


01 型&式



Created by GT F

Last updated about 2 hours ago

- Javaの基本型
- 変数
 - 変数の宣言と値の代入
 - 既に宣言された変数への値の代入
 - 定数
- 式（しき）
 - 代入
 - 算術演算子（えんざんし）
 - 関係演算子
 - 論理演算子
 - 条件論理演算子
- 質問集

Javaの基本型

| 型 | 日本語読み方 | 値の種類 |
|---------|--------|----------------------|
| byte | バイト | バイト型 |
| short | ショット | 整数型 |
| int | | 整数型 |
| long | ラング | 整数型 |
| float | | 浮動小数点数 |
| double | ダブル | 浮動小数点数（ふどうしょうすうてんすう） |
| char | | 省略（現時点特に覚えなくてもいいです） |
| boolean | ブリアン | true また false |

現時点値の範囲を覚える必要がありません。以下 3 型を覚えてください：`byte`、`int`、`float`、`char`、`boolean`。

変数

1. インスタンス変数
2. クラス変数（静的変数）
3. ローカル変数
4. パラメータ

Javaの仕様では、変数という言葉はローカル変数、インスタンス変数などを意味しますが、本章はローカル変数について説明します。なお、これ以後で**変数**と書かれていれば、全てローカル変数を意味します。

変数の宣言と値の代入

データ型の直後に変数名を続きます。変数名は**半角英字**か**下線**(_)から始まり、後に半角英数字に書かれてる文字列の組み合わせが続きます。

例1. 変数の宣言：

```
1 boolean result = true;
2 char capitalC = 'C';
3 byte b = 100;
4 short s = 10000;
5 int i; // 初期値を設定しない
```

例2. 変数名：

```
1 int _a; // OK
2 int $a; // OK,あまりよくない。商用コードに書かないで
3 int 10a; // NG,頭文字は数字
4 int Abc; // NG,ローカル変数の頭文字は小文字で（ただし、プログラミング上問題ない）
5 int _a; // NG,ローカル変数は既に定義されています。（名称重複）
```

変数を宣言した後、初期値を設定可能です。その後の例1は2種類の変数宣言の形を示しています。最初の4つの変数は初期値を設定する初期化を含んでいますが、最後の変数では初期化子は省略可能であることを示しています。この場合、変数をすぐ使えない。（ローカル変数を初期化しないと利用できませんが、インスタンス変数の場合、初期化子を省略する場合、デフォルト値を自動的に設定されています）。

既に宣言された変数への値の代入

初期値を持っているかどうかに関わらず、変数に新しい値を代入できることを示します。

例1. 変数の値を再設定

```
1 int a = 10; // a = 10
2 a = 100;    // 現在 変数aの値は100
```

これらの変数は静的な型を持っているので、誤った型のデータを代入しようとしても、コンパイルエラーが発生します。

例2. 誤まった型の値を代入

```
1 int a = 10; // 変数aの型はint
2 a = 10.0f; // Float型の値を代入しよう
```

質問：

- 1. 変数の宣言方法 ⇒ 型 変数名;
- 2. 変数の代入 ⇒ 変数名 = 値;
- 3. 変数の命名規則
- 4. 変数は重複不可。
- 5. 変数の値を再代入型は宣言時の型は同じ。

定数

final static 修飾子による定数の定義する。定数は必ず代入演算子を使って明示的に初期化を行います。

式（しき）

代入

変数に値を割り当てるには代入文 = を使用します。変数を利用する為に宣言しておかなれなりません。

```
1 int a = (1 + 2) * 3;
```

= の左辺の変数 a に右辺の式 (1 + 2) * 3 の計算結果を値に代入されます。文の終わりにはセミコロン ; で示します。

算術演算子（えんざんし）

| 演算子 | 説明 | 実例 | int a = 10; int b = 20 の場合 |
|-----|-----------|-------|----------------------------|
| + | 加算 | a + b | 20 |
| - | 減算 | a - b | -10 |
| * | 乗算 | a * b | 200 |
| / | 除算 | a / b | 0 (精度) |
| % | 剰余（MOD） | a % b | 10 |
| ++ | 前置インクリメント | ++x | 11 |
| | 後置インクリメント | x++ | 10 |

| | | | |
|----|----------|-----|----|
| -- | 前置デクリメント | --x | -9 |
| | 後置デクリメント | x-- | 10 |

※グレースアウト箇所は後で理解してもおkです。

- 整数 + 整数 = 整数;
- 整数 + Float = float;
- 数値型（算術演算子） = 数値型

以下コードを実施して、算術演算子を理解してください。

```

1  int a = 10;
2  int b = 20;
3
4  System.out.println(a + b); // 30
5  System.out.println(a - b); // -10
6  System.out.println(a * b); // 200
7  System.out.println(a / b); // 0
8  System.out.println(a % b); // 10
9
10 int x = 0;
11 System.out.println(++x); // 1
12 x = 0;
13 System.out.println(x++); // 0
14
15 int y = 0;
16 System.out.println(--y); // -1
17 y = 0;
18 System.out.println(++y); // 1

```

関係演算子

関係演算子はすべての数値型をサポートしており、結果をbooleanで返します（`boolean x = 10 > 20`）。

| 演算子 | 説明 | int a = 10; int b = 20 の場合 |
|-----|-----------|----------------------------|
| == | 等しい（ひとしい） | a == b → false |
| != | 等しくない | a != b → false |
| > | より大きい | a > b → false |
| >= | 以上 | a >= b → false |
| < | より小さい | a < b → true |

| | | |
|----|----|---------------|
| <= | 以下 | a <= b → true |
|----|----|---------------|

以下コードを実施して、関係演算子を理解してください。

```
1 int a = 10;
2 int b = 20;
3 boolean c = a > b;
4 System.out.println(a > b);
5 System.out.println(a >= b);
6 System.out.println(a == b);
7 System.out.println(a != b);
8 System.out.println(a < b);
9 System.out.println(a <= b);
```

論理演算子

| 演算子 | 説明 | 例 | |
|-----|-----------|-------------|----|
| | 条件論理和 | a b c | また |
| && | 条件論理積（せき） | a && b | かつ |
| ! | 論理否定 | !a | |

```
1 public static void main(String[] args) {
2     // 変数 代入 初期値
3     int a = 10;
4     int b = 20;
5     if (a < b || b == a) {
6         // true ? false ?
7     }
8     if (a < b && b == a) {
9         // true ? false ?
10    }
11    if (!(a > b)) {
12        // ここに到着できますか
13    }
14 }
```

条件論理演算子

| 演算子 | 説明 | 例 |
|-------------------------|--|-------------------------|
| 条件 ? trueならば : falseならば | ? 記号の前の式を評価する、true の場合、: の式を返す、false の場合、: を返し | int max = x > y ? x : y |

質問集

質問 1 : 以下変数nの値は？

```
1  int i = 10;  
2  int n = i%5;
```

質問 2 : 右の式を解釈してください: `a = a > 10 ? 11 : 0;`

質問 3 : 変数値を比較する演算子は `=` or `==` ?

質問 4 : ブリアン (`boolean`) 変数に設定可能な値は「?」 「?」。

 Like Be the first to like this

No labels 