

# なんとなくわからん オブジェクト指向 ～入門編～

JAVA勉強会

# おしながき

2/24

- オブジェクト指向って？
- オブジェクト指向って？ (個人的な解釈)
- クラスとオブジェクト
  - アイスブレイク 1
- 継承
- カプセル化
- インターフェースとポリモフィズム
  - アイスブレイク 2

# オブジェクト指向って？

3/24

- システムを管理しやすい単位(オブジェクト)で分割し、それらを組み合わせることでプログラムを構築する
  - 以前に作成したプログラムを再利用する
  - 大勢のエンジニアで共同開発を行う
  - プログラムの変更箇所をいち早く特定し対応する
  - あるプログラムの変更がほかのプログラムに影響しないようにする

# オブジェクト指向って？

4/24

## ■ なんとなくわからん

- そもそもオブジェクト指向のメリットだけ述べてオブジェクト指向の説明してない書籍が多い
- 「オブジェクトとは」 から入る書籍が多い

# オブジェクト指向って？ (個人的な解釈)

「そうだ。文明の利器を使おう。」

- すでにある道具(機能)を使って、効率よく目的を果たそう

・・・という考え方では？(´▽` ;)

# オブジェクト指向って？ (個人的な解釈)

6/24

```
カレー 作る () {  
    トマトを切る;  
    ピーマンを切る;  
    なすを切る;  
    たまねぎを切る;  
  
    お水をくんでくる;  
    燃えるものを拾ってくる;  
  
    火をおこす;  
    鍋を温める;  
  
    (中略)  
  
    お皿を用意する;  
    夏野菜カレー = お皿にカレーを入れる;  
    if(お皿が汚れている){  
        綺麗にする;  
    }  
    return 夏野菜カレー;  
}
```

# オブジェクト指向って？ (個人的な解釈)

6/24

```
カレー 作る () {  
    トマトを切る;  
    ピーマンを切る;  
    なすを切る;  
    たまねぎを切る;  
    お水をくんでくる;  
    燃えるものを拾ってくる;  
    火をおこす;  
    鍋を温める;  
    (中略)  
    お皿を用意する;  
    夏野菜カレー = お皿にカレーを入れる;  
    if(お皿が汚れている){  
        綺麗にする;  
    }  
    return 夏野菜カレー;  
}
```

```
カレー 作る () {  
    野菜[] = {トマト、ピーマン、なす、たまねぎ};  
    切る(野菜);  
    炒める(お肉、野菜);  
    煮込む(お肉、野菜);  
    夏野菜カレー = 盛り付け();  
    return 夏野菜カレー;  
}
```

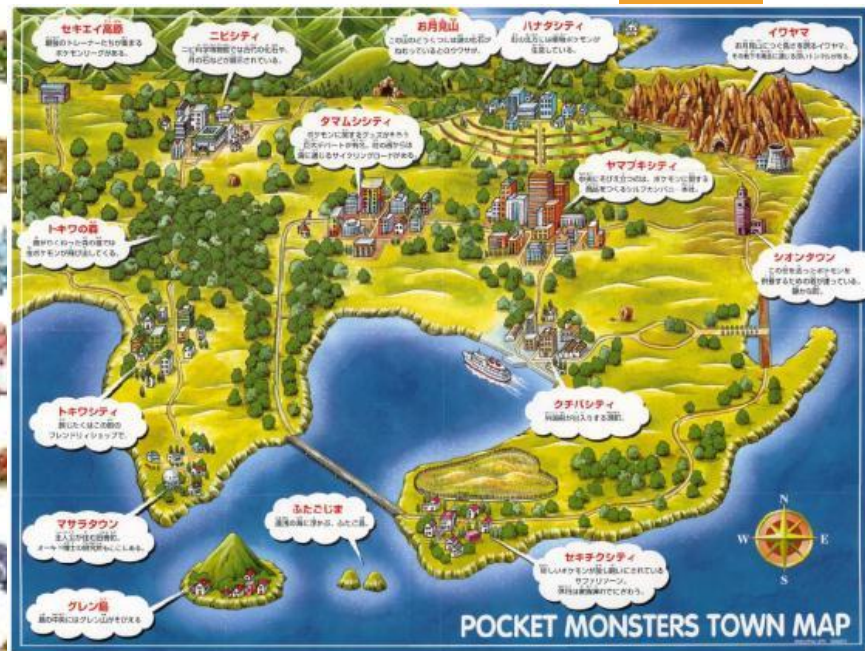
ルールが必要

# クラスとオブジェクト

7/24

- クラスはひな型、オブジェクトはモノ
  - 「データ」と「機能」を持っている
  - Ex: 犬、車、たいやき...
- 初心者には抽象的過ぎてピンとこない(´・ω・`)





# ポケットモンスター

9/24

旅をする



ポケモンと出会う



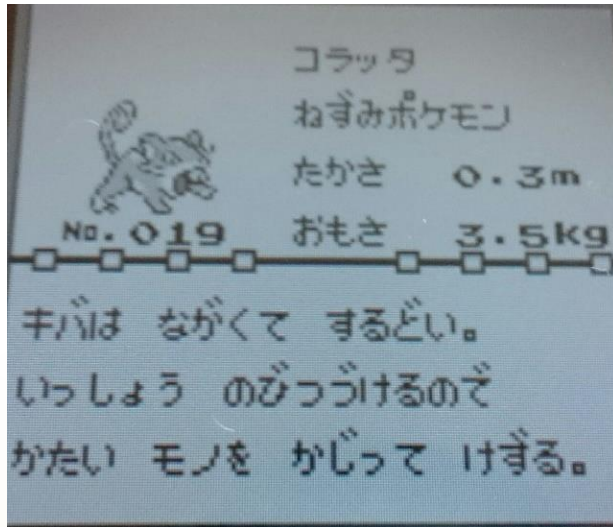
捕まえて仲間にする





# ポケットモンスター

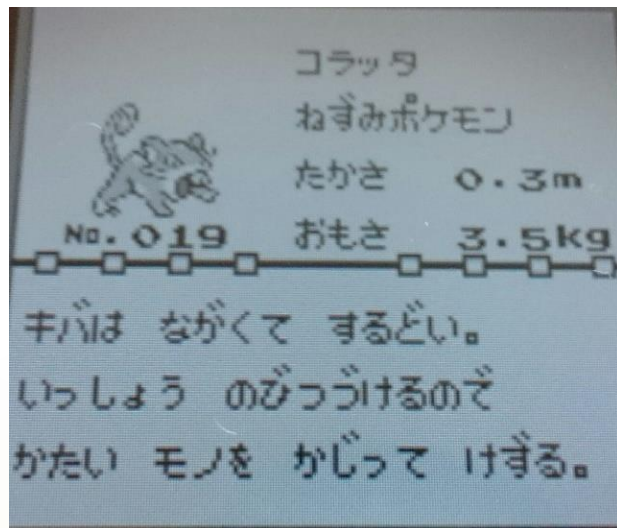
10/24



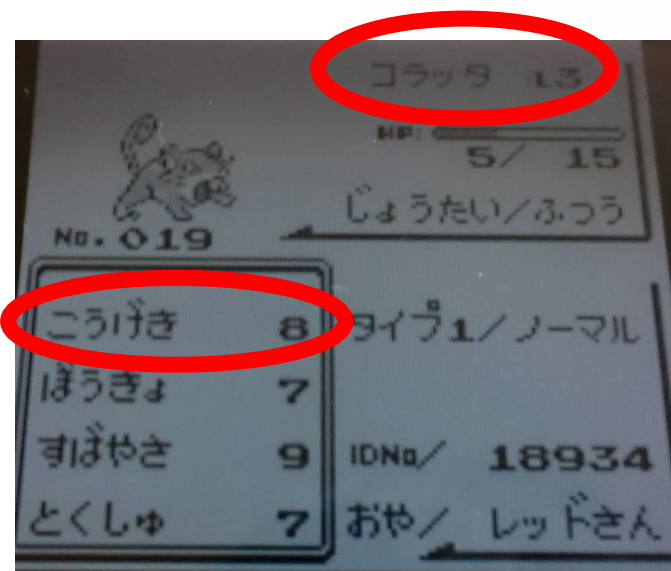
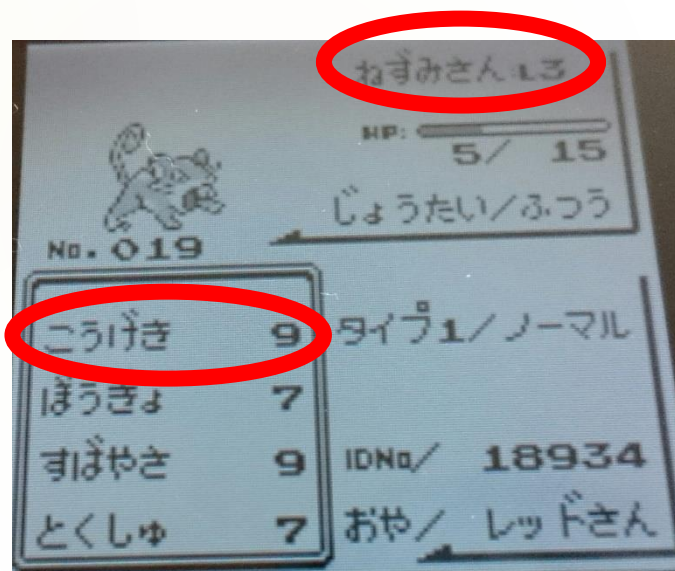
- 同じ種類のポケモンが何匹も出てくる

# ポケットモンスター

10/24



- 同じ種類のポケモンが何匹も出てくる
- 1匹1匹パラメータが違う
  - 1匹1匹つくるの大変では...?



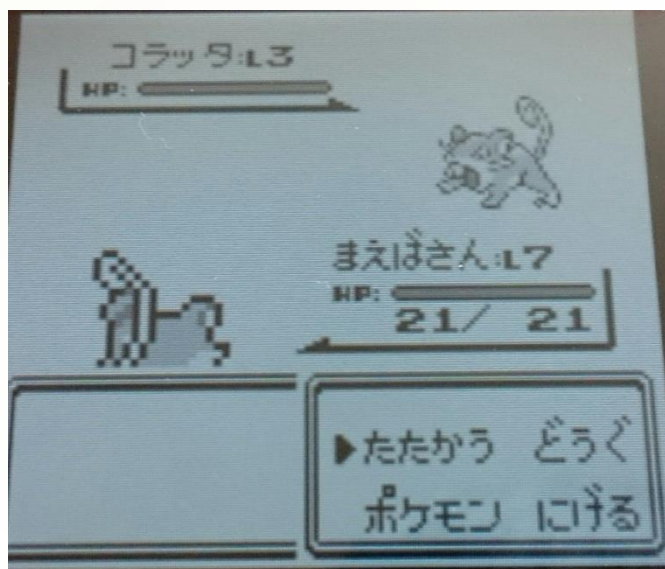
# ポケモン×オブジェクト指向

11/24



## ■ クラス名：コラッタ

- ニックネーム：－
- レベル：－
- こうげき：－
- ぼうぎょ：－



- たたかう
- どうぐを使う
- 他のポケモンと交代する
- 逃げる

# ポケモン×オブジェクト指向

12/24

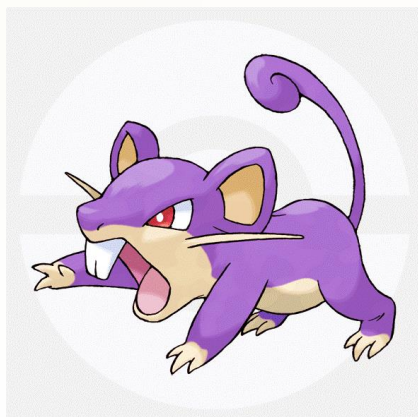
クラス：コラッタ



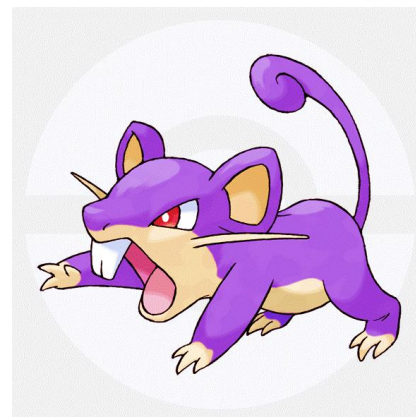
オブジェクト生成



オブジェクト生成



コラちゃん  
レベル：3  
こうげき：8  
ぼうぎょ：7

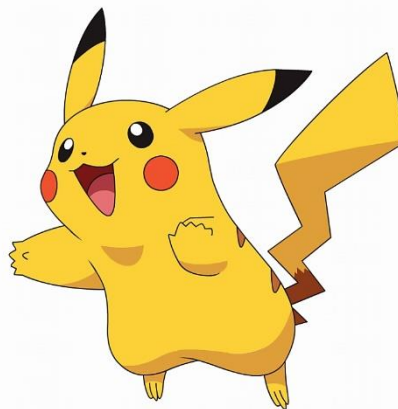


まえばさん  
レベル：4  
こうげき：12  
ぼうぎょ：9

# ポケモン×オブジェクト指向

13/24

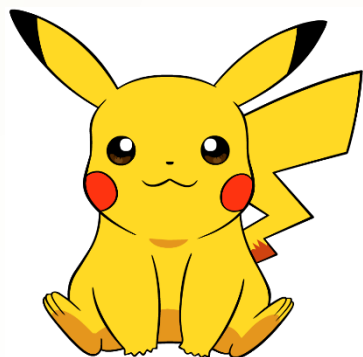
クラス：ピカチュウ



オブジェクト生成



オブジェクト生成



ピかちゃん  
レベル：4  
こうげき：1 1  
ぼうぎょ：9



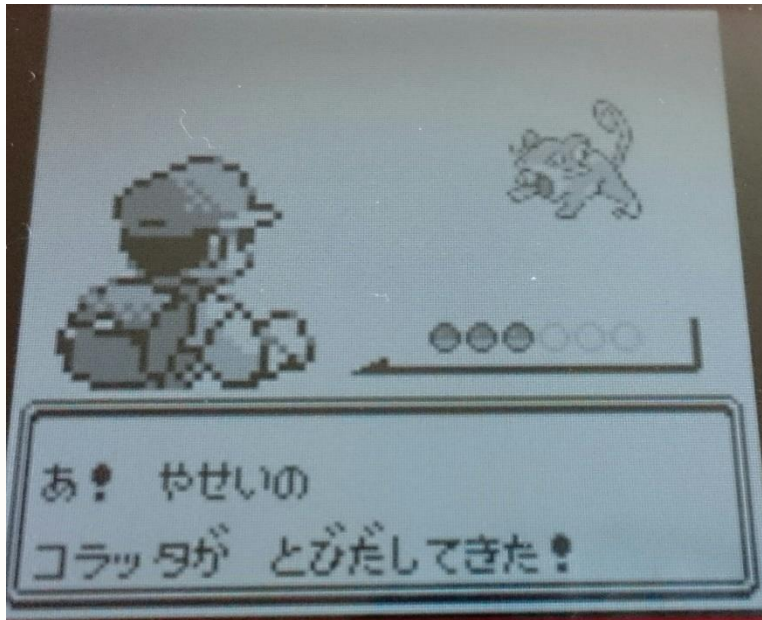
ピカピ  
レベル：7 9  
こうげき：6 5  
ぼうぎょ：4 8



# ポケモン×オブジェクト指向

14/24

## ■ オブジェクト生成





# ポケモンGO×オブジェクト指向

15/24



オブジェクト生成

オブジェクト生成



# ポケモンGO×オブジェクト指向

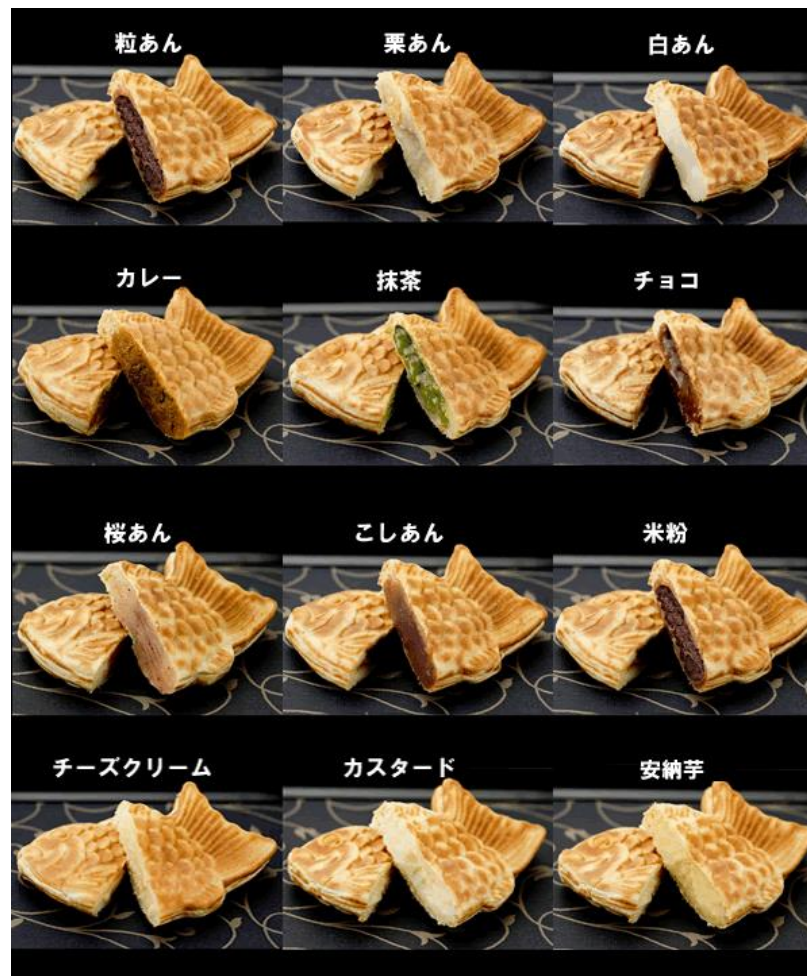
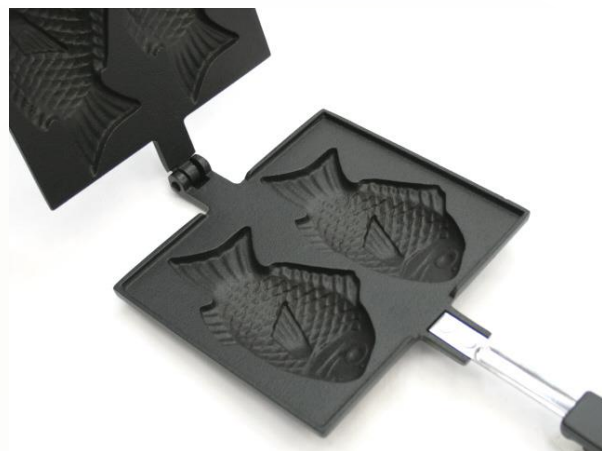
15/24

オブジェクト生成



# たいやき×オブジェクト指向

16/24



# 車×オブジェクト指向

17/24



軽自動車



ハッチバック



ミニバン



ステーションワゴン



SUV・クロカン



ハイブリッド



商用車・バン



福祉車両



オープン



クーペ



ピックアップトラック



キャンピングカー

# アイスブレイク 1

18/24

- クラスとオブジェクトの例を考えてみよう



# 継承

19/24

- 機能を引き継いだ新しいクラスを作る
  - 独自の機能を持たせて拡張する

クラス：ピカチュウ



継承



クラス：ライチュウ

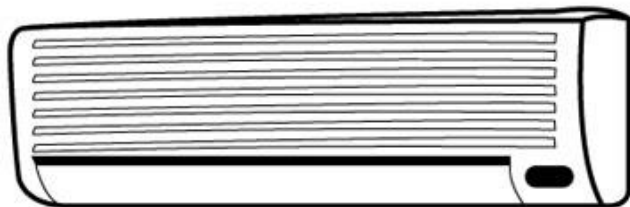
フラッシュ  
10万ボルト  
かみなり  
かみなりパンチ



- 機能を引き継いだ新しいクラスを作る
  - 独自の機能を持たせて拡張する

クラス：エアコン

冷風  
温風



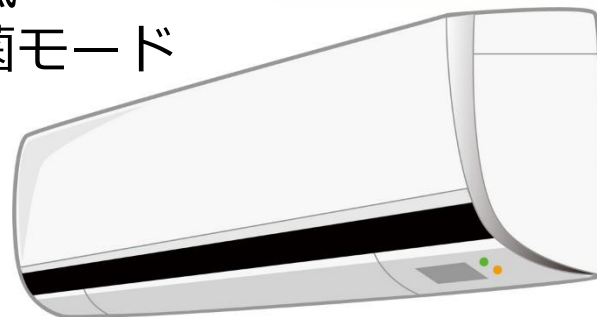
親クラス  
スーパークラス

継承



クラス：除菌機能付きエアコン

冷風  
温風  
除菌モード



子クラス  
サブクラス

# カプセル化

21/24

- データを機能経由で操作する(データと機能がセットになっている)
  - 内部構造を知る必要がない
  - 不整合な値が入らないようにチェックできる

- エアコン
  - 設定温度変更
- ポケモン
  - なまえをかえる





# インターフェースと ポリモフィズム

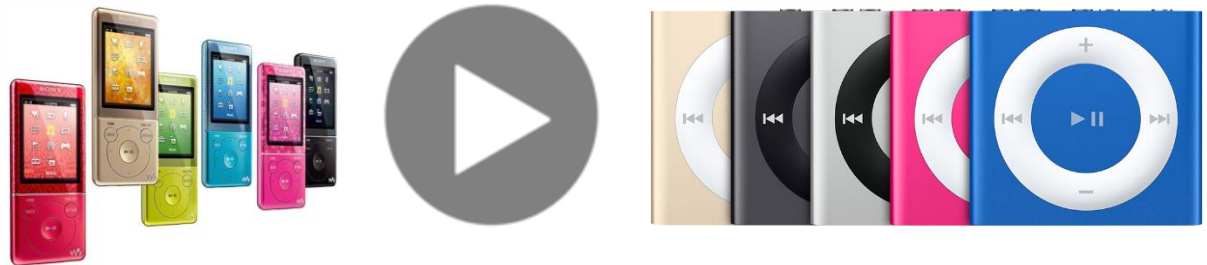
## ■ インターフェース

- 共通の操作方法の定義(クラスの仕様書みたいなもの)

## ■ ポリモフィズム

- 同じ操作方法でもオブジェクトごとに動作が異なること

### □ 再生ボタン



### □ なきごえ

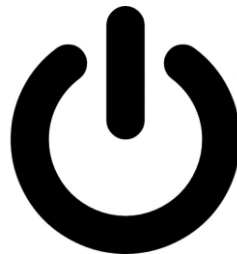


# アイスブレイク 2

23/24

- インターフェースとポリモフィズムの例を考えてみよう

解答例：電源ボタン



# オブジェクト指向って？ (再掲)

24/24

- システムを管理しやすい単位(オブジェクト)で分割し、それらを組み合わせることでプログラムを構築する
  - 以前に作成したプログラムを再利用する
  - 大勢のエンジニアで共同開発を行う
  - プログラムの変更箇所をいち早く特定し対応する
  - あるプログラムの変更がほかのプログラムに影響しないようにする