

---

# Java

Javaの基礎

---

## 2 時間目

# プログラミングとは

## 解説

プログラミングとは、ある目的の為に必要な手順を「特定のルール」に沿ってプログラムを書く作業です。私たちの周囲の多くのモノがプログラミングで出来ています。

- 1 複雑なwebサイト
- 2 ゲーム
- 3 スマートフォンのアプリ
- 4 銀行の預金口座・ネットバンキング
- 5 カーナビ

# プログラミングとは

プログラミングの作業は大きく2つ

- 1 必要な手順を考える作業（設計/アルゴリズム）
- 2 プログラミング作業（開発）

# Javaの基本ルール

- ① エディタとコンパイラ、または統合開発環境を使用
- ② ソースファイルを〇〇〇.javaとして保存
- ③ コンパイルして実行ファイル（exe）を作成

# Javaを書くためのツール

Eclipse

JBuilder

IntelliJ IDEA

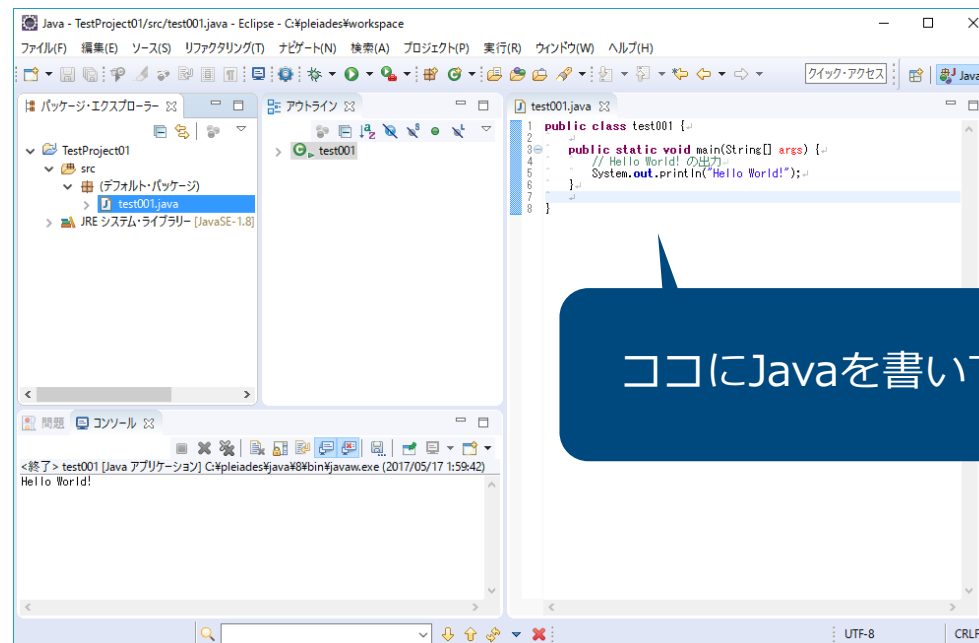
Xcode

Emacs

メモ帳

・・・などなど

本講座では **Eclipse** (エクリプス/イクリプス) を利用

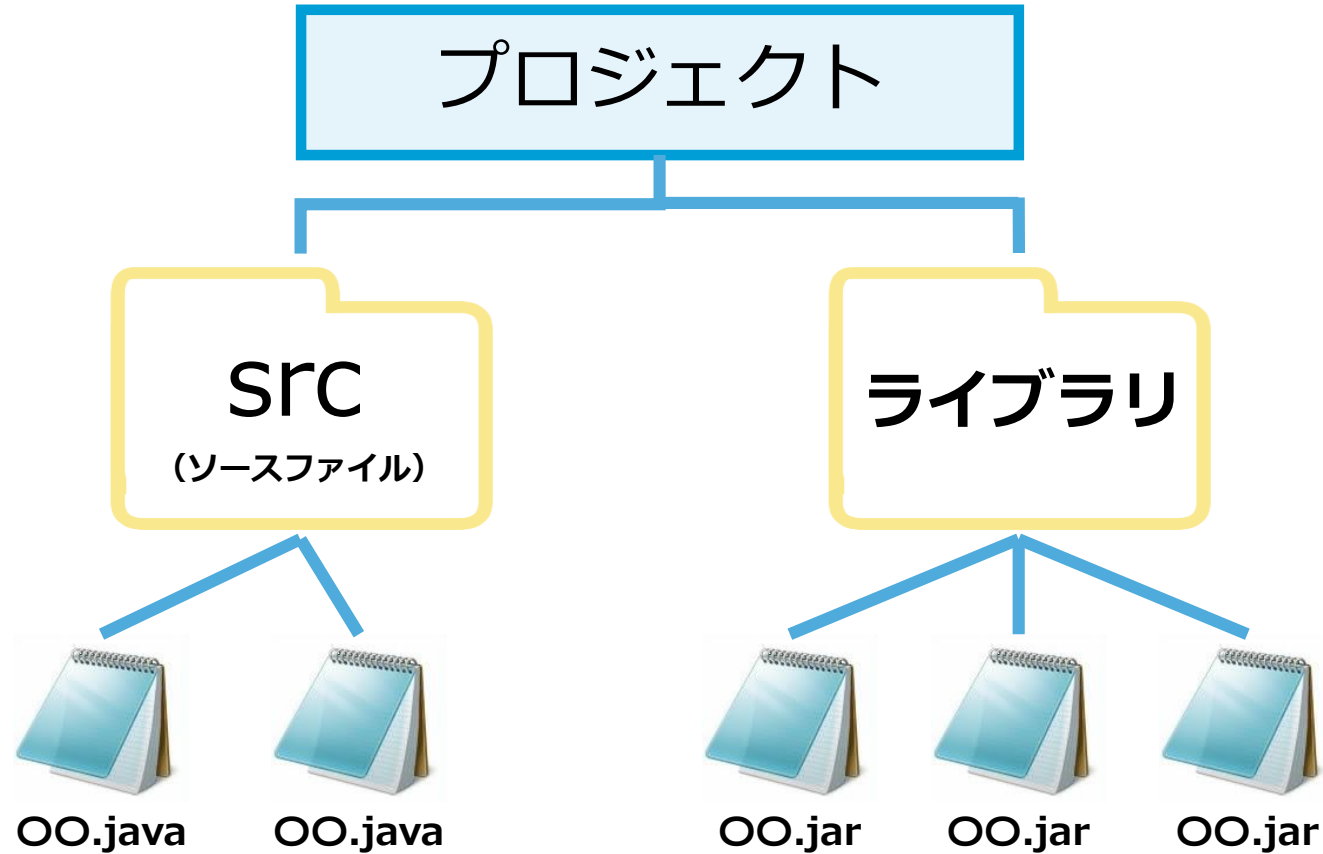


ココにJavaを書いています。

# Javaプロジェクトの新規作成

解説

Javaのプログラムは、"プロジェクト"と呼ばれるワンセット単位で管理します。



補足

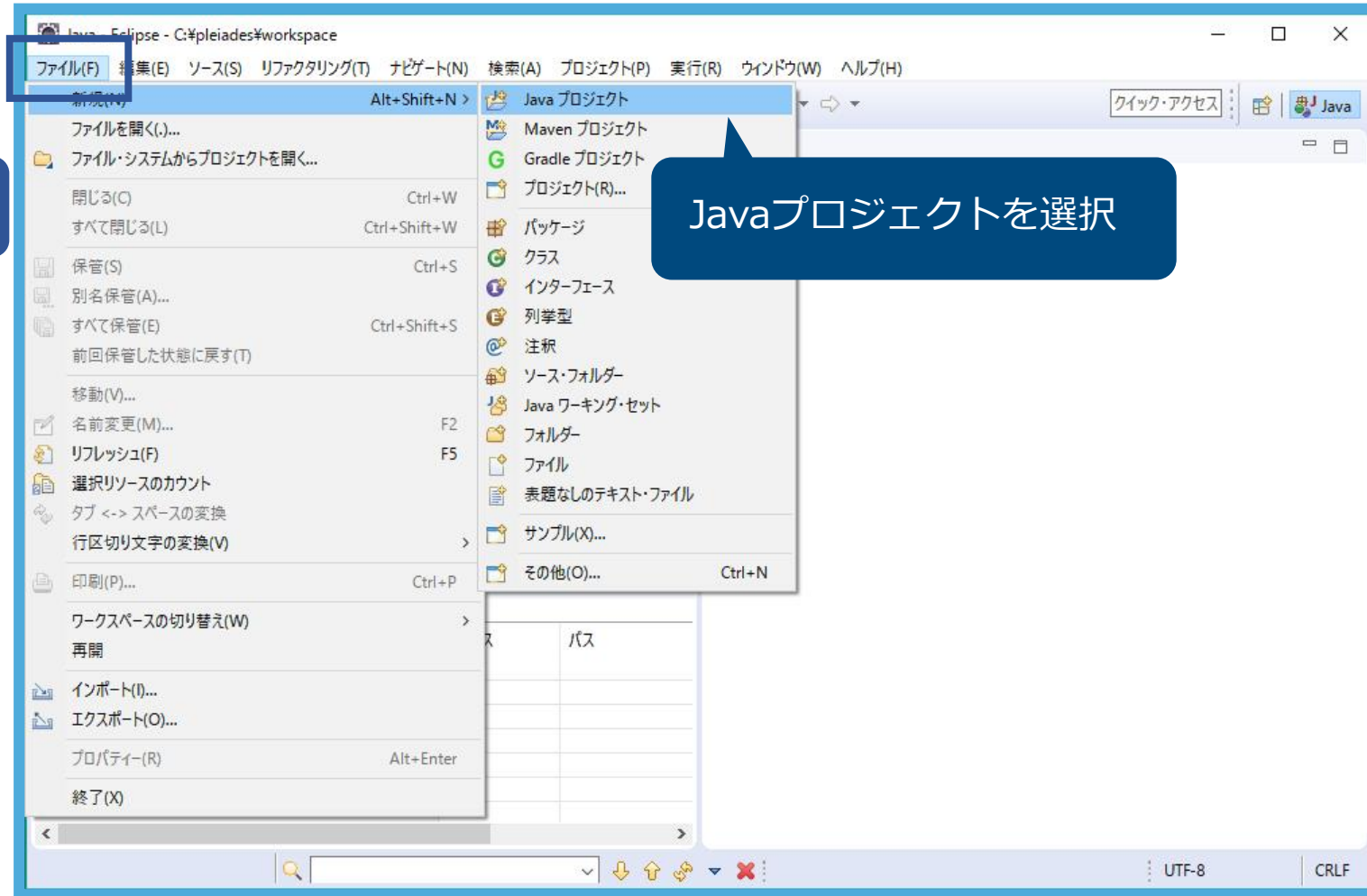
ライブラリとは、よく使うプログラムを再利用可能な部品として用意したものです。プログラムの中から自由に呼び出して使うことができます。

# Javaプロジェクトの新規作成

解説

Eclipseを起動し、メニューバーより[ファイル(F)]→[新規(N)]→[Javaプロジェクト]を選択します。

ファイルをクリック



Javaプロジェクトを選択

# Javaプロジェクトの新規作成

解説

"プロジェクト名"に任意の名前を入力し、"次へ"ボタンをクリックします。  
今回は、"TestProject01"と入力して進めます。

ソースファイルなど、  
プロジェクトが保存されている  
場所のパスが表示される

プロジェクト名を入力  
今回は、『TestProject01』と入力

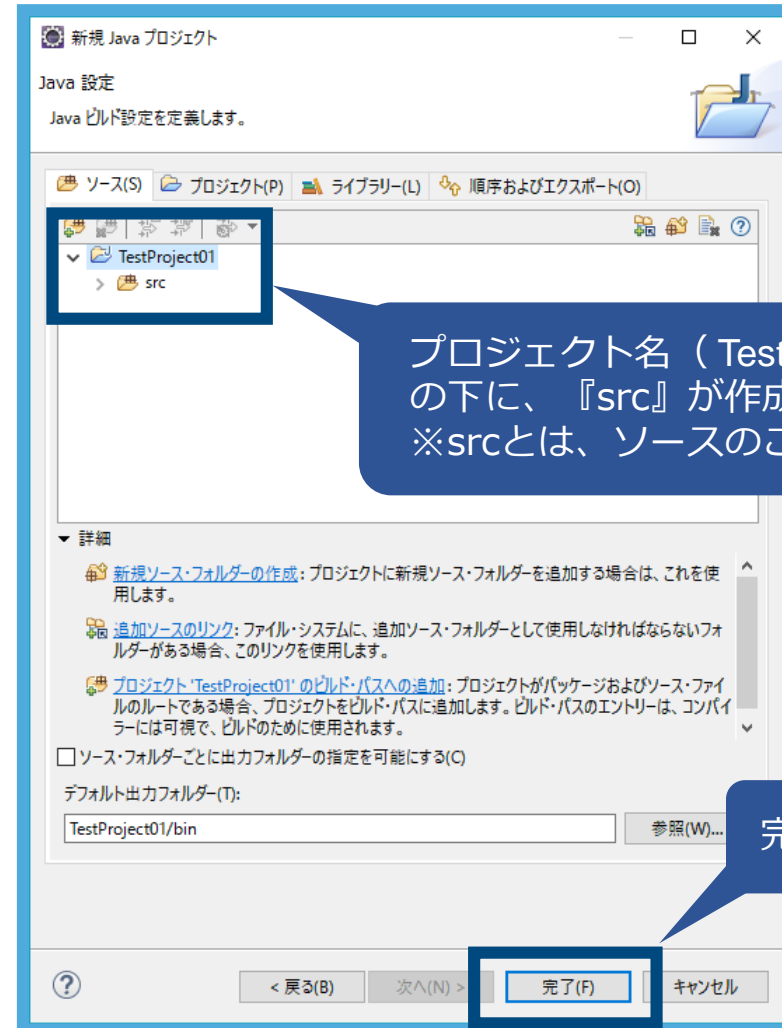
次へをクリック



# Javaプロジェクトの新規作成

解説

"プロジェクト名"のフォルダ下に"src"フォルダが作成されていることを確認し、"完了"ボタンをクリックします。



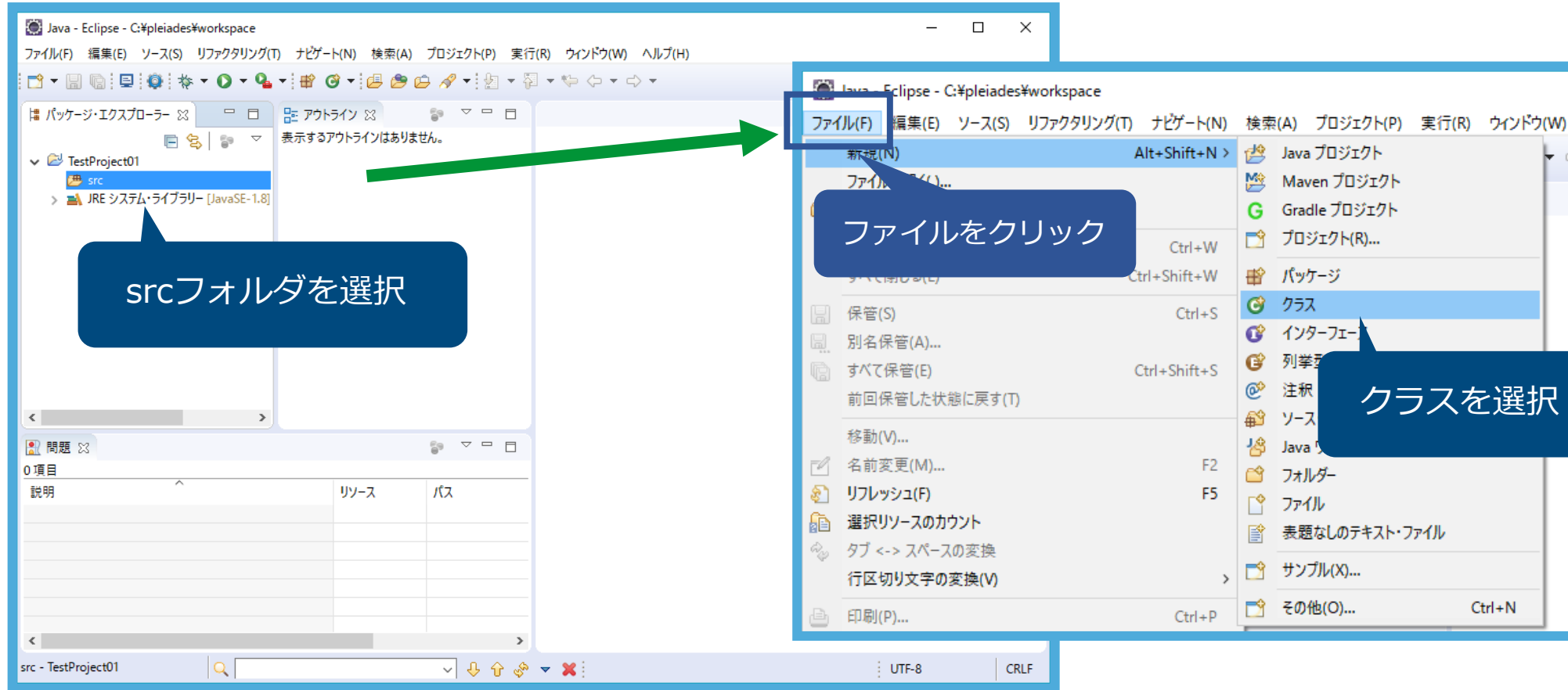


# Javaプログラム（クラス）の新規作成

## 解説

プログラムの本体となる、ソースファイルを作成します。

"src"フォルダを選択した状態で、メニューバーより[ファイル(F)]→[新規(N)]→[クラス]を選択します。



## 補足

Javaのプログラムはクラスの集合で作られます。  
今はJavaのソースファイル=クラスという認識で問題ありません。

# Javaプログラム（クラス）の新規作成

解説

"名前(M)"の欄に任意のクラス名を入力し（クラス名の頭文字には、一般的に大文字を使用します）、  
"Public static void main(String[] args)(V)"にチェックを入れ、"完了"ボタンをクリックします。

クラス名を入力

新規 Java クラス

Java クラス

デフォルト・パッケージの使用は推奨されません。

ソース・フォルダー(D): TestProject01/src 参照(O)...

パッケージ(K): (デフォルト) 参照(W)...

☐ エンクロージング型(V): 参照(W)...

名前(M): HelloWorld

修飾子: ☒ public(P) ☐ パッケージ(C) ☐ private(V) ☐ protected(T)  
☐ abstract(T) ☐ final(L) ☐ static(C)

スーパークラス(S): java.lang.Object 参照(E)...

インターフェース(I):

どのメソッド・スタブを生成しますか?

☒ public static void main(String[] args)(V)

☐ スーパークラスからのコンストラクター(C)

☒ 継承された抽象メソッド(H)

コメントを追加しますか? (テンプレートの構成およびデフォルト値についてはここを参照)

☐ コメントの生成(G)

完了(F) キャンセル

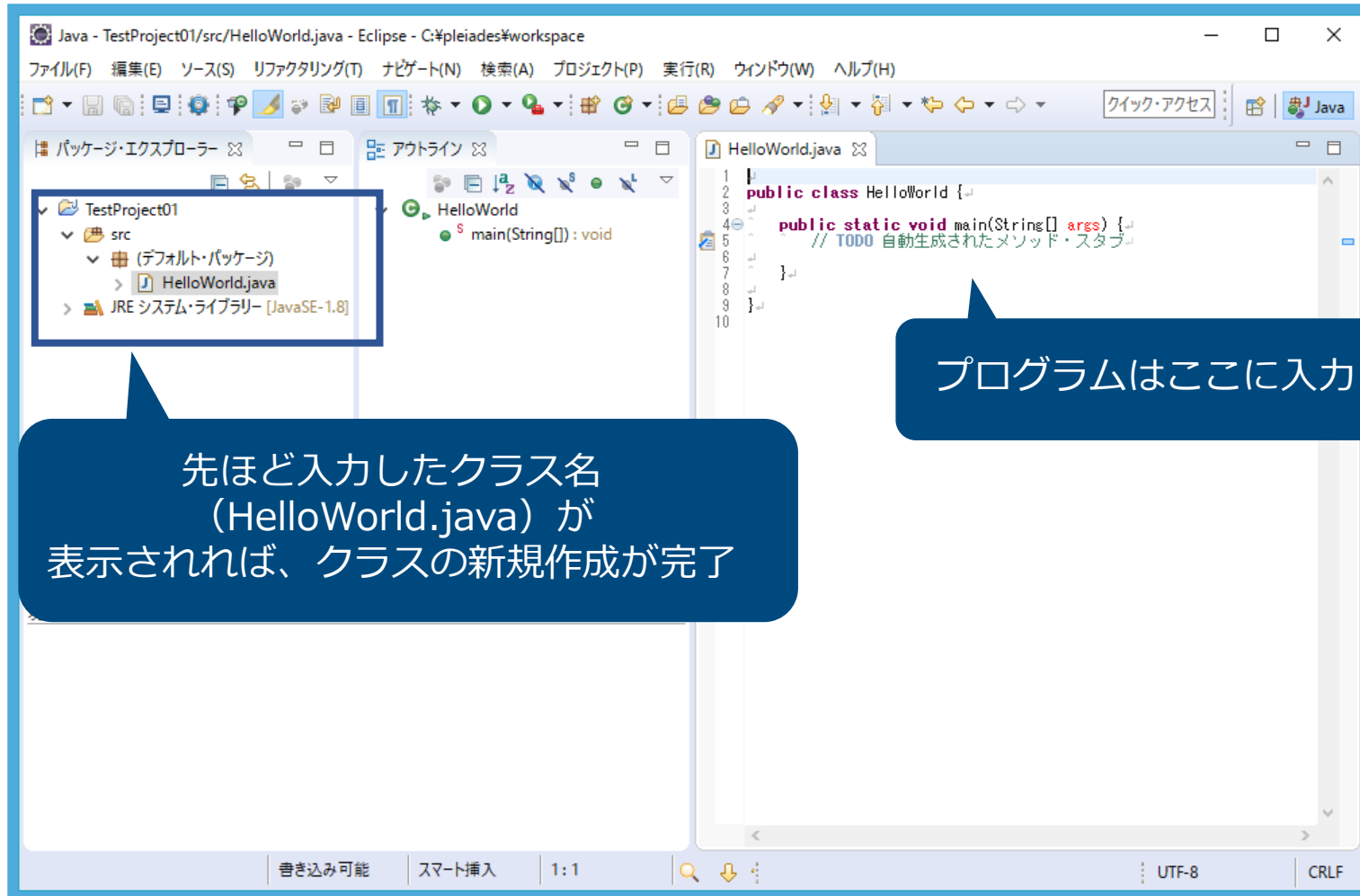
Public static void main(String[] args)(V)  
にチェック

完了をクリック

# Javaプログラム（クラス）の新規作成

解説

"src"フォルダ下に作成した.javaファイルが表示されたら、完了です。



# Javaプログラムの書き方

## 解説

これまでの手順でクラスを作成すると、下のような枠が自動生成されます。  
自動生成で出来上がった{}の内側に、プログラム本体を書いていきます。

クラス名.java(javaファイル)

```
public class クラス名 {  
    public static void main(String[] args) {  
        // TODO 自動生成されたメソッド・スタブ  
  
        . . . プログラムを記述する場所 . . .  
    }  
}
```

## 補足

"//"から始まる「// TODO 自動生成されたメソッド・スタブ」は、コメント（プログラムに影響を与えない注釈）です。初期状態で表示されているので不要な場合は修正・削除しても問題ありません。

HelloWorld.java(javaファイル)

```
public class HelloWorld {  
    public static void main(String[] args) {  
  
        // Hello World! の出力  
        System.out.println("Hello World!");  
  
    }  
}
```

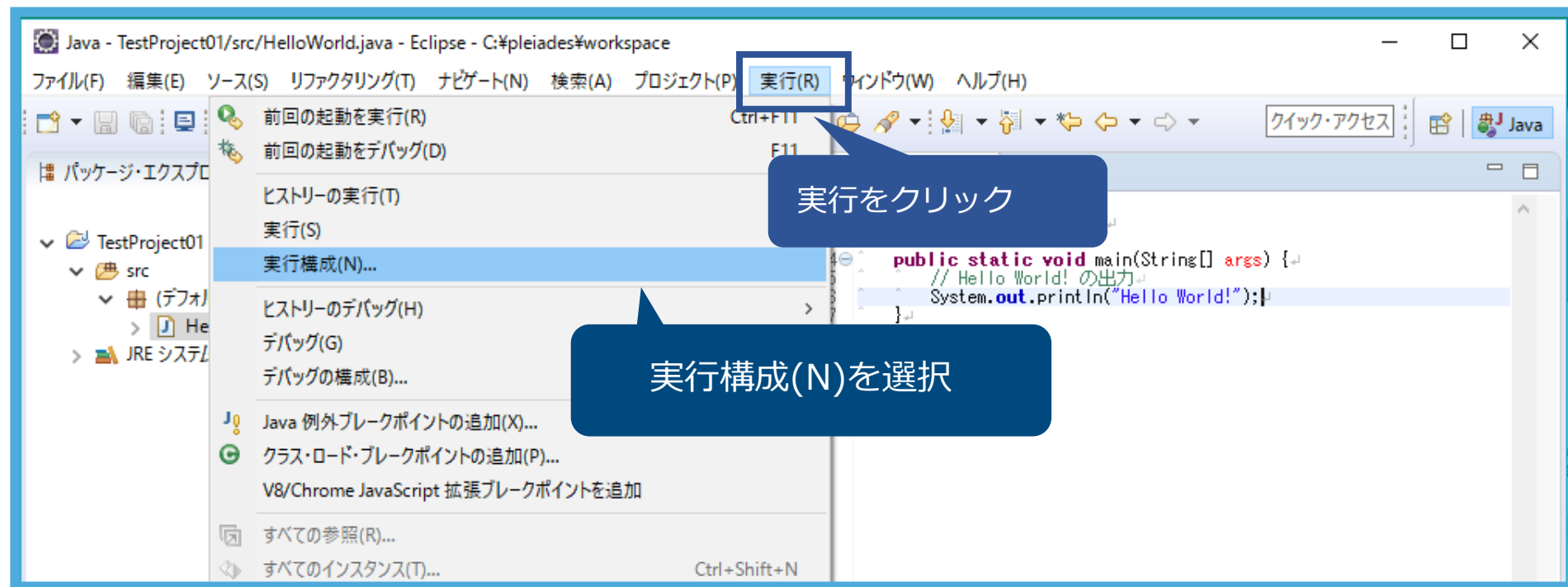
コメントを修正し、  
プログラムを追記した

# Javaプログラムの実行


## 解説

作成したプログラムを実行します。

メニューバーより[実行(R)]→[実行構成(N)]を選択します。



## 補足

2回目以降は  ボタンを押すだけで、自動的に前回登録した構成を選択して実行が始まります。

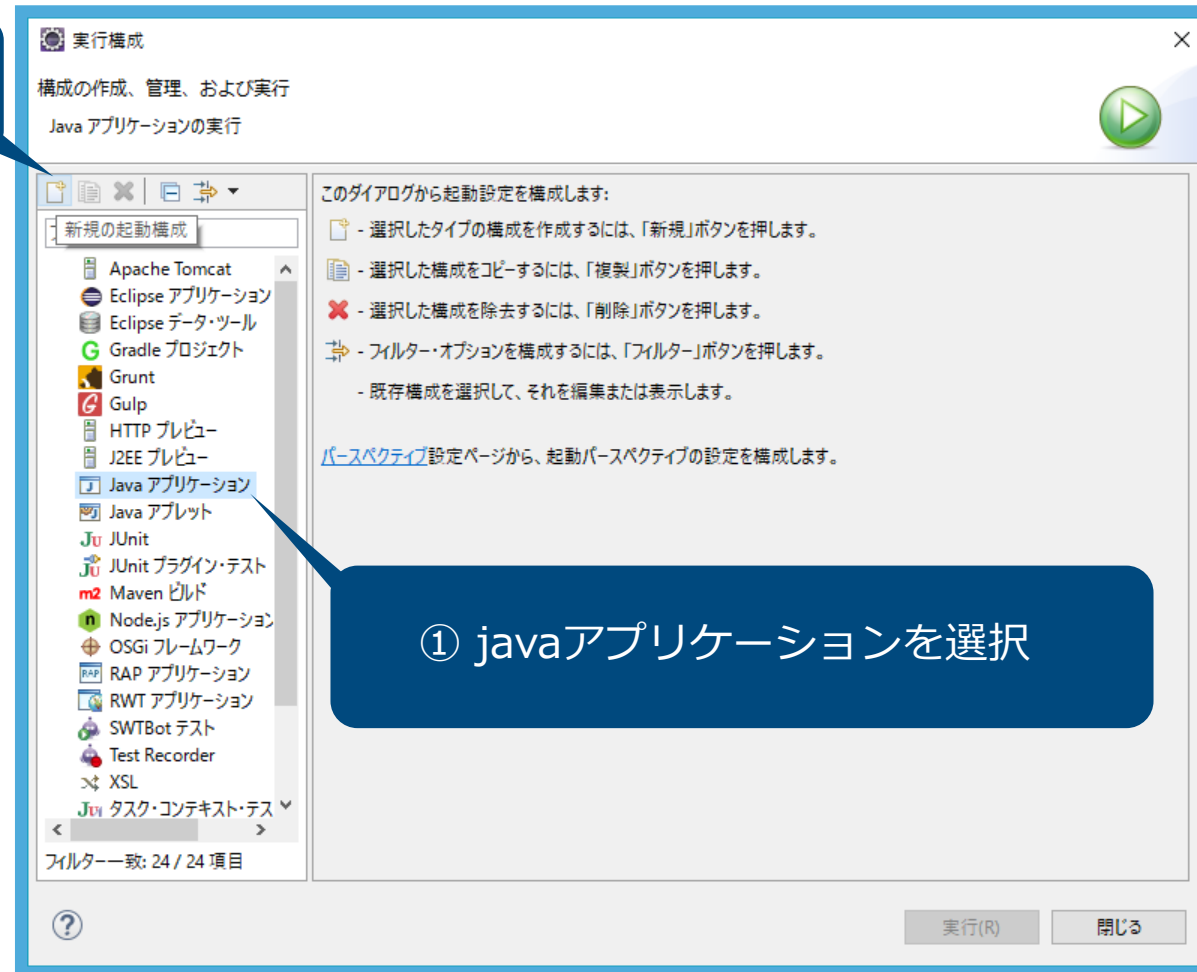


# Javaプログラムの実行

解説

"javaアプリケーション"を選択した状態で、左上の新規作成ボタンをクリックします。

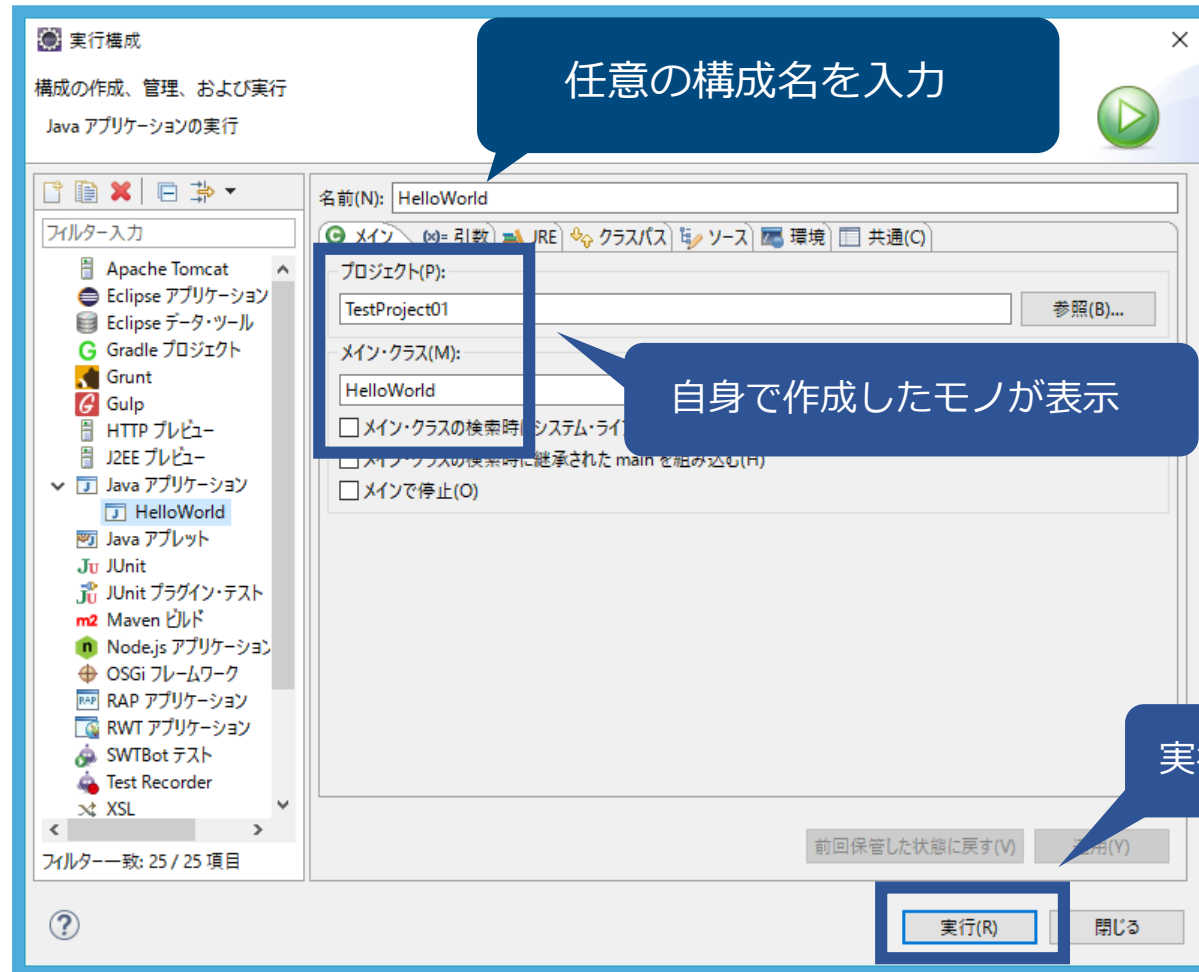
② 新規作成をクリック



# Javaプログラムの実行

## 解説

任意の構成名を入力し、“プロジェクト名”と“メイン・クラス”が自身で作成したものであることを確認します。  
構成名にはあらかじめ自動で入力されていますが、分かりやすいものに変更して問題ありません。



# Javaプログラムの実行

## 解説

"実行"ボタンをクリックすると、実行が始まります。

プログラムに問題がなければ、画面左下にある"コンソール"というタブ内に実行結果が表示されます。

