

1-3.Javaプログラムの実行方法と課題提出方法

やるべきこと: 提出する

やるべきこと: 評価を受ける

■簡単なJavaプログラムを実行してみる

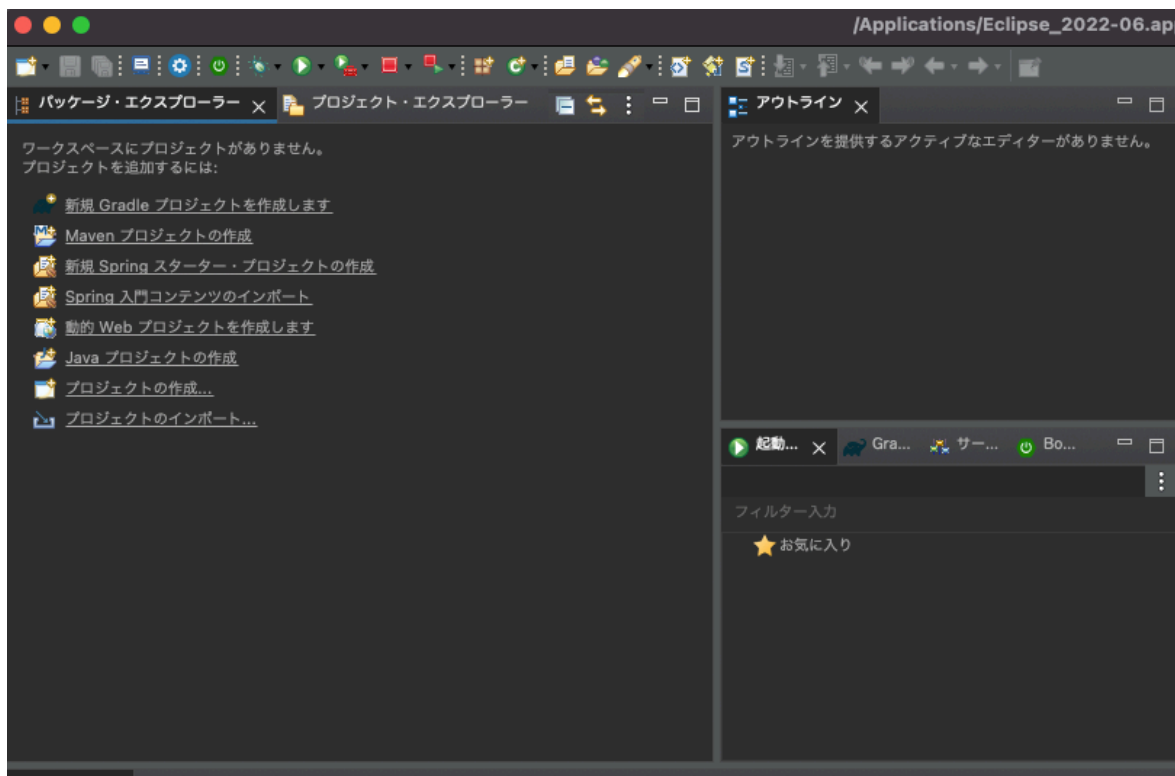
Javaのプロジェクトの作成と、プログラムの実行方法についてまとめます。

カリキュラム用のプロジェクトはほとんどインポートして実行できるようにしていますが、自作する課題や、練習のコードを書く場合などに利用して下さい。

①

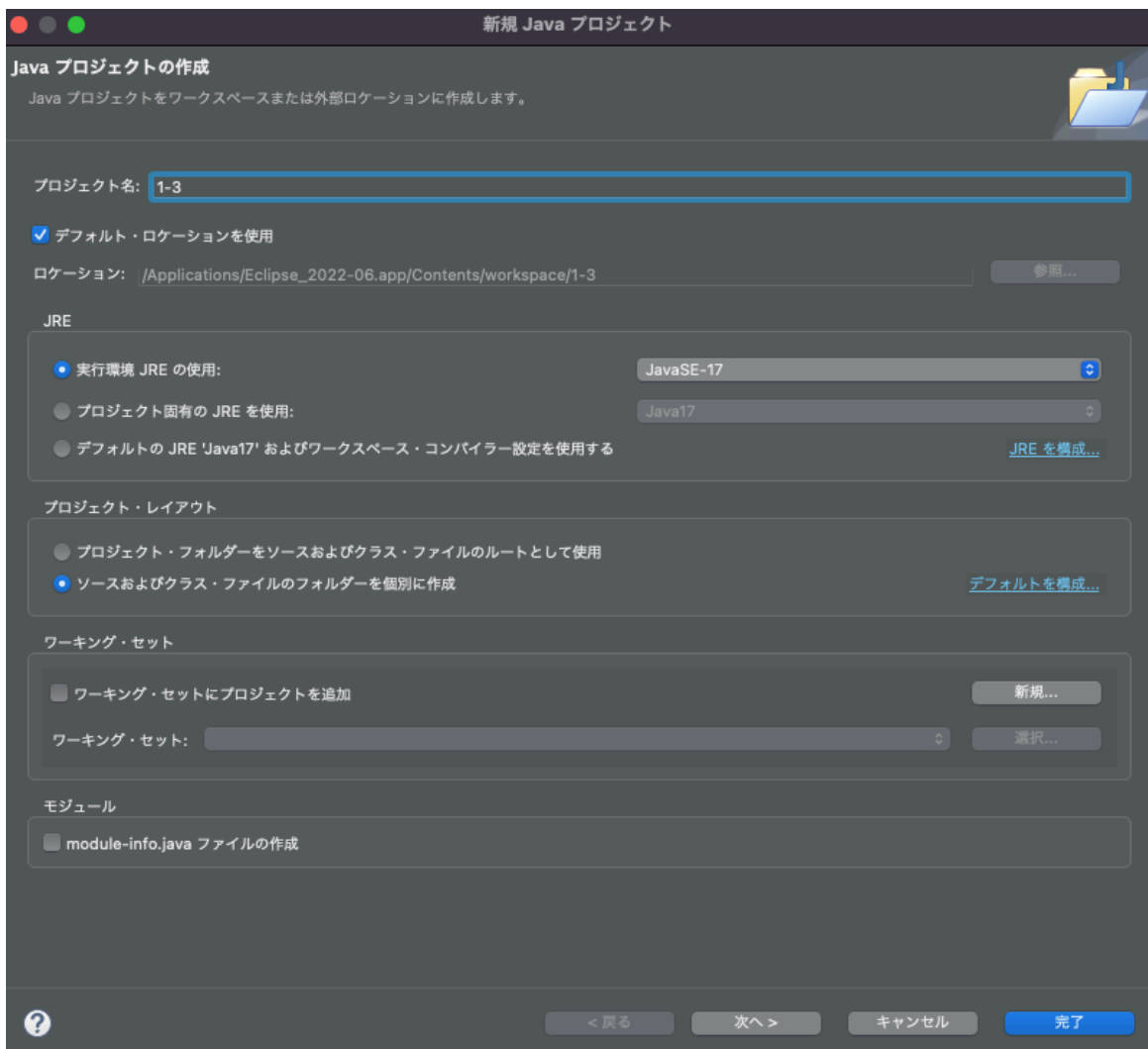
パッケージエクスプローラー欄にある「Javaプロジェクトの作成」をクリックします。

「ファイル」メニュー、または右クリックメニューから「新規」→「プロジェクト」でも選択できます。



②

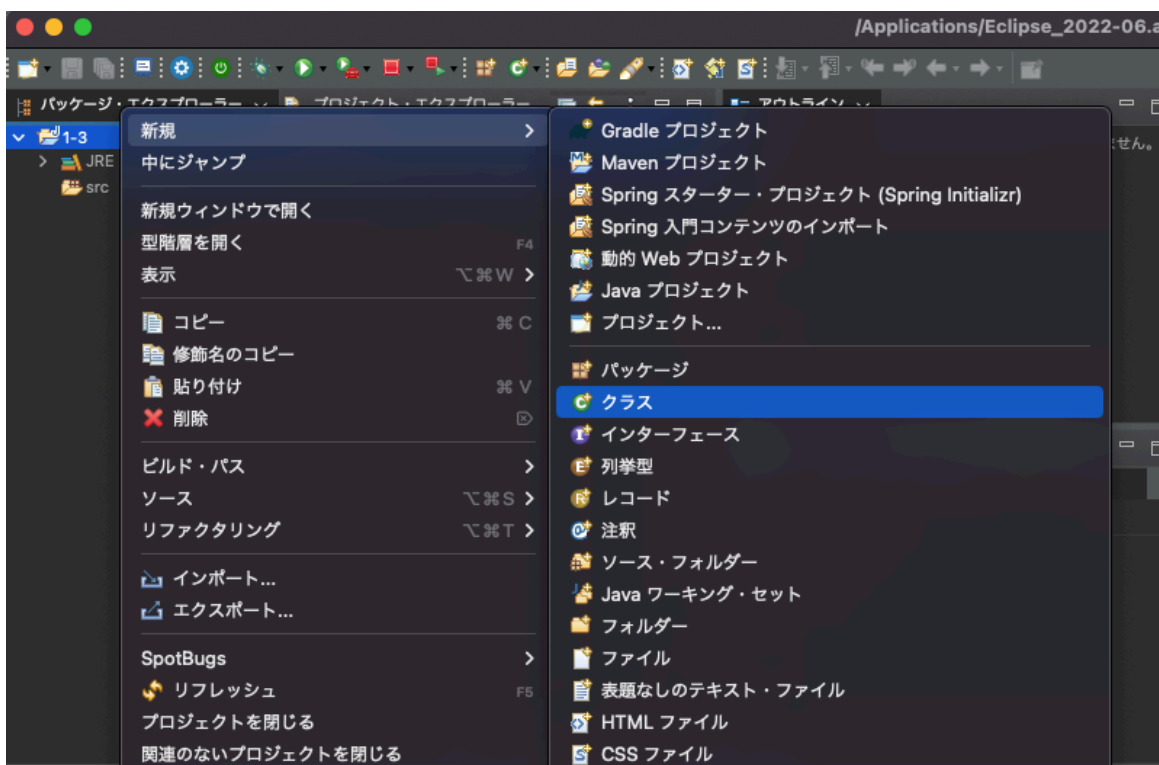
プロジェクト名を入力して、「完了」をクリックします。ここでは「1-3」として下さい。



③

パッケージ・エクスプローラーに作成したプロジェクトが追加されています。

プロジェクトを右クリックして、「新規」→「クラス」をクリックします。



④

新規Javaクラス ダイアログが表示されます。名前 の欄にクラス名を入力します。

※ クラス名の最初の文字は、半角英大文字にする決まり がありますので注意してください。ここでは「Test」とします。
続けて、`public static void main(String[] args)` のチェックボックスにチェックを入れて、完了を押します。

⑤

下記のようにエディタ部分にソースファイルが作成されます。

⑥

`public static void main` という箇所から波括弧で囲まれている部分に、


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

と入力して下さい。

エディタ画面で書き込んだコードは、下記ショートカットで上書き保存することができます。

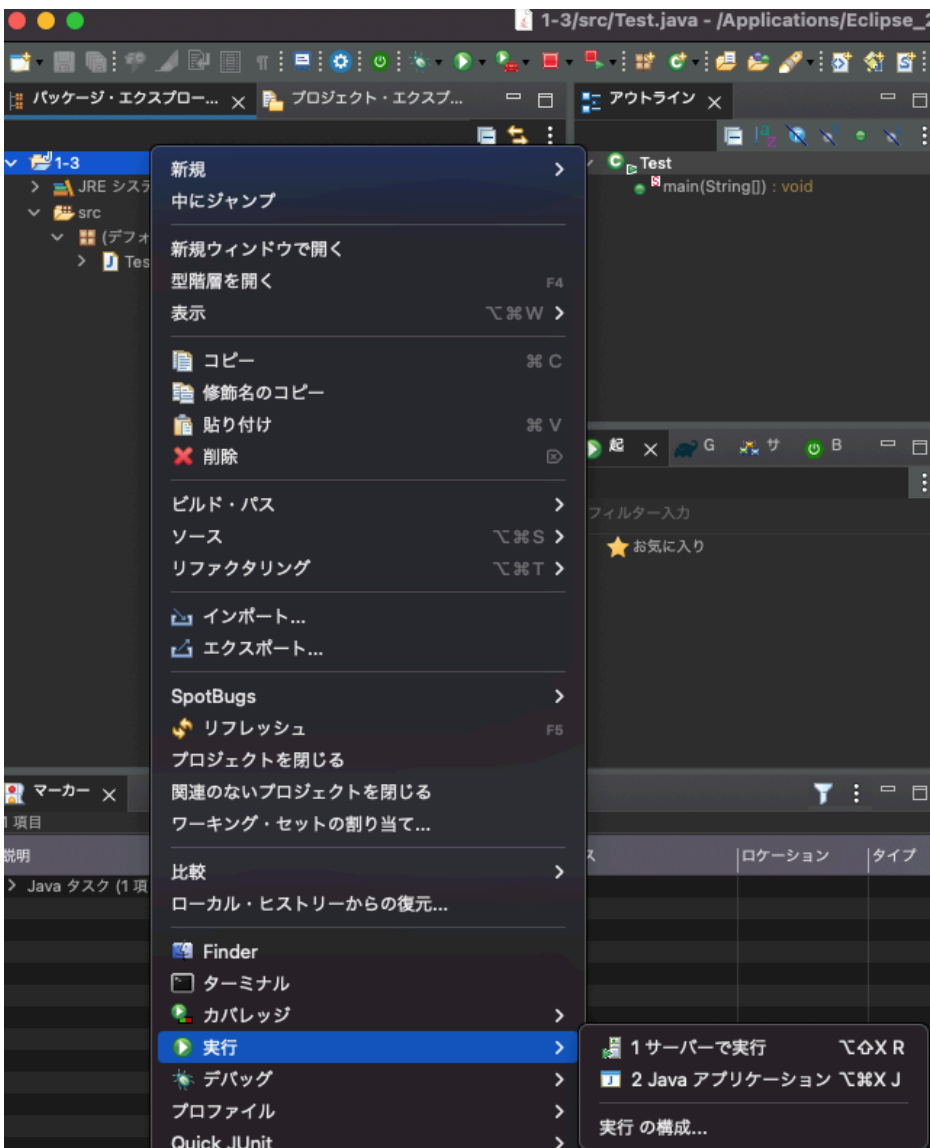
- 上書き保存ショートカット

- Windows : `ctrl + S`
- Mac : `command + S`

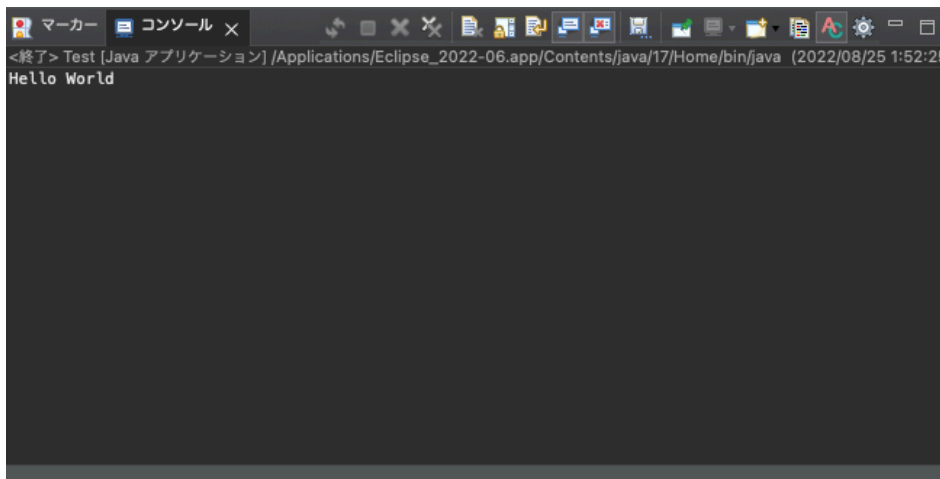


```
1
2 public class Test {
3
4     public static void main(String[] args) {
5         // TODO 自動生成されたメソッド・スタブ
6         System.out.println("Hello World");
7     }
8
9 }
10
```

⑦
コードを保存したら、プロジェクトフォルダ、または起動したいjavaファイルを右クリックし、「実行」→「Javaアプリケーション」をクリックしてください。



⑧
Eclipseの左下の「コンソール」というタブの場所に、今回のコードの実行内容が表示されます。



Javaのプログラムの簡単な解説

まず初めに、必ず「`public class クラス名`」と記述し、中括弧（`{}`の記号）で囲みます。
Javaのプログラムでは、このように中括弧で囲まれたプログラムコードのまとまりを**ブロック**と呼びます。

今回は、**Testクラス**を定義したことになります。
クラスを定義して、その中に処理を組み込みます。その積み重ねで、アプリやゲームは動いています。

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println("Hello World");  
    }  
}
```

次に、Testクラスのブロックの中に、更に **main** のブロックがあります。これを **mainメソッド** と呼びます。
Javaでは、プログラムを実行すると、mainメソッドの処理から開始されます。
このmainメソッドは、`public static void main(String[] args)` という形式で記述する決まりになっています。
まずは深いことは考えずに、そういうものだと思って下さい（mainメソッドは、後ほど詳しく説明します。）

※**public class (クラス名) = (クラス名).java** というように、
プログラム内のpublicのクラス名とファイル名は同一にする必要があります

(例えば、`public class Test`を書いたファイル名は、`Test.java`にする。)

System.out.println()

mainメソッド内に一行だけ、`System.out.println()` というコードが記述されています。
Testクラス の **処理** は、`System.out.println()` を実行する。と読むことができます。

ちなみに `System.out.println()` のことを一般的にprintlnメソッドと省略して呼称します。

System.out.println() の括弧の秘密

丸括弧内に指定される **引数（ひきすう）** の文字のデータを画面に表示して改行を行います。
上記のプログラムでは、printlnメソッドの引数に、ダブルコーテーションで囲んだ `Hello World` を指定しています。

```
// このコードの指示は、Hello Worldを表示します。  
System.out.println("Hello World");
```

；（セミコロン）の秘密

printlnメソッドの引数の丸括弧を閉じたあとに、セミコロン `;` が付いています。
Javaにおいては、処理の末尾に「`;`」を付与することがルールとして定められており、日本語で言うところの「。（句点）」に相当します。

エラーの原因の一つとなりますので、意識して付けるようにしましょう！

コメント

// コメントその1

・コメントその1 (//)

コード上に、 // (スラッシュ2つ) で記載された内容がある場合、
その行のコメント以降の文章はすべてプログラムコードとして認識されません。
コードに関する説明やメモ書きの用途で使われます。

```
/* コメントその2 */
```

```
/*  
コメントその2  
このコメントは、  
複数行に渡って機能します。  
...  
*/
```

・コメントその2 (/* コメント */)

前述のコメントとの違いは、「**複数行に対応する**」という点です。
1行でも記述可能ですが、内容が複数行必要な場合はこちらのコメントを使用するケースが多いです。

- 1行で説明するには長すぎる
- 1行コメントを複数行記述するとまとまりが分かりづらい

等の場合はこちらを使用しています。

コメントは記号を打ち込んでもいいですが、対象の行で下記ショートカットキーの入力でコメント化することができますので活用しましょう。

Windows: ctrl + / (スラッシュ)

Mac: command + / (スラッシュ)

その他の便利機能

Eclipseでコードを書く時に便利なショートカットキーがいくつかありますので、下記ページなど参考にして使ってみてください！
コード補完、コード整形などは特に使う機会が多いです。

<https://blog.codecamp.jp/eclipse-shortcut>

■課題の提出方法

プロジェクトフォルダを右クリックし、Windowsは「Explorer」、Macは「Finder」をクリックします。



エクスプローラーやFinderで、ファイルが格納されているworkspaceを開くことができます。



提出するフォルダを、右クリックメニュー等からzipファイルに圧縮して、カリキュラムページから提出して下さい。

「Discord」→課題提出窓にてzipファイルを提出し、カリキュラムページの入力欄にそのメッセージのURLを入力して提出して下さい。

※投稿したメッセージにカーソルを合わせる → 矢印のアイコン「メッセージを共有」 → 「リンクをコピー」 →

テキスト入力欄にてチェーンのアイコン「リンク」 → コピーしたURLを貼り付け → 「リンクを作成する」

評価概要

学生から秘匿	No
参加者	103
提出	59
要評価	3

提出物をアップロード・入力する

提出ステータス

提出回数	これは 1 回目の提出です。
提出ステータス	まだ提出されていません。
評価ステータス	未評価
最終更新日時	-