



# 独習Java

---

- 4. 1 ステートメントのネスト
- 4. 2 forループのバリエーション
- 4. 3 whileループ



# Ifステートメントのネスト

---

## ネストとは？

あるifステートメントが別のif又はelseの評価対象になっていること。

「あるifステートメントが外側のif内にネストされている」と言う。

## 例

```
if(count > max)
```

```
if(error) System.out.println("Error");
```



# if-else-ifラダー

---

- Elseは関連つけられているelseがない、同じブロック内の最も近いifに関連付けられる。

- 例

```
if(expression) statement;  
else  
    if(expression) statement;  
else  
    if(expression) statement  
    .  
    .  
    .  
else statement;
```



# forループのバリエーション

---

- forループの初期化、テスト、インクリメントの各部分の役割は狭く限定されていません。
- 3つの条件は必ずしも指定しなくても良い
- forの内側に空の式があってもよい



# forの初期化がない例

---

```
Class ExpressionLoop {  
    public static void main(String args[]){  
        int i = Integer.parseInt(args[0]); //初期化  
        for( ; i > 0; i--) System.out.print(i + " ");  
        System.out.println(" ");  
    }  
}
```

引数5を与えた場合

出力 5 4 3 2 1



# forによる無限ループ

---

- for内に一つも式がない場合はコンパイラは真であると見なす。よってループは続く。

例

```
for( ; ; ) {  
    .  
    .  
    .  
}
```



# whileループ

---

- Whileステートメント

While (expression) statement;

- 式の値はループの最初にチェックされ、式が真である限りその対象の評価を繰り返す。
- 最初の式が偽であった場合はループは二度と実行されない。



# Whileループの例

---

```
Class WhileDemo {  
    public static void main(String args[]){  
        // i が0以上のループ  
        int i= Integer.parseInt(args[0]); //引数をint型に変換  
        While (i > 0) {  
            System.out.print(i + " ");  
            i--;  
        }  
        System.out.println(" ");  
    }  
}
```

引数が5の場合の出力      5 4 3 2 1



# 宿題

---

- forループを使って1から100までの奇数を標準出力するプログラムを作成せよ。ただしインクリメントの部分は空白とし、for文の中でインクリメントせよ。