

# 第4章解説

繰り返し文と繰り返し制御文

# 目次

- while文
- do-while文
- for文
- 拡張for文
- break文
- continue文
- ラベル

# while文

```
while (条件式) {  
    処理文 ;  
}
```

- boolean
- trueの時、処理文を実行

# while文

```
while (true) {  
    System.out.println("☆");  
}
```



```
while (false) {  
    System.out.println("☆");  
}
```



# do-while文

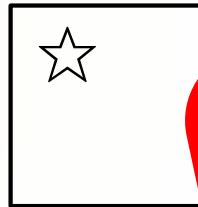
```
do {  
    処理文 ;  
} while (条件式) ;
```

①処理文実行

②条件判定

# do-while文

```
int num = -1;  
do {  
    System.out.println("☆");  
} while (num > 0);
```



**何が何でも  
必ず1回は  
処理文を実行**

# for文

一番最初に  
一回だけ実行

- ・ 処理文実行前に毎回判定
- ・ trueの時、処理文を実行

```
for (前処理 ; 条件式 ; 後処理) {  
    処理文 ;  
}
```

処理文実行後に  
毎回実行

## 前処理、後処理の制約(詳しくはp.125)

- ・ "式"を入れる必要がある
- ・ カンマ(,)で区切れば、2つの式を書ける
- ・ 前処理での宣言式は1つのみ

# for文

```
for ( ;; ) {  
    System.out.println("☆");  
}
```



セミコロン(;) 2つさえ  
書いておけば、  
中身省略しても実行可能



うえーん(;;)

23:39

どしたの?

23:40



うえーん(;;)

無限ループに  
見える病



# 拡張for文

```
for (変数宣言 : 参照変数名) {  
    処理文 ;  
}
```

# 拡張for文

```
char[ ] abc = {'a', 'b', 'c', 'd', 'e'};
```

コロン!

```
for (char moji : abc) {  
    System.out.print(moji);  
}
```

既に用意されてる配列や  
コレクション(ArrayList)  
の名前

for文の中で使う  
新しい変数

※必ず同じ型

abcde

# break文

処理 ;



```
while (条件式) {  
    処理 ;  
    break ;  
    処理 ;  
}
```



処理 ;

繰り返し文が  
スキップされる  
(繰り返し文から抜ける)

# continue文

処理 ;



```
while (条件式) {
```

```
  処理 ;
```

```
  continue ;
```

```
  処理 ;
```

```
}
```



処理 ;

繰り返し文中の  
continueより後ろの処理  
がスキップされる  
(繰り返し文は続行)

# 繰り返し制御文

```
for (int i = 0; i < 5; i++) {  
    if (i == 2) {  
        System.out.print("★");  
        break ;  
    }  
    System.out.print("☆");  
}
```



☆ ☆ ★

```
for (int i = 0; i < 5; i++) {  
    if (i == 2) {  
        System.out.print("★");  
        continue ;  
    }  
    System.out.print("☆");  
}
```



☆ ☆ ★ ☆ ☆

# ラベル

```
while (条件式) {
```

```
while (条件式) {
```

```
    while (条件式) {  
        break ;  
    }
```

```
}
```

```
}
```

抜け出せるのは  
1つだけ！

# ラベル

```
loop1: コロン!  
while (条件式) {  
    loop2: コロン!  
    while (条件式) {  
        while (条件式) {  
            break ;  
        }  
    }  
}  
}
```

# ラベル

```
loop1:  
while (条件式) {
```

```
    loop2:
```

```
        while (条件式) {  
            while (条件式) {  
                break loop2 ;  
            }  
        }  
    }
```

```
}
```



# ラベル

```
loop1:  
while (条件式) {  
    loop2:  
    while (条件式) {  
        while (条件式) {  
            break loop1 ;  
        }  
    }  
}  
}
```

# 目次(再掲)

- while文
- do-while文
- for文
- 拡張for文
- break文
- continue文
- ラベル

**おわり**