



独習Javaゼミ

4.4: doループ

4.5: ループのネスト

4.6: breakステートメント

4.7: continueステートメント



doループ(I)

- ループにdoステートメントを使う方法

```
do {  
    statements  
} while (expression);
```

- ・繰り返されるステートメントが1つだけの場合には中括弧は必要ない。
(doを終了するwhileは、doループの一部であり、whileループの開始を意味するものではないので、通常は中括弧を使うのが望まれる。)



doループ(Ⅱ)

- doループの特徴

- 式が真である限りステートメントを繰り返す。
- 式が偽になると停止。
- ループの内側のコードを少なくとも一回実行する。
(ループを制御する式が、ループの最後でテストされるため。)



doループ (Ⅲ)

- 「例」. . . doループを使用して14個の素数を作成。

```
class PrimeNumbers{
    public static void main(String args[]){
        int count = 0;
        int number = 2;

        do {
            //要素があるかテストする
            boolean prime = true;
            for(int i = 2;i < number / 2 + 1;i++){
                if((number % i) == 0) prime = false;
            }
            //素数がある場合は、表示する
            if(prime == true){
                System.out.print(number + " ");
                ++count;
            }
            ++number;
            //数値を増分する
        }while (count < 14);
        System.out.println("");
    }
}
```



doループ(Ⅲ)

- 実行結果

2 3 5 7 11 13 17 19 23 29 31 37 41 43



ループのネスト(I)

- 「解説」

- ・あるループ本体に別のループが入っている場合、「**2つ目のループは1つ目のループの内側にネストされている**」と言う。

- 「例①」. . . 1~10までの数値を画面上に10回出力するリスト。

```
for(i = 0; i < 10; i++){  
    for(j = 1; j < 11; j++)        //ネストされたループ  
        System.out.print(j + " ");  
        System.out.println("");  
}
```



ループのネスト(Ⅱ)

- 「例②」. . . 3つのforループを使用してアルファベットを3回出力し、各文ごとに2回ずつ出力。

- ```
class Alphabet3 {
```
- ```
public static void main(String args[]){
```
- ```
int i,j,k;
```
- ```
for(i = 0;i <3;i++) //3回全体を表示するためのループ
```
- ```
for(j = 0;j < 26;j++) //A~Zまで表示のためのループ
```
- ```
for(k = 0;k < 2;k++) //2回表示のためのループ
```
- ```
System.out.print((char)('A' + j)); //Aの文字コードに加算してZまで表示
```
- ```
System.out.print();
```
- ```
}
```
- ```
}
```



ループのネスト(Ⅲ)

- 実行結果

```
AABBCCDDEEFFGGHHIIJJKKLLMMNNO  
OPPQQRRSSTTUUVVWWXXYYZZAABBC  
CDDEEFFGGHHIIJJKKLLMMNNOOPPQQ  
RRSSTTUUVVWWXXYYZZAABBCCDDEE  
FFGGHHIIJJKKLLMMNNOOPPQQRRSSTT  
UUVVWWXXYYZZ
```



breakステートメント(I)

- 「特徴」

- ・ 本体内の任意のポイントからループを出て、通常の終了式を飛ばす事が可能。
- ・ ループの内側でbreak式が検出されると、そのループは直ちに停止し、プログラム制御はループの次のステートメントから再開される。



breakステートメント(Ⅱ)

- 「例①」. . . 1～10の数値の出力。

- ```
class BreakDemo {
```
- ```
    public static void main(String args[]){
```
- ```
 int i;
```
- ```
        for(i =1;i < 100;i++){
```
- ```
 System.outprintln(i); //10になったらループから抜ける
```
- ```
            if(i == 10) break;
```
- ```
 }
```
- ```
    }
```
- ```
}
```



# breakステートメント(Ⅲ)

---

- 実行結果

1

2

3

4

5

6

7

8

9

10



# breakステートメント(IV)

---

- 「例②」. . . コマンドライン引数の中で、文字 '\$' から始まる文字列を検索。

- ```
class DollarArgument {
```
- ```
public static void main(String args[]){
```
- ```
for(int i = 0;i < args.length;i++)
```
- ```
if(args[i].startsWith("$")){ // $から始まるかどうかを検査
```
- ```
    System.out.println(args[i]);
```
- ```
 break; //forループから抜ける
```
- ```
}
```
- ```
}
```
- ```
}
```
- ```
}
```



# breakステートメント (V)

---

- 実行結果

- 例えば (a b \$c \$d) を指定したら以下のようにになる。

\$c



# continueステートメント ( I )

---

- 特徴

- continueはbreakと正反対の動作をするステートメント。
- continueステートメントは、ループの次の繰り返しを強制的に発生させ、continueステートメントとループのステートメント条件との間の全てのコードをスキップする。



# continueステートメント(Ⅱ)

---

- 「例①」. . . ループの最後の5回の繰り返しに対する出力。

```
class ContinueDemo{
 public static void main(String args[]){
 for(int x = 0;x < 100;x++){
 if(x < 95)
 continue; //95以下ならループに戻る
 System.out.println(x);
 }
 }
}
```



# continueステートメント(Ⅲ)

---

- 実行結果

95

96

97

98

99



# continueステートメント(IV)

---

## ● 仕組み

- continueステートメントに達する度に、ループは繰り返され、println()ステートメントは飛ばされる。
- whileループとdo-whileループでcontinueステートメントを使用すると、制御はテスト条件に直接移動し、ループプロセスは続行する。
- forの場合には、ループのインクリメント部分が実行され、条件テストが実行されて、ループが続行する。



# continueステートメント (V)

- 「例②」. . . 0.2~0.8間の乱数が表示され、breakステートメントによりwhileループは停止。

- ```
class TenRandomNumbers{
public static void main(String args[]){
int i = 0;
while (true){
double d = Math.random();
if(d > .2 || d > .8)           //0.2~0.8の範囲にない場合は無視
continue;
System.out.println(d);       //10個表示したらループを抜ける
if(++i == 10)
break;
}
}
}
```



continueステートメント (VI)

- 実行結果 (例)

0.740385....

0.369343....

0.568518....

0.454643....

0.218156....



宿題

- 1～100の間の奇数だけ出力するプログラムを以下の構文を使用し作成しなさい。またcontinue文を使用しなさい。

```
for(i = 1; i < 101; i++)....
```