



# プログラム演習

- Java編 -

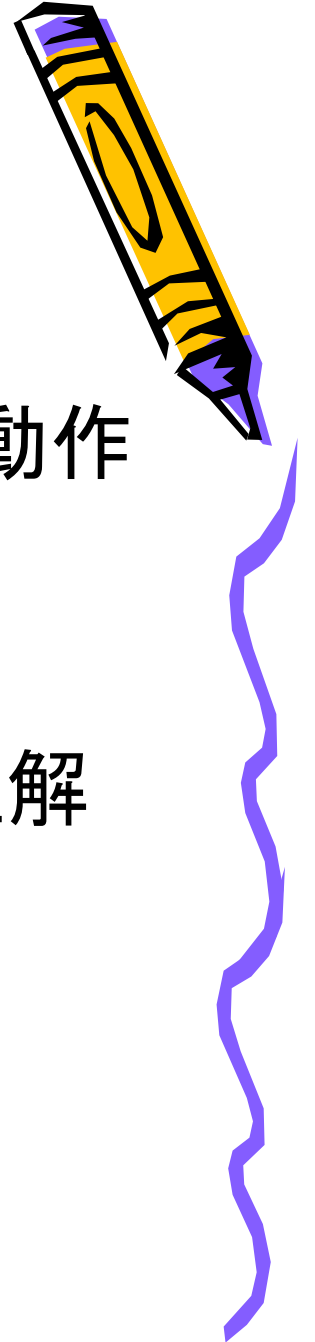
8月22日(月)2限目

Hello World!



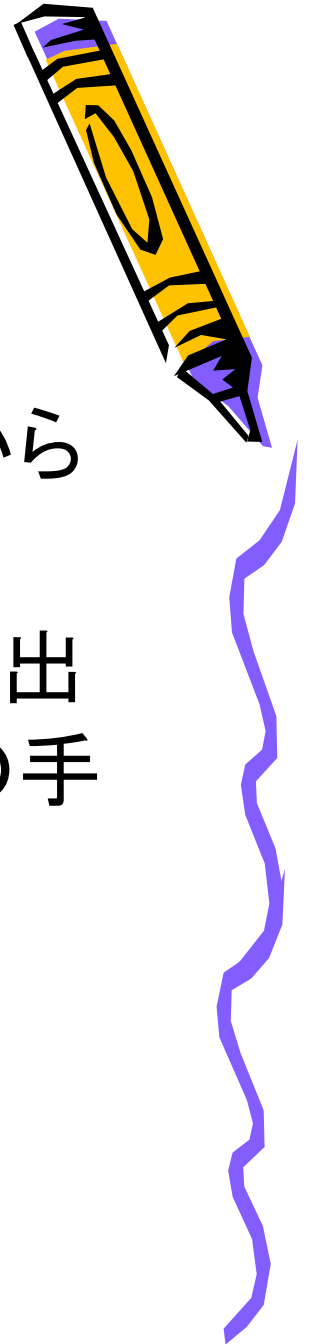
## 2限目の到達目標

- 白紙の状態からプログラムを書いて、動作させる。
- 作成・コンパイル・実行までの手順を理解し、関係する用語の概念を理解する。



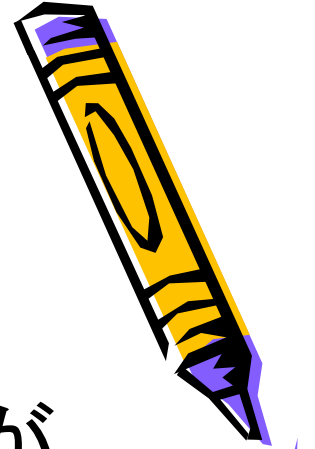
# テキストエディタの準備

- TeraPadのプログラムを、USBメモリからコピーして下さい。
- 演習が終了しても各自の成果物を取り出せるように、自分宛にメールするなどの手段を講じて下さい。



## プログラムフォルダの準備

- My Documentの下に作りたいのですが、再起動すると消えてしまうため、可能ならば、C:\JavaTestなど、作成できる場所にプログラムフォルダを作成します。
- 名称: JavaTest, JavaProgなど



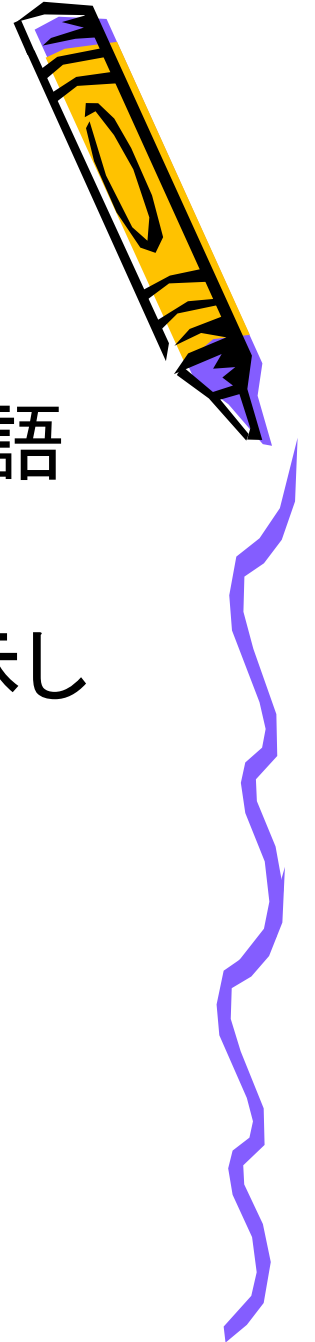
# プログラムの入力

- 教科書P33
- まずは、入力してみてください。



# コンパイルとは

- 人間が読める形のプログラムを、機械語に置き換える作業。
- わずかでもミスがあると「エラー」を表示して、文句を言ってきます。



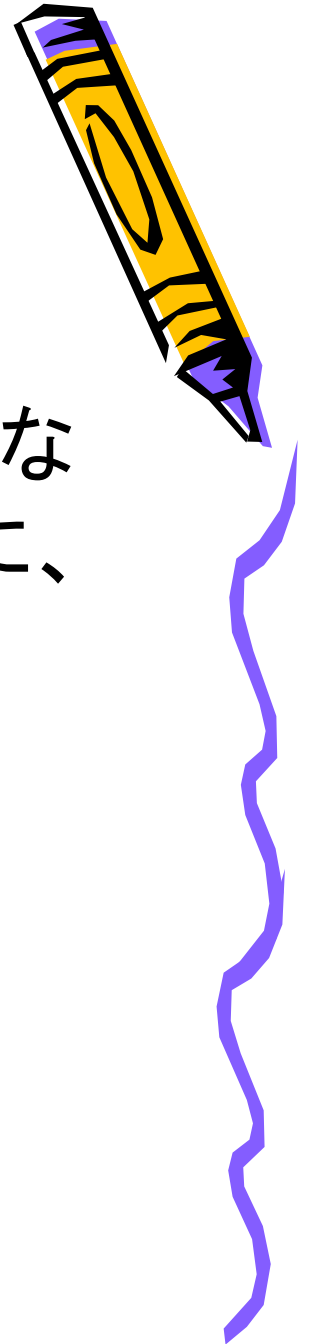
# コンパイラとインタプリタ

- コンパイラは、まとめて、予め機械語に翻訳しておく処理。
- インタプリタは、「逐次実行」で、プログラムを実行しながらその都度、機械語に翻訳していく処理、です。



# アプレットの作成

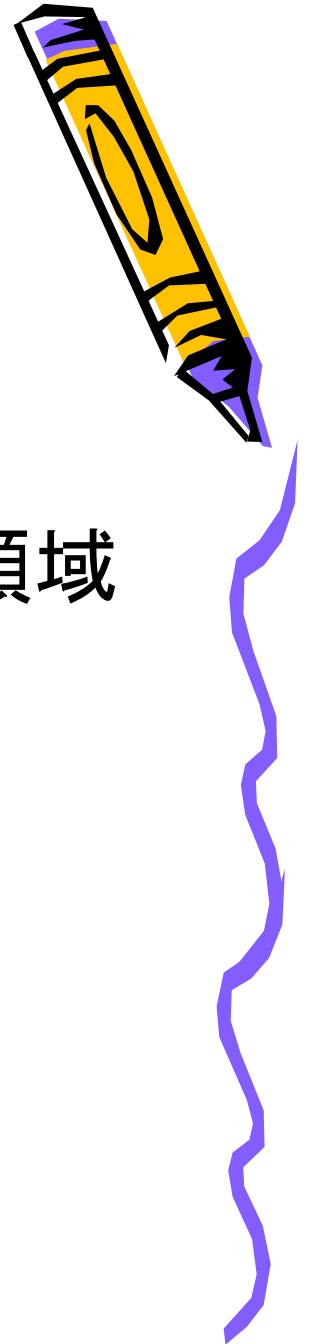
- アプレットを動作させる場合は、HTMLなどブラウザが直接処理できるファイルに、「アプレットの存在」を記述します。
- 教科書P37





# アプレットを読み込むHTML

- 教科書P37
- クラスファイルの名称、画面での占有領域の広さ、などを指定します。



# アプレットの動作

- ブラウザが開くために、そこにJavaのプログラムが存在することがわからないほど、画面の一部になっています。
- アプレットビューアなどで、動作を見ることができます。



# クラスファイル

- 「オブジェクト指向」で、Javaの中間コードは「クラスファイル」という形で記録されます。この形のファイルは、Java仮想マシンが動作する環境ならば、Windowsだけではなく、MacでもLINUXでも動作します。



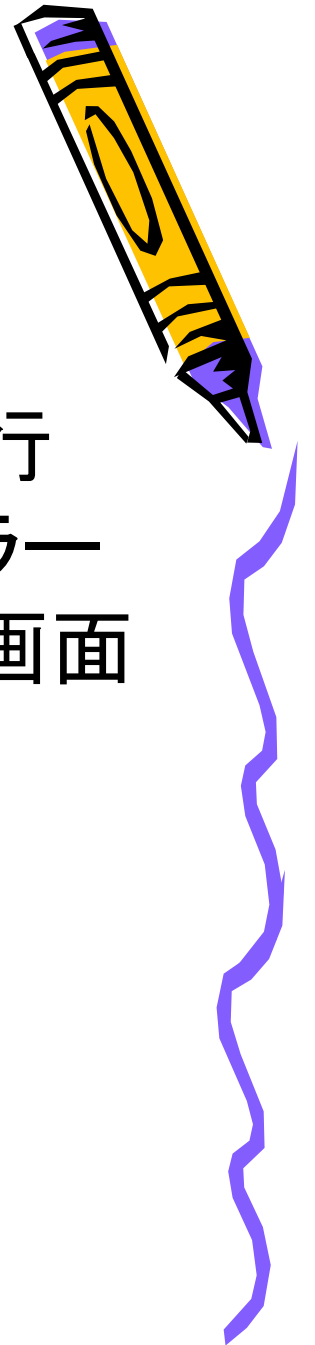
# 予約語

- 教科書P45
- Javaのプログラムを構成する「特別な意味のある言葉」です。
- この組み合わせで、プログラムを構成していきます。プログラムの「骨格」を作る言葉です。



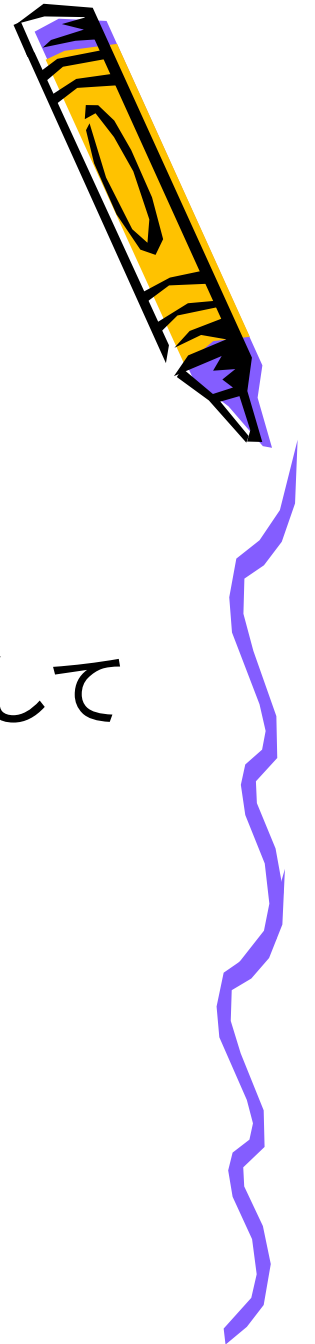
# コンソール

- コマンドを入力したり、プログラムの実行中の様子を出力したり、システムがエラーを通知したりする「司令塔」のモニター画面です。



# プログラムの動作

- プログラム中の  
`System.out.println( "メッセージ" );`  
というプログラムコードで、画面に出力して  
見ることができます。



## 2限目の到達目標

- 白紙の状態からプログラムを書いて、動作させる。
- 作成・コンパイル・実行までの手順を理解し、関係する用語の概念を理解する。

