GAME-MASTER MISSION_1

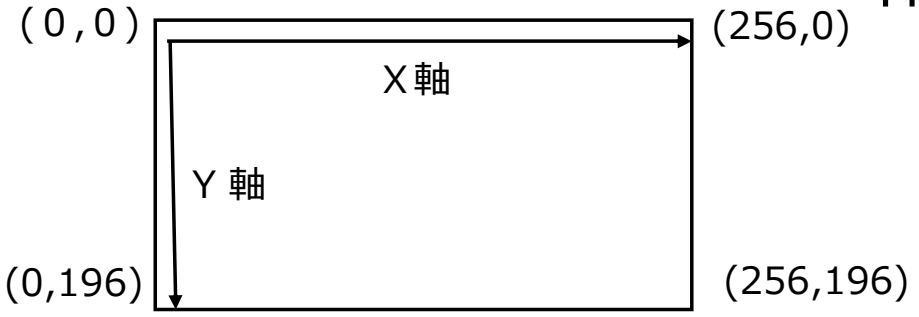
アニメでゲーム作りを学ぼう（1/3）

ホワイトボード

ゲーム開発マスター　MISSION 1

ピクセルマン the アニメを作る（1/3）

●ゲーム制作における座標



●ゲームのプログラムの基本

```
import pyxel              ←Pyxelﾓｼﾞｭｰﾙを読み込む
class Anime :             ←設計図(クラス)「Anime」の定義がはじまる
    def __init__(self) :
        ～                ←ここに初期処理を書く
    def update(self) :
        ～                ←ここにキャラの座標の更新(移動)を書く
    def draw(self) :
        ～                ←ここにキャラ（ピクセルマン）の描画を書く
Anime()                   ←設計図「Anime」を動かす
```

■時間割

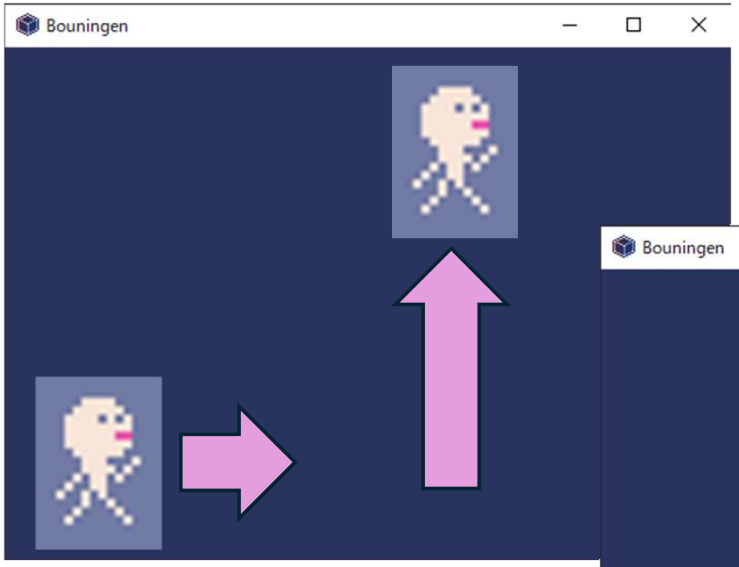
MISSION_1

寿司打	5分
ｷｬﾗ(ﾋﾟｸｾﾙﾏﾝ)作り	20分
勝ってに右に走る(基本1)	60分
勝ってに右に走る(基本2)	45分
キーを押すとジャンプする	45分
寿司打	5分

合計　180分

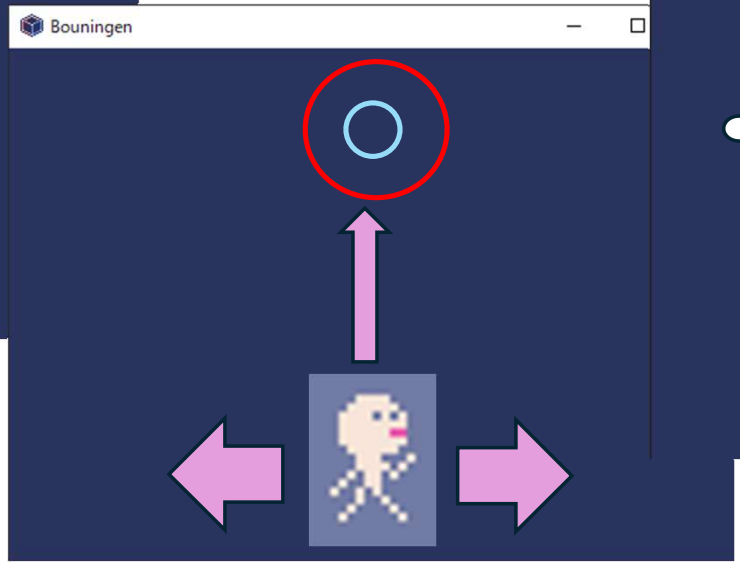
アニメでゲーム作りを学ぶ

今回(MISSION_1)



ﾋﾟｸｾﾙﾏﾝ、走る、ジャンプ

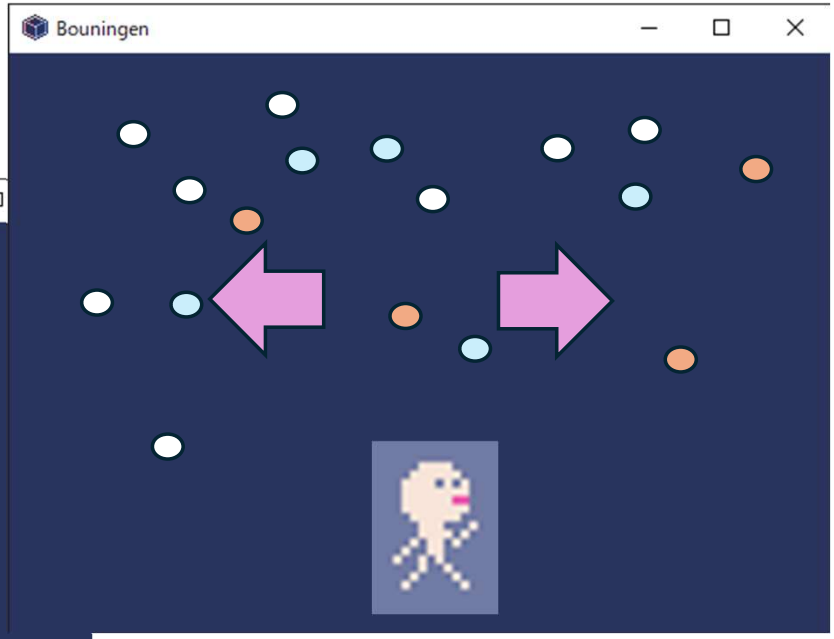
次回(MISSION_2)



ﾋﾟｸｾﾙﾏﾝ、左右に走る、花火を上げる

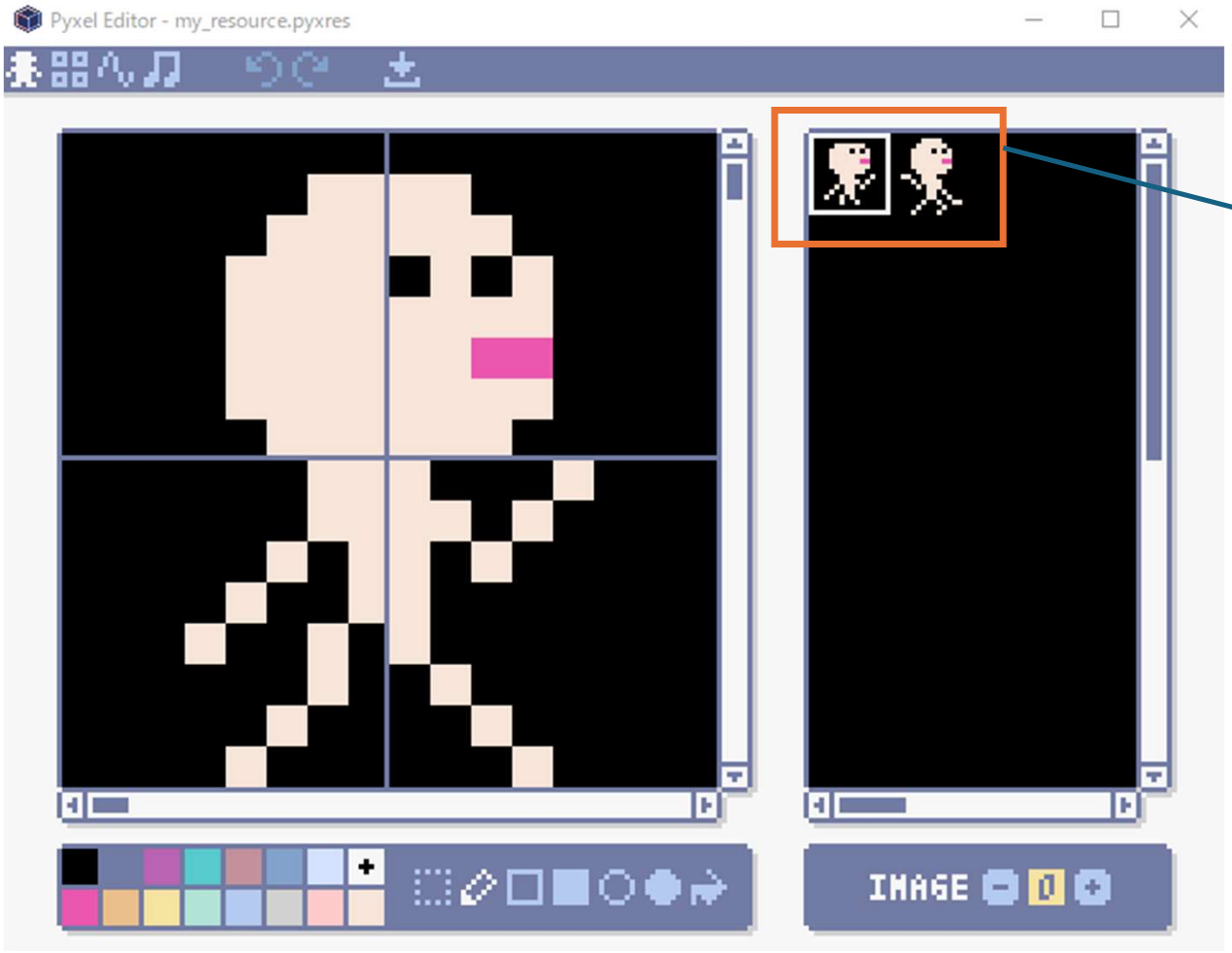
次々回(MISSION_3) （完成）

ﾋﾟｸｾﾙﾏﾝ、走って、ジャンプ、花火を上げる、
背景を動かして完成させよう！



pyxel editでキャラ(ピクセルマン)を作ろう

・この画面で背景の画像や、効果音、BGMも作れる。



走っているような
ピクセルマンを2種類
作る

ゲームプログラムの基本1

プログラムを打ち込んで、動かしてみよう！

- クラスという機能を使って、プログラムの設計図を作って、それを必要な時に実体化させる。

```
import pyxel                ←Pyxelﾓｼﾞｭｰﾙを読み込む

class Anime :               ←設計図「Anime」の定義がはじまる
    def __init__(self) :
        ～                  ←ここに初期処理を書く                    （最初に1回だけ実行）
    def update(self) :
        ～                  ←ここにキャラの座標の更新(移動)を書く（1秒間に30回更新）
    def draw(self) :
        ～                  ←ここにキャラ（ﾋﾟｸｾﾙﾏﾝ）の描画を書く　（1秒間に30回描画）

Anime()                     ←設計図「Anime」を動かす（クラスを実体化させる）
```

ゲームプログラムの基本2(1/2)

プログラムを修正して、同じ動きになることを確認しよう！

- プログラム全体だけでなく、キャラも別のクラスで定義する。

```
import pyxel              ←Pyxelﾓｼﾞｭｰﾙを読み込む

class Pixelman :          ←設計図「Pixelman」の定義を新たに作る
    ～

class Anime :             ←設計図「Anime」の定義がはじまる
  def __init__(self) :
    ～                    ←ここに初期処理を書く                    （最初に1回だけ実行）
     Pixelman(self)       ←ゲームのはじめからﾋﾟｸｾﾙﾏﾝを動かす（設計図「Pixelman」を
                                                                実体化させる）
  def update(self) :
    ～                    ←ここでキャラの座標の更新(移動)を呼び出す（1秒間に30回）
  def draw(self) :
    ～                    ←ここにキャラ（ﾋﾟｸｾﾙﾏﾝ）の描画を呼び出す（1秒間に30回）
Anime()                   ←設計図「Anime」を動かす（実体化させる）
```

ゲームプログラムの基本2(2/2)

設計図「Anime」の中のプログラムも修正するよ！

- 設計図「Pixelman」の中身は次のとおり、設計図「Anime」と似てるよ。

```
class Pixelman :      ←設計図「Bouningen」の定義を新たに作る

    def __init__(self) :
        ～                ←ここに初期処理を書く                （最初に1回だけ実行）

    def update(self) :
        ～                ←ここにキャラの座標の更新(移動)を書く
                            設計図「Anime」の中から「self.pixelman.update()」
                            で呼び出せる

    def draw(self) :
        ～                ←ここにキャラ（棒人間）の描画を書く
                             設計図「Anime」の中から「self. pixelman.draw()」
                            で呼び出せる
```

走る(勝手に右に)

説明を良く聞いて、右に走る仕組を理解しよう！

・走るために必要なのは、現在のx座標にスピードvxを足すことと、画面の右端からはみ出たら、画面の左端に戻すこと。
・そして、5ピクセル移動するごとに、キャラのパターンを切り替え、動きを出す。
（-1を掛けることで、ptn は 1→ -1 → 1 → -1 →・・・ に切り替わる）

```
class Pixelman :
  ～
    def update(self) :
        self.bn_x += self.bn_vx                    ←現在のx座標にスピードvxを足す
        if self.bn_x > Anime.SCREEN_WIDTH :←画面の右端からはみ出たら
            self.bn_x = 0
        if self.bn_x % 5 == 0 :                    ←5ピクセル移動するごとに、
            self.bn_ptn *= -1                        キャラのパターンを切り替える
```

飛ぶ(1/2)

プログラムを修正して、ジャンプする
仕組を作って行こう！

・スペースキーが押されたら、キャラのパターン(状態)であるptnを
　0 にし、ジャンプのモードに切り替える。

```
class Pixelman :
  ～
    def update(self) :
      ～
    if pyxel.btnp(pyxel.KEY_SPACE) == True :
        self.bn_ptn = 0
```

←スペースキーが押されたら

```
pyxel.btnp(pyxel.KEY_SPACE)
  スペースキーが押されたら True

pyxel.btn(pyxel.KEY_SPACE)
  スペースキーがおされている間 True

pyxel.btnr(pyxel.KEY_SPACE)
  スペースキーがはなされたら True
```

飛ぶ(2/2)

ジャンプは飛んだあと、だんだんスピードがおそくなって、０になったら、地面に落ちはじめるぞ！

・普段は勝手に右に移動するが、キャラのパターン(状態)であるptnが０の場合は、ジャンプ動作を行う。

```
class Pixelman :
    FS = -30          ←飛びはじめのスピード（上に移動するからマイナス）
    GA = 3            ←だんだんスピードが遅くなる（早くなる）量
 ～
    def update(self) :
        if self.bn_ptn == 0 :
            self.bn_y  += self.bn_vy      ←y座標にスピードを足す
            self.bn_vy += Pixelman.GA     ←スピードvyには3を足す
            if self.bn_y > Anime.SCREEN_HIGHT :←地面に戻ったら、ジャンプ
                self.bn_y = Anime.SCREEN_HIGHT - 16  の動作を止める
                self.bn_ptn = 1
                self.bn_vy = Pixelman.FS
```