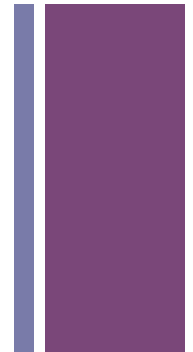


# プログラミング基礎

プログラミング言語Java (2)  
メソッド・制御構造 (繰り返し)

# + 今日の学習目標

- メソッドでのデータの受け渡しの基本を学ぶ
- データの型について学ぶ
- 簡単な制御構造を学ぶ（繰り返し）



## + 猫の鳴く回数を指定してみる

- 前回の猫オブジェクト（Cat.java）を以下のように変更

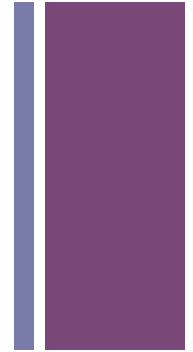
```
class Cat {  
    void cry(int num) {  
        int i = 0;  
        while( i < num ) {  
            System.out.println("Mew!!");  
            i = i + 1;  
        }  
    }  
}
```

# + 猫オブジェクトを呼び出す側も変更

- Main.java を以下のように変更
- 前回と同じように実行してみる
  - 画面に「Mew!!」と10回表示されたら成功

```
class Main {  
    public static void main(String[] args) {  
        Cat mike = new Cat();  
        mike.cry(10);  
    }  
}
```

# + 型



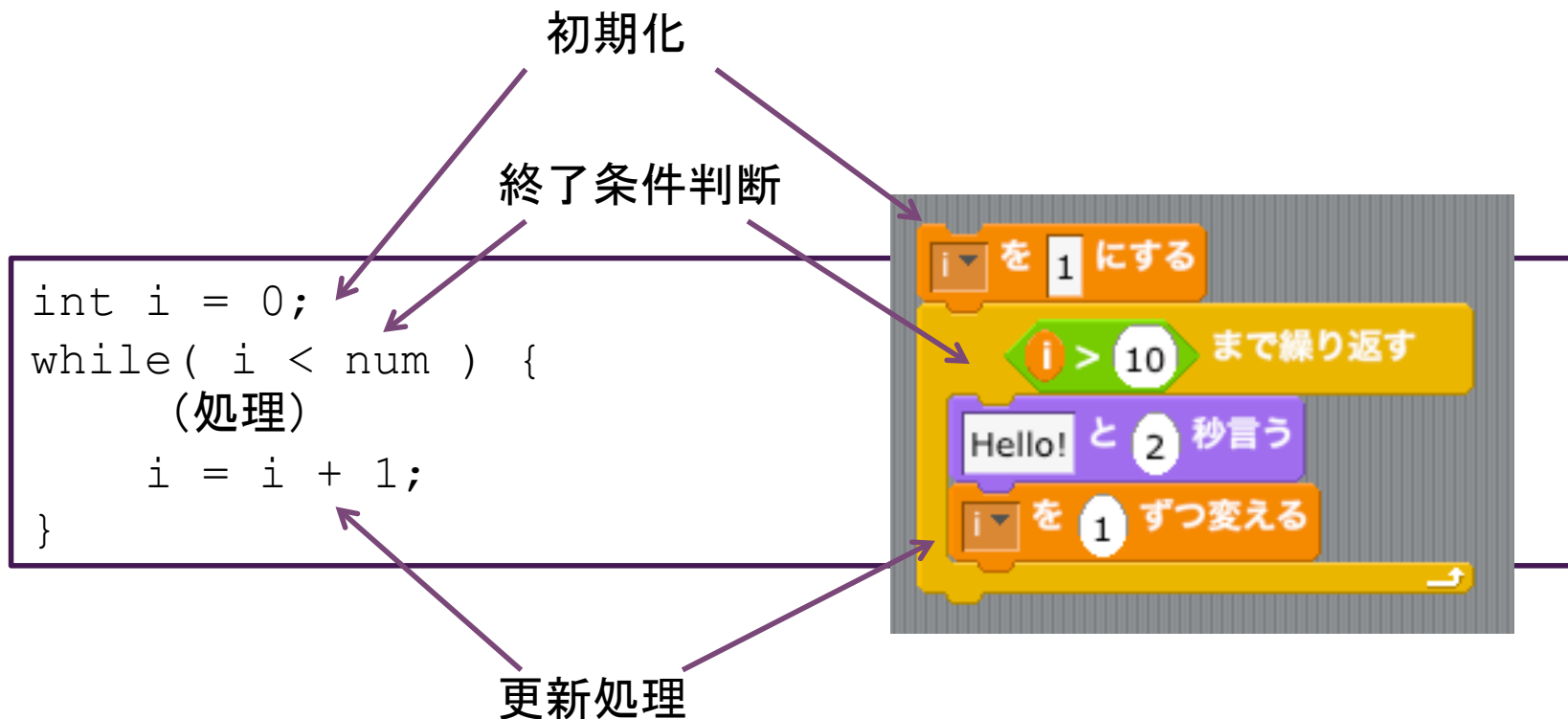
- Javaでは全てのデータに“型”がある
- 変数はどの型のデータを入れられるか決まっている  
→ 変数の型
- 変数の型は変数を最初に宣言するときに決める  
→ 猫オブジェクトでは“int i”とか“int num”とかの部分
- 代表的な型
  - int（整数） float（実数） double（倍精度実数）
  - boolean（真・偽の2値） char（1文字）
  - オブジェクト型（インスタンスを入れるために使う）

# + 変数の宣言

- 変数の宣言 = 変数の準備 (Scratchで言う と 変数の作成)
  - 変数を使うまでに宣言しておく必要がある
- 変数の宣言
  - 【書式】 型 変数名
  - 【例】 `int a` とか `double x` とか
- 変数の宣言と同時に初期化 (最初にデータを入れておく)
  - 【書式】 型 変数名 = 初期値
  - 【例】 `int a = 0` とか `double pi = 3.14` とか

# + Javaにおける繰り返し(while命令)

- Scratchでの定番の繰り返しとの類似性に注目
  - ただし条件は「逆」になっていることに注意 (while=～の間)



# + Javaにおける繰り返し(for命令)

- for命令＝初期化＋条件判断＋更新処理を1行で書ける

```
for(int i = 0 ; i < num ; i = i + 1) {  
    (処理)  
}
```



# + for命令で猫を書き換えてみる

## ■ ちょっとすっきり

```
class Cat {  
    void cry(int num) {  
        for(int i = 0 ; i < num ; i = i + 1 ) {  
            System.out.println("Mew!!");  
        }  
    }  
}
```

# + メソッドでのデータ受け渡し

- 「これで仕事をしてくれ」の「これ」の部分

```
class Main {  
    public static void main(String[] args) {  
        Cat mike = new Cat();  
        mike.cry(10);  
    }  
}
```

```
class Cat {  
    void cry(int num) {  
        for(int i = 0 ; i < num ; i = i + 1 ) {  
            System.out.println("Mew!!");  
        }  
    }  
}
```