

情報学部 プログラミング体験教室 (初級編)

長嶺也・丸山一貴
10:10 ~ 11:30

1

サンプルゲームを動かそう！

- まずは実行してみましょう
 - デスクトップのoc2016を開きます
 - index.htmlを開きます
 - ブラウザが起動します
 - キャラクターはマウスポインタについてきます
 - たまに弾を撃ちます
- 実行をやり直す場合は、再読み込みボタンを押します



2

プレイヤーの画像を変えよう！(1)

- neko.pngを開く → 表示したい絵を選ぶ
- 絵の番号を調べる



3

プレイヤーの画像を変えよう！(2)

- main.jsを開く(テキストエディタが起動)
- class Player という行の場所まで移動する
- プログラムを追加する。

this.p=\$pat_neko+選んだ絵の番号;
例: this.p=\$pat_neko+45;

```

19 |
20 | class Player extends Actor {
21 |     *main() {
22 |         this.p=$pat_neko+45;
23 |         this.count=0;
24 |         while (true) {
25 |             this.x=$mouseX;

```

- Ctrl+Sで保存しよう
- ブラウザを再読み込みして確認してみよう

4

弾の画像を変えよう！

- 同じように一枚絵を選ぶ
- class PlayerBullet という行の場所まで移動する
- プログラムを追加する。

this.p=\$pat_neko+選んだ絵の番号; 例: this.p=\$pat_neko+48;

```

27 | class PlayerBullet extends Actor {
28 |     *main() {
29 |         this.p=$pat_neko+48;
30 |         var vx=10;
31 |         while (this.x<400) {
32 |

```

実行(再読み込み)して確認してみよう！

5

敵を出してみよう！

- 今の状態では敵が出てこない...
- class Start という行の場所まで移動する
- プログラムを追加する。

new Enemy({ x:300, y:200, vx:-3, vy:1 });

```

10 | $score=0;
11 | class Start extends Actor {
12 |     *main() {
13 |         new Player;
14 |         new Enemy({ x:300, y:200, vx:-3, vy:1 });
15 |     }
16 | }
17 |

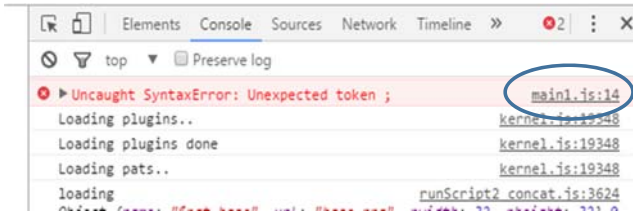
```

: (コロン)と ; (セミコロン)
の違いに注意！

6

エラーが出たら…

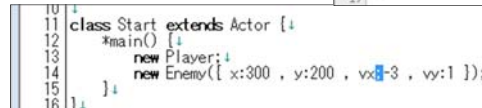
- ブラウザの画面をクリック
- Ctrl+Shift+J を押して、「コンソール」を表示
- エラーメッセージの左端 main].js:XX をクリック



7

エラーが出たら…

- エラーの発生箇所を確認
- テキストエディタで直す



8

敵と敵の弾の画像を変えよう！

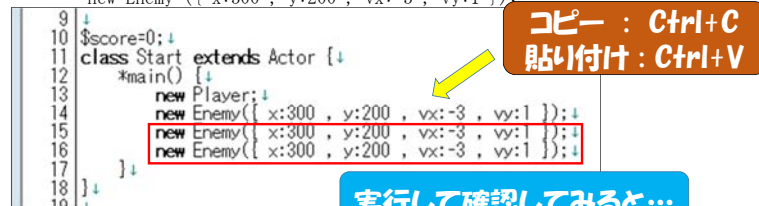
- 一枚絵を選ぶ
- class Enemy という行の場所まで移動する
class Enemy {
 *main () {
 this.p=\$pat_neko+**選んだ絵の番号**;
- 「EnemyBullet」も同様に行なう

実行して確認してみよう！

9

敵をたくさん出してみよう！(1)

- 敵が一体しか出てこない…
- class Startに行き、先ほど追加したプログラムをコピーする。
new Enemy ({ x:300, y:200, vx:-3, vy:1 });

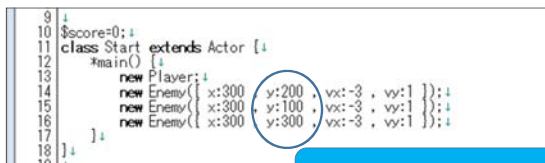


実行して確認してみると…

10

敵をたくさん出してみよう！(2)

- やっぱり敵が一体しか出てこない？
- …実は3体重なっている
- それぞれのyの値を変えてみよう



実行して確認してみよう！

11

敵をたくさん出してみよう！(3)

- 敵の出現位置をランダムにしてみよう
- rnd(数値)→0から数値未満のランダムな値

```
new Enemy({ x:300, y:this.rnd(400), vx:-3, vy:1});
```



実行して確認してみよう！

12

敵をたくさん出してみよう！(4)

- 敵の移動速度をランダムにしてみよう
- rnd(5): 0から4の乱数
- rnd(5)-2 -2から2の乱数

```
new Enemy({ x:300, y:this.rnd(400), vx:-3, vy:this.rnd(5)-2});
```

実行して確認してみよう！

13

敵をたくさん出してみよう！(5)

- ずっと敵が出てくるようにしたい！
- 繰り返し文(while文)を使う
- yield; → 処理を少し待つ

```
class Start extends Actor {
  *main() {
    new Player;
    while(true) {
      new Enemy({ x:300, y:this.rnd(400),
        vx:-3, vy:this.rnd(5)-2 });
      yield;
    }
  }
}
```

実行して確認してみよう！

14

敵をたくさん出してみよう！(6)

- 敵が出すぎ！
- ランダムで出すようにする

```
class Start extends Actor {
  *main() {
    new Player;
    while(true) {
      if (this.rnd(100)<10) {
        new Enemy({ x:300, y:this.rnd(400),
          vx:-3, vy:this.rnd(5)-2 });
      }
      yield;
    }
  }
}
```

実行して確認してみよう！

15

スコアを表示してみよう！

- 画面にスコアを表示させる
- 「Start」の最初にプログラムを追加

```
class Start extends Actor {
  *main() {
    new Player;
    new Score({x:200, y:300});
    while(true) {
      if (this.rnd(100)<10) {
        new Enemy({ x:300, y:this.rnd(400),
          vx:-3, vy:this.rnd(5)-2 });
      }
      yield;
    }
  }
}
```

自分の好きな位置に変更しよう

実行して確認してみよう！

16

パラメータの変更

- パラメータを変えるとゲームのバランスが調整できます
- プレイヤーが弾を撃つ頻度を変える
 - Playerのプログラムを一部変更

```
if(this.count==30){
// 30の値を多くすると撃つ頻度が下がり、少なくすると頻度が上がる
```

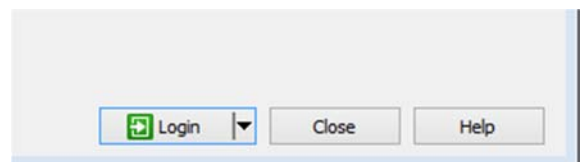
- 敵が弾を撃つ頻度を変える
 - Enemyのプログラムを一部変更

```
if(this.rnd(100)<1){
// 1の値を多くすると撃つ頻度が上がる
```

17

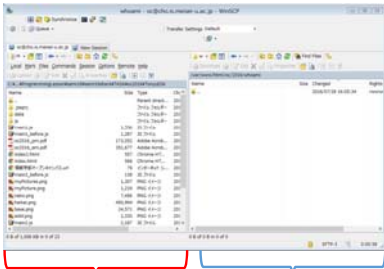
スマートフォンで遊んでみよう！(1)

- デスクトップの「ファイル転送」アプリを起動
- Loginを選択



18

スマートフォンで遊んでみよう！(2)

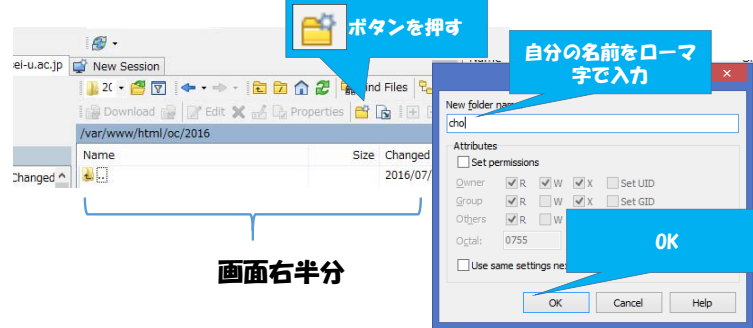


画面左半分
手元のPCのフォルダ

画面右半分
公開用サーバのフォルダ

19

スマートフォンで遊んでみよう！(3)

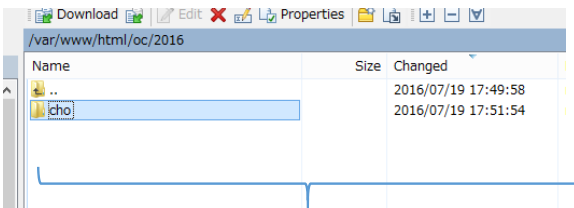


画面右半分

20

スマートフォンで遊んでみよう！(4)

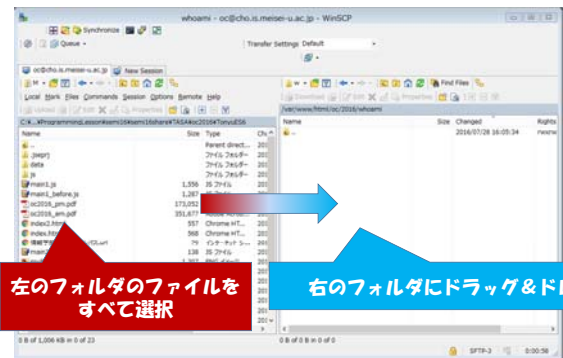
- 作成した自分のフォルダを開く(ダブルクリック)



画面右半分

21

スマートフォンで遊んでみよう！(5)



左のフォルダのファイルを
すべて選択

右のフォルダにドラッグ&ドロップ

22

スマートフォンで遊んでみよう！(6)

- スマートフォンで、次のアドレスを開く
<http://cho.is.meisei-u.ac.jp/oc/2016/>
- 自分の名前をさがして選択



23

最後に

- 今日作成したプログラムは、次のWebブラウザで動作します
 - Google Chrome
 - Firefox
- ※必ず最新のバージョンをお使い下さい
- 詳しい解説は次のページでも見られます
<http://cho.is.meisei-u.ac.jp/profile/>

24