

教科書第2章 Javaの基本 の内容を理解する

- プログラムを少しずつ変えて、いろいろと実行してみる (JCPadの操作に慣れよう ~ 自分でどんどん試せるように)
- プログラム(コード)の中身を、細かく見てみる (わけのわからない呪文でなく、人に説明できるようになる)
- 「リテラル」について理解する (何か、どんなものがあるのか、どう操作することができるのか、説明できるようになる)

1. 教科書第1章の補足説明

- p 10の「エラーが表示されてしまったら」について ⇒ まず、読んでみよう。
 - 用語 「デバッグ」 (と「バグ」)
 - 2種類のエラー: コンパイルするときに出るエラーと、実行したときに出るエラー
 - コンパイル時のエラー ~ これからの演習でたくさん経験してください。
 - 実行時のエラー ~ コンパイルは正常に終了するが、実行するとエラーになるもの。
 - 見えるエラー エラーメッセージが出て異常終了する。例: ゼロで割った
 - 見えないエラー エラーメッセージは出ない。正常終了する。でも結果がおかしい。例: 5と3を足して2で割る を $5 + 3 / 2$ と書いた??
 - コンパイル時エラーと実行時エラーを分けるには、コンパイルを単独ステップとして行う?
- p 16 練習2 で
 - プログラム中のクラス名 (class XXXXXと書いたXXXXXがクラス名) とファイル名 (保存するときにつけたファイルの名前) は、一致していないといけない (エラーになる)。⇒ あとで試してみよう。

2. 教科書 2. 1 いろいろと試して、JCPadに慣れる+画面への出力に慣れる

- p 18のSample1を試してみる
JCPadを起動して、教科書p18のSample1.javaを入力する。できたらコンパイルし、実行する。
 - ⇒ 細かい意味は、教科書 2. 3以降で詳しく学ぶが、ここでは「プログラムの全体のイメージ」を捉えて欲しい
 - ⇒ 教科書を見ないで、Sample1.javaを再現できますか??
- P20のいろいろな出力方法を知る ⇒ Sample2.javaで、printlnとprintの違いは何か?
 - ⇒ printlnは () だが、
 - printは ()
 - ⇒ 格好よく出力したいときにもう一度見直せばよさそう。(今はprintlnだけ覚えておけばいいかも)
- エラーの例

3. 教科書 2. 2 コードの内容 ~ (プログラム) コードの細かい説明

- 教科書p22~23。ここはきちんと理解しておこう。教科書p 22の Sample1.javaを眺めてみたい。

```

class Sample1
{
    public static void main(String[], args)
    {
        System.out.println("ようこそJavaへ");
    }
}

```

- 名前が () という「クラス」がある。このクラスの範囲は () ~ () 行目である。そのクラス () の中に、名前が () というメソッドが1個ある。このメソッドの範囲は () ~ () 行目である。この2つについて、 "{ " と " } " の役割は、 () この名前 () の付いたメソッドは特別扱いで、[重要]にあるように ()
- 教科書p 23~24。メソッドの中は、「文」 (Statement) がいくつか (普通はたくさん) 書いてある。「文」は (の) で、最後に「 () 」 (呼び方は ()) を付ける。
← 文の終りを示すマークになっている。(脱線) 記号文字には名前がある。調べてみると面白い。文は、(書いてある) 順番に実行される。教科書 図2-1では、() の次に () を実行。よく見ると「原則として」 ~ あとで、この原則でない場合が出てくる。(5章、6章)
- 教科書p 24~25。改行を入れる位置、空白を入れる位置について ⇒ 教科書を読んでみよう。(注意) 「入れてもよい」と「入れなければならない」とは違う。「入れてはいけない」にも注意。p25の2行目のように、単語の途中で空白を「入れてはいけない」のだが、単語の区切り(単語と単語の間)には空白を「入れなければならない」。
- インデント (字下げ、段下げ) について：
p 25の [重要] コードを読みやすくするためにインデントや改行を使う。(注意) どこで改行や段下げをするか ⇒ コーディングの「スタイル」
~ 授業内では教科書のスタイルを守ろう。
世の中にはいくつかの異なるスタイルが見られるし、企業ではその中のスタイルが決められていることも多い。(スタイルおまけ情報) 「名前」の付け方のスタイルもいろいろある。(「しなければいけない」ではないが)
例: クラスの名前 (Sample1とか) は先頭が大文字、2つの語をくっつけるときは下線「_」で繋ぐ、等
>> 「名前」の先頭は数字は不可で、英字でなければならない (スタイルではなくて文法上の決まり)
- 教科書p 26~27。コメントを書こう p 26の [重要] コメントを使って、コードの内容を分かりやすくする何のために? プログラムを作った時の意図「つもり」をメモしておく。①自分が後で見る ②人に渡す何について? 例) 何をするプログラム (やクラス・メソッド) か、何をする「部分」か、「変数」の意味付け...
その他、企業などで形が決まっていることもある 作成者・作成日・製品番号・バージョン番号.....
書き方 2通り。① //から行の終わりまで (1行単位) ② /* から */まで囲まれている部分
- 教科書p 28の [重要] Javaのコードは1つ以上のクラスから成り立つ
- 教科書p 28の「開発の際の規則」 (上の「スタイル」の話) たとえば:
インデント (段下げ) いくつか改行・いくつか段下げ、どれだけ段下げ (多少スタイルの違いがある)
コメントの書き方 何の情報を、どこに書くか
名前付けの規則 教科書では「クラスの名前の先頭を大文字」。その他いろいろな名前付けの規則を決める。

4. 教科書 2.4 文字と数値 ～ リテラル

- 教科書p 29の、リテラルとは??

Javaの文字・数値・文字列の(プログラム中での)書き方

前提 → `System.out.println(***)` で *** に書かれているものをプリント(画面に表示)できる。

*** に、リテラルを書くことができる。リテラルとは教科書p 30には ()

追加説明: 次の章で「変数」を学ぶが、それに相対する言葉として「定数」がある。その「定数」に当たる。

リテラルには、() () () などがある。

- [課題] 教科書p 29のSample3.javaを試してみよう。

- 教科書p 31~34 文字リテラル= 1文字のリテラル 例= () 書き方= () で囲む
エスケープ・シーケンス = キーボードで書けない字などを書く方法

[家で考えて欲しい課題] 表2-1の字がなぜエスケープ・シーケンスでないと書けないのか、考えてみよう。

- [課題] 教科書p 32のSample4.javaを試してみよう
- [追加課題(時間があれば)] Sample4.javaでは\\と\'しか試していないが、表2-1の他の文字を試そう
- [課題] 教科書p 34の文字コードのサンプルSample5.javaを試してみよう。

エスケープ・シーケンス 「\ooo」は8進数で文字を表す書き方、「\uhhhh」は16進数で文字を表す書き方。

[補足] 文字はコンピュータ内では数で表される。どの文字を何の数で表すかを決めるのが「コード体系」

Sample5にある「8進数の101(10進に直すと65)が文字Aを表す」のは

Shift-JIS コード体系やUnicodeコード体系でAを表すのは65と決めているから。

コード体系は標準としていくつかのものがある。Windowsで使われるShift-JISや、LinuxやMacOSで使われる

Unicodeはそれの1つ。他にもいろいろある。← 自分で調べてみると面白い。

[補足] 数の10進表記・8進表記・16進表記のやり方や変換法は、p 38~39。詳しくは別の授業で。

大文字 A は $65_{10} = 101_8 = 41_{16}$ で表され小文字 a は $97_{10} = 141_8 = 61_{16}$ で表される

- 教科書p 35 文字列リテラル=複数文字の並びのリテラル 例= () 書き方= ()

文字リテラルと文字列リテラルは、くくる文字(引用符)の形が違うので注意。

文字列を(楽に)扱うための便利な手続きがある ⇒ 第10章(春学期授業ではやらない)

- p 36文字列同士は、「+」で結合できる。

- 教科書p35~37 数値リテラル=数値 例= () 書き方= () ~ 「'」や「"」でくくらない

整数のリテラル と 小数(浮動小数点数)のリテラル ~ 整数で書くか、小数で書くか。

→ 整数と浮動小数点数は、コンピュータ内では別物として扱う。細かい話はアーキテクチャの授業で。

8進表記、16進表記もある。

先頭に余分な0が付いている → 8進表記と解釈 例 $010 \rightarrow 10$ 進で数えると ()

先頭に0xが付いている → 16進表記と解釈 例 $0x10 \rightarrow 10$ 進で数えると ()

それ以外(先頭に何も無い、いきなり0以外から始まる) → 10進表記と解釈

- [課題] 教科書p 36のSample6.javaを試してみよう。

第2回 小テスト（今日の復習）教科書p 41～42の練習問題をやってみます。 この回答欄に書き込んでください。

1. 誤りがあれば訂正してください ←どんなエラーが出るか試してみましょう（自習課題）

```
// 画面に文字を出力するコード
class Sample1{public static
    void main (String[] args)
    (System.out.println("ようこそJavaへ!");
    System.out.println("Javaをはじめましょう!"); } }
```

2. 適切な位置に、コメントとして「文字と数値を出力する」を挿入してください

```
class Sample2
{
    public static void main(String[] args)
    {
        System.out.println('A');
        System.out.println("ようこそJavaへ!");
        System.out.println(123);
    }
}
```

3. 次の出力をするコードを記述してください（PC上で実際に動かしてみましよう）

```
123
¥100もらった
またあした
```

4. タブ記号を表すエスケープシーケンス(\t)を使って、次の出力をするprintlnのコードを記述してください

```
1 2 3          ← 数字の間にタブ記号が入っている
```

5. 次のように出力するprintlnのコードを記述してください。8・16進数を使って2通りのコードを作ること

```
6
20      10進の20は8進では（    ）、16進では（    ）
13
```
