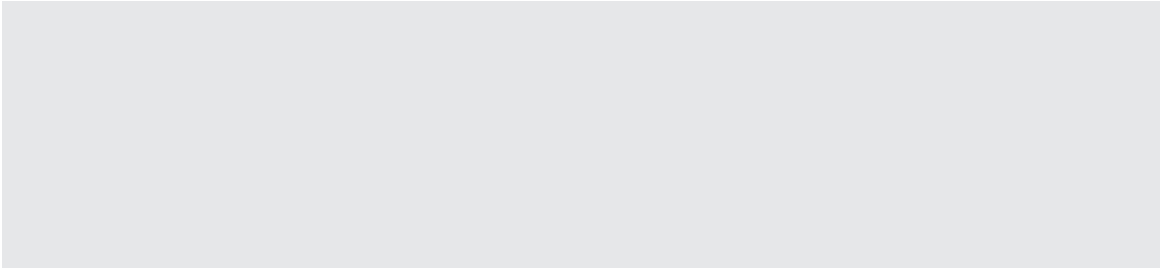


Java プログラムを書く準備をする

```
public class クラス名 {  
    public static void main(String[] args) {  
        // ▼ここから行いたいプログラムの処理を書きます。  
  
          
  
        // ▲ここまで  
    }  
}
```

Java プログラムの書くには規定の書式が必要となるので、
上記ソースコードをひな形として準備しましょう。

Javaプログラムの実行手順

拡張子の
有無に注意

```
javac ファイル名.java  
java ファイル名
```

Step1. コンパイル (javac ファイル名.java)

Java プログラム実行するには必ず実行前にコンパイルを行う必要があります。

コンパイルするには「javac」コマンドを使用します。

※コンパイルが出来ない場合はエラー文が表示されます。

Step2. 実行 (java ファイル名)

コンパイルが無事完了すれば、

次は「java」コマンドを使用してJavaプログラムを実行します。

画面に文字を表示する

```
System.out.println(#####);
```

任意の文字列を表示する時は "" で表示したい文字を囲んで表記します。

```
System.out.println("Hello");
```

→ Hello

計算結果を表示する時は計算式をそのまま表記します。

```
System.out.println((2+5*2)/2);
```

→ 6

文字列と計算結果を表示するときは、計算式を括弧 () で囲み、文字列と計算式を + で繋ぎます。

```
System.out.println("問題：10×2= 答え：" + (10*2));
```

→ 問題：10×2= 答え：20

変数の使い方（手順）

宣言（初期化） → 代入 → 参照

1. まず変数の宣言を行います。（同時に初期値を代入する事を初期化といいます）

<code>int i;</code>	<code>int i = 0;</code>
<code>String str;</code>	<code>String str = "";</code>

2. 「変数名 = 値」の書式で右辺の値を左辺の変数名に代入します。

```
i = 10;  
String str = "文字列";
```

3. 変数の値を使用する時は、変数名を使用したい場所に表記します。

```
System.out.println(i+5);  
System.out.println(str);
```

変数の種類

基本型は全部で 8 種類

☆ boolean 型	論理型 (true または false)
char 型	整数型 (文字型) (0~65535)
byte 型	整数型 (-128~127)
short 型	整数型 (-32768~32767)
☆ int 型	整数型 (-2147483648~2147483647)
◎ long 型	整数型 (-9223372036854775808~9223372036854775807)
◎ float 型	単精度浮動小数点型 (有効桁数 6 桁)
☆ double 型	倍精度浮動小数点型 (有効桁数 15 桁)

☆印はよく使う型なので優先して覚えましょう。

基本型ではありませんが、文字列を扱う型として String 型 (クラス) があります。

キーボードからの文字入力を取得する方法

```
変数名 (String 型) = reader.readLine();
```

キーボードからの文字入力を取得するには
BufferedReader クラスの `readLine()` メソッドを使用します。

(使用例)

```
line = reader.readLine();
```

「line」という String 型の変数にキーボード入力された値（文字）が代入されます。

`readLine()` メソッドを使用する為には、いろいろ手続きが必要になるので、
なれるまでは次ページのひな形ソースを利用しましょう。

キーボードからの文字入力を取得する時の ひな形ソース

```
import java.io.*;

public class クラス名 {
    public static void main(String[] args) {
        BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
        try {
            // ▼ここから行いたいプログラムの処理を書きます。

            // ▲ここまで

        } catch (IOException e) {
            System.out.println(e);
        }
    }
}
```

キーボードから入力された値を数値に変換する方法

変数名 (int 型) = Integer.parseInt(line);

`reader.readLine();` で取得した値を

計算などで使用するときには、一度 int 型に変換します。

文字列を数値に変換するには Integer クラスの `parseInt()` メソッドを使用します。

(使用例)

```
line = reader.readLine();
```

```
num = Integer.parseInt(line);
```

「num」という int 型の変数に、数値変換された値が代入されます。

他に、小数点を含む数値に変換する `Double.parseDouble();` というメソッドもあります。

演算の時の型変換ルール

- (1) どちらかの値が `double` 型の場合は、もう片方の値を `double` 型に変換して計算する。
- (2) どちらかの値が `float` 型の場合は、もう片方の値を `float` 型に変換して計算する。
- (3) どちらかの値が `long` 型の場合は、もう片方の値を `long` 型に変換して計算する。
- (4) (1) から (3) に該当しない場合は両方の値を `int` 型に変換して計算する。

型変換は上記の (1) ~ (4) の規則に従います。

この規則は (1) から順に適用され、いずれかに適用されればその後の変換は行われません。

(例)

- (1) `double` 型 = `double` 型 + `long` 型 (`double` 型に変換) ;
- (2) `float` 型 = `float` 型 + `byte` 型 (`float` 型に変換) ;
- (3) `long` 型 = `long` 型 + `int` 型 (`long` 型に変換) ;
- (4) `int` 型 = `short` 型 (`int` 型に変換) + `short` 型 (`int` 型に変換) ;