲しいこと（3）

- 勉強会などに出席してみる
  - 大体無料であちこちにて開催されている
  - 話が分からなくても刺激になることも多い
  - ヒントが得られることも
  - 何となく知っているようなものへの出席がオススメ
  - 簡単な名刺とか持って行って懇親会とかで出せば、向こうもくれるはず
- 好奇心
  - ちょっとでも知っていると全く知らないでは大違い
  - 意外な知識が意外なところで役立つことも

CSS3を利用した動き

- CSS3からtransformプロパティが利用可能
  - 移動
  - 拡大、縮小
  - 回転
  - 傾斜変形

CSS3　移動

- transform:translate（X方向の距離、Y方向の距離）
- transform:translateX()
  - X方向の距離
- transform:translateY()
  - Y方向の距離
- transform:translateZ()
  - Z方向の距離
- transform:translate3d（X方向の距離、Y方向の距離、Z方向の距離）

CSS3　拡大・縮小

- transform:scale（X方向の比率、Y方向の比率）
- transform:scaleX（X方向の比率）
- transform:scaleY（Y方向の比率）
- transform:scaleZ（Z方向の比率）
- transform:scale3d（X方向の比率、Y方向の比率、Z方向の比率）
  - Z方向の比率は機能しない

CSS3　回転

- transform:rotate（回転角度）
- transform:rotateX（回転角度）
  - X軸を軸とする角度によって時計回りの回転
- transform:rotateY（回転角度）
  - Y軸を軸とする角度によって時計回りの回転
- transform:rotateZ（回転角度）
  - Z軸を軸とする角度によって時計回りの回転
- transform:rotate3d（方向ベクトル、方向ベクトル、方向ベクトル、回転角度）

- 角度の単位はdeg
  - 10deg＝10度

CSS3　傾斜変形

- transform:skewX（X軸の傾斜角度）
- transform:skewY（Y軸の傾斜角度）
- transform:skew（X軸の傾斜角度，Y軸の傾斜角度）

- 角度を指定するので、degで指定する

CSS3　アニメーション

- transitionで指定するとアニメーション可能

```
.prefix_sample {
background-color:blue;
width:200px;
height:50px;
transition: background-color 1s, width 1s, height 1s;
}
.prefix_sample:hover {
background-color:aqua; width:400px; height:100px;
}
```

- :hoverなどの疑似イベントなどを利用してアニメーションさせる

CSS3　animation

- @keyframes 名前でアニメーションを定義する
- CSSのanimationプロパティで@keyframesで指定したアニメーションを指定すると動作する
- 現時点ではブラウザー接頭辞が（一応）必要

```
.sample {
animation: anime1 5s ease infinite alternate;
}

@keyframes anime1 {
0% {width: 50px; height: 50px; background-color: #F90;}
100% {width: 200px; height: 50px; background-color: #09F;}
}
```

CSS3 Animationで使えるプロパティ

- animation-name：アニメーション名
- animation-duration：アニメーションの長さ
- animation-timing-function：アニメーションの動き方
- animation-delay：アニメーションの開始を遅らせる
- animation-iteration-count：アニメーションの繰り返し回数
- animation-direction：アニメーションを交互に反転再生させるかどうか（normalかalternate）
- animation-fill-mode：アニメーションの前後のスタイルを指定（none,forwards,backwards,both）
- animation-play-state：再生中か一時停止かを指定