

プログラミングA2 第2回

練習問題解答例

練習問題

以下のprint文を実行すると以下のような出力となる

get_monsters.py

省略

```
print(monsters)
print(monsters[0])
```

オブジェクトが保持されている
メモリ上のアドレス

実行例

省略

```
[<pokemon.Pikachu object at 0x0000017030C90FD0>,
<pokemon.Pikachu object at 0x0000017030C90F70>]
<pokemon.Pikachu object at 0x0000017030C90FD0>
```

以下のような出力となるようにせよ

実行例

省略

```
[
Pikachu({'name': '野生のピカチュウその1', 'types': ['でんき'], '_Pikachu__level': -10, '__level': -20}),
Pikachu({'name': '野生のピカチュウその2', 'types': ['でんき'], '_Pikachu__level': 5})
]
Pikachu({'name': '野生のピカチュウその1', 'types': ['でんき'], '_Pikachu__level': -10, '__level': -20})
```

解答例

pokemon.py

```
class Pikachu:
    num = 0
    title = "ピカチュウ"

    def __init__(self, name, types):
        self.types = ["でんき"]

    def __repr__(self):
        return f"{self.__class__.__name__}({self.__dict__})"
```

例題：pikachu.py

野生のピカチュウとトレーナーに飼われているピカチュウがバトルする。レベルの大小を比較して、大きい方が勝利する。

①野生のピカチュウ(WildPikachu)クラスを定義する

pikachu.py

```
class WildPikachu():
```

```
    num = 0
```

```
    title = "野生のピカチュウ"
```

```
    def __init__(self, level):
```

```
        self.level = level
```

```
        self.name = f"{{__class__}.title}その{{__class__.num+1}}"
```

```
        __class__.num += 1
```

```
    def __str__(self):
```

```
        return f"{{self.name}}(Lv.:{{self.level}})"
```

```
    def __gt__(self, other):
```

```
        return self.level > other.level
```

野生には名前がないので、
numを使って通し番号を名前とする
例：「野生のピカチュウその2」

↓

文字列化

例：「野生のピカチュウその2(Lv.:89)」

↓

←比較演算のための特殊メソッド (greater_than)

練習問題：pikachu.py

②飼われたピカチュウ(TamePikachu)クラスを定義する

pikachu.py

```
class TamePikachu():
    def __init__(self, name, partner, level):
        self.name = name
        self.partner = partner
        self.level = level

    def __str__(self):
        return f"{self.partner}の{self.name}(Lv.:{self.level})"

    def __gt__(self, other):
        return self.level > other.level
```

練習問題：pikachu.py

③WildPikachuクラスとTamePikachuクラスの共通する部分を親クラス(Pikachuクラス)として定義する

pikachu.py

```
class Pikachu:
    title = "ピカチュウ"

    def __init__(self, level):
        self.level = level

    def __gt__(self, other):
        return self.level > other.level
```

練習問題：pikachu.py

④Pikachuクラスを継承して，WildPikachuクラスとTamePikachuクラスを定義しなす

pikachu.py

```
class WildPikachu(Pikachu):
```

```
    クラス変数 略
```

```
    def __init__(self, level):
```

```
        super().__init__(level)
```

```
        self.name = f"{{__class__}.title}その{{__class__}.num+1}"
```

```
        __class__.num += 1
```

```
    def __str__(self): そのまま
```

```
    def __gt__(self, other): 削除
```

```
class TamePikachu(Pikachu):
```

```
    def __init__(self, name, partner, level):
```

```
        self.name = name
```

```
        self.partner = partner
```

```
        super().__init__(level)
```

```
    def __str__(self): そのまま
```

```
    def __gt__(self, other): 削除
```