

A man's profile is shown in a light, faded background. His hair is dark and slightly messy. Overlaid on his head is a circular pattern of small, handwritten-style code snippets in a light gray color. The main title is written in large, bold, black Japanese characters across the center of the image. Below the title, there is a subtitle in smaller black Japanese characters.

# ウズウズカレツジ プログラマーコース

接続の確立  
はじまる!!

## 《データベース接続の概要》

□Javaアプリケーションからデータベース(RDBMS)に接続する場合、`java.sql.DriverManager`クラス(JDBCドライバマネージャ：その名のとおりJDBCドライバを扱うために必要な機能を持ったモジュール)を使用し、以下の手順で行います。

### ①JDBCドライバのロード

`Class.forName`メソッドを使ってドライバをロードします。

### ②接続の確立

`DriverManager`クラスの`getConnection`メソッドを使用してデータベースとの接続に必要な**Connectionオブジェクト**を取得します。

### ③SQL文の送信

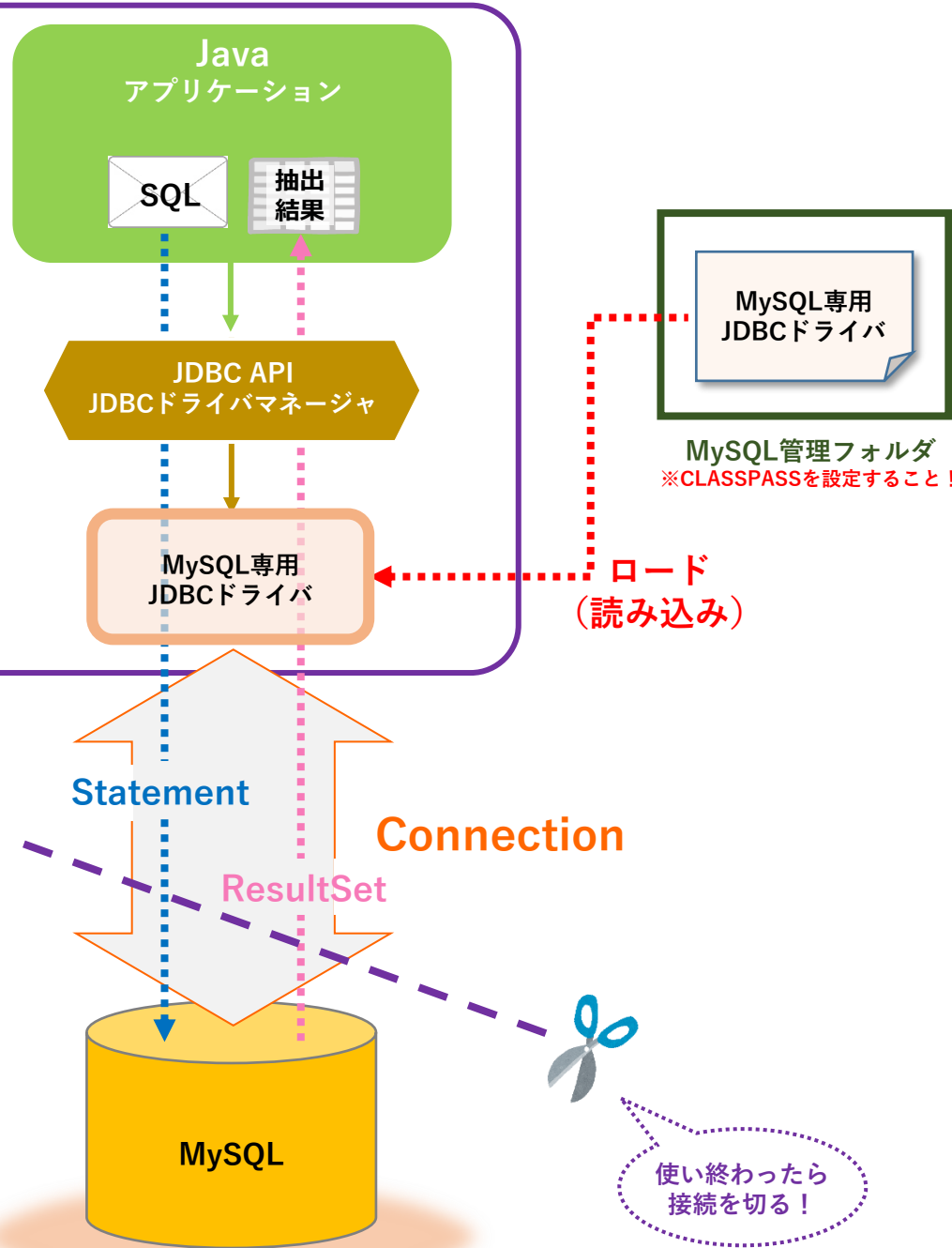
`Statement`系オブジェクトの`executeQuery`メソッドや`executeUpdate`メソッドでSQL文を実行するようRDBMSに依頼します。

### ④抽出結果の取得

`ResultSet`オブジェクトでSQLの実行結果が返ってくるので読み取って処理します。

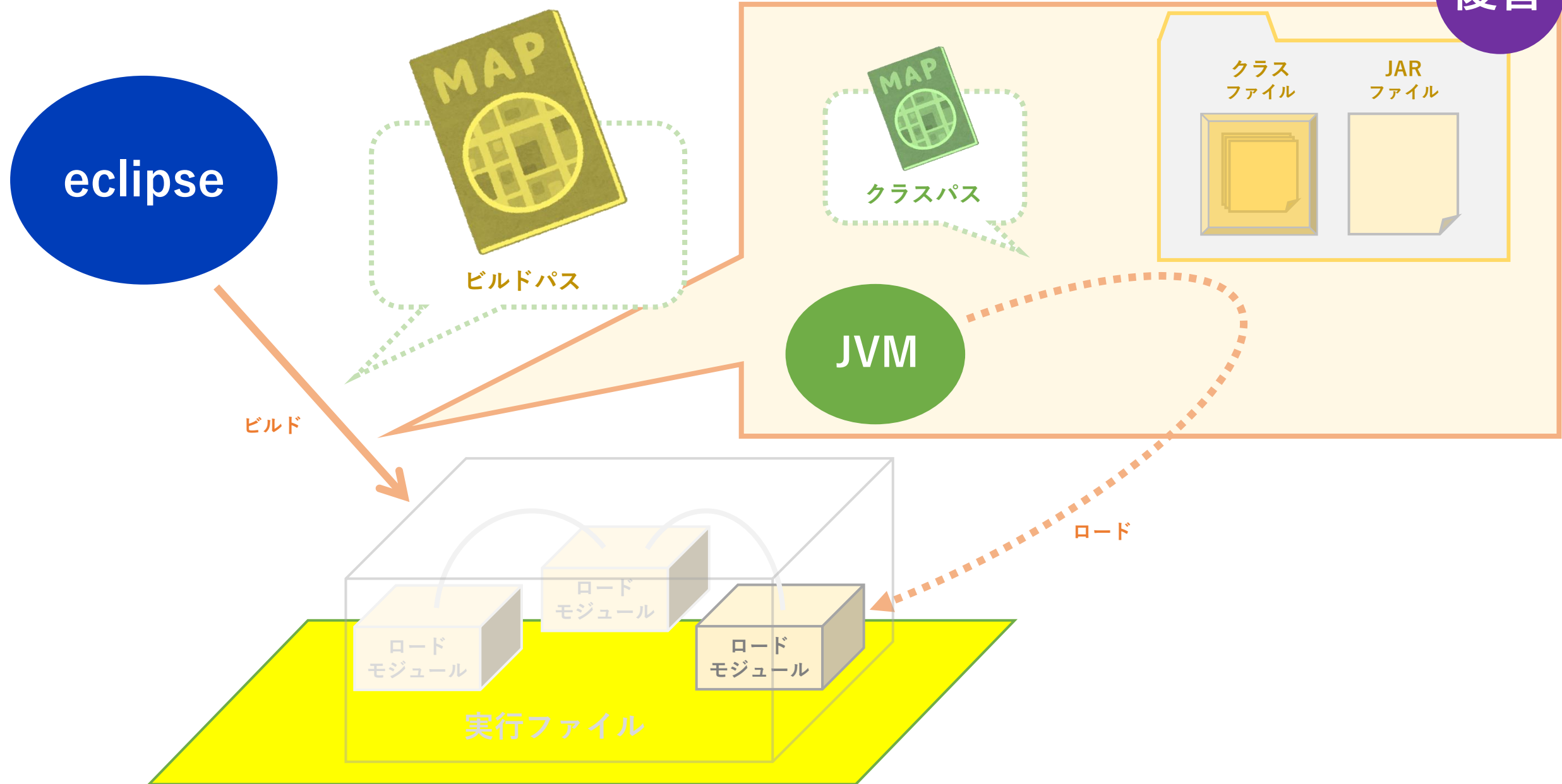
### ⑤接続の解除

データベースへのアクセスが終了したら明示的に接続を解除する必要があります。`Connection`オブジェクト、`Statement`系オブジェクト、`ResultSet`オブジェクト、それぞれで`close`メソッドを起動することで接続を解除できます。



# ～ビルドパスとクラスパスの違い～

復習



# ～データベース接続の5ステップ～

ステップ①  
JDBCドライバのロード

ステップ②  
接続の確立

ステップ③  
SQL文の送信

ステップ④  
抽出結果の取得

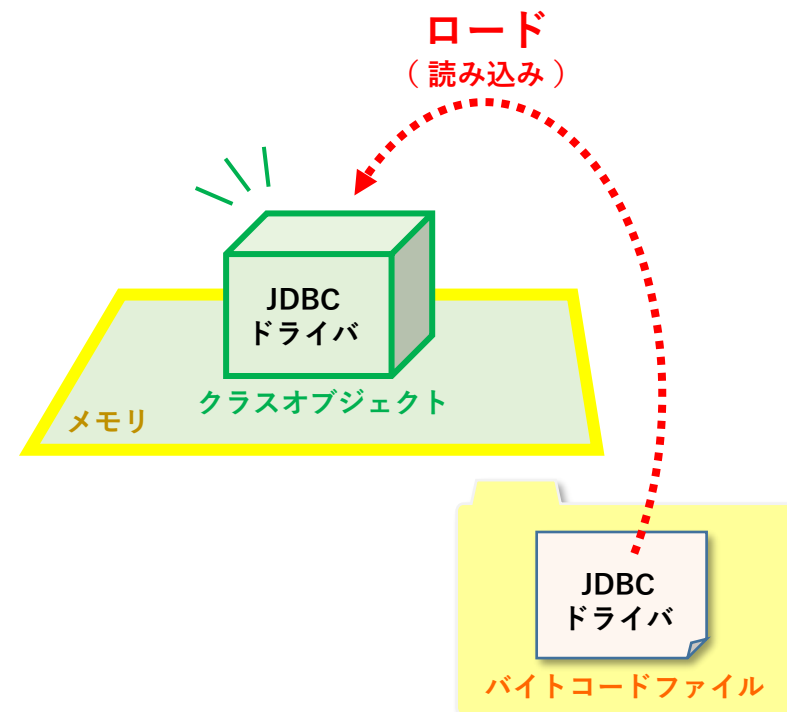
ステップ⑤  
接続の解除

```
//JDBCドライバの相対パス<
//※バージョンによって変わる可能性があります（MySQL5系の場合は「com.mysql.jdbc.Driver」）<
String driverName = "com.mysql.cj.jdbc.Driver";<
```

```
//-----<
//① JDBCドライバのロード<
//-----<
try {<
  ^ Class.forName(driverName); //JDBCドライバをロード&接続先として指定<
} catch (ClassNotFoundException e) {<
  ^ e.printStackTrace();<
}<
```

## Class.forNameメソッド

- Javaのバイトコードファイル（拡張子がclassのファイル）をロードするためのメソッド。
- ロードとはファイルを読み込み、その情報を元にclassオブジェクトを生成、メモリ上に配置することです。
- 引数としてクラスを表す文字列（パッケージ名.クラス名）を指定する必要があります。
- 使用の際はClassNotFoundExceptionでtry～catchする必要があります。



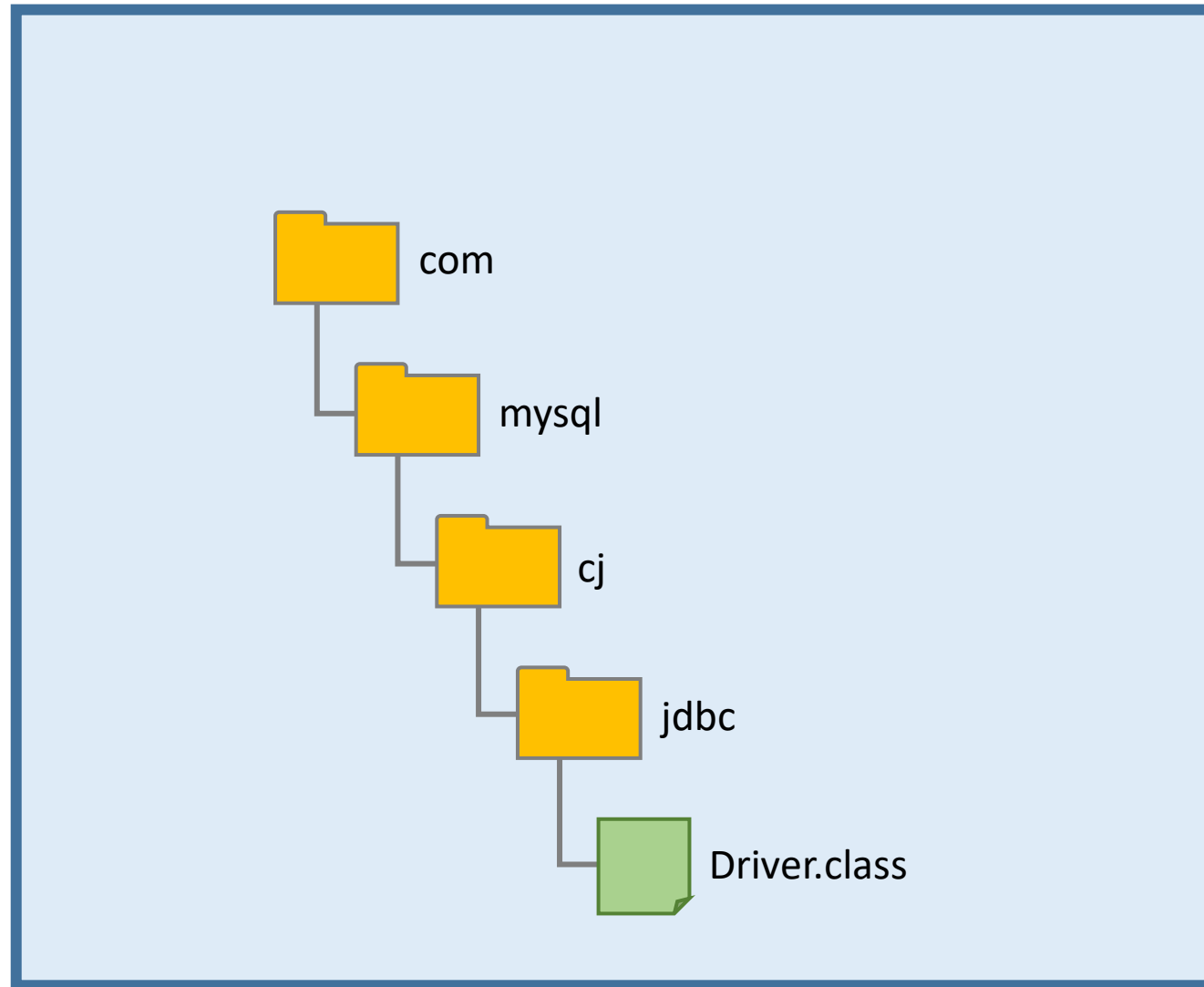
～JARファイル～

mysql-connector-java-8.0.21.jar

ZIPファイル  
(圧縮ファイル)

# ～JARファイル～

mysql-connector-java-8.0.21.jar

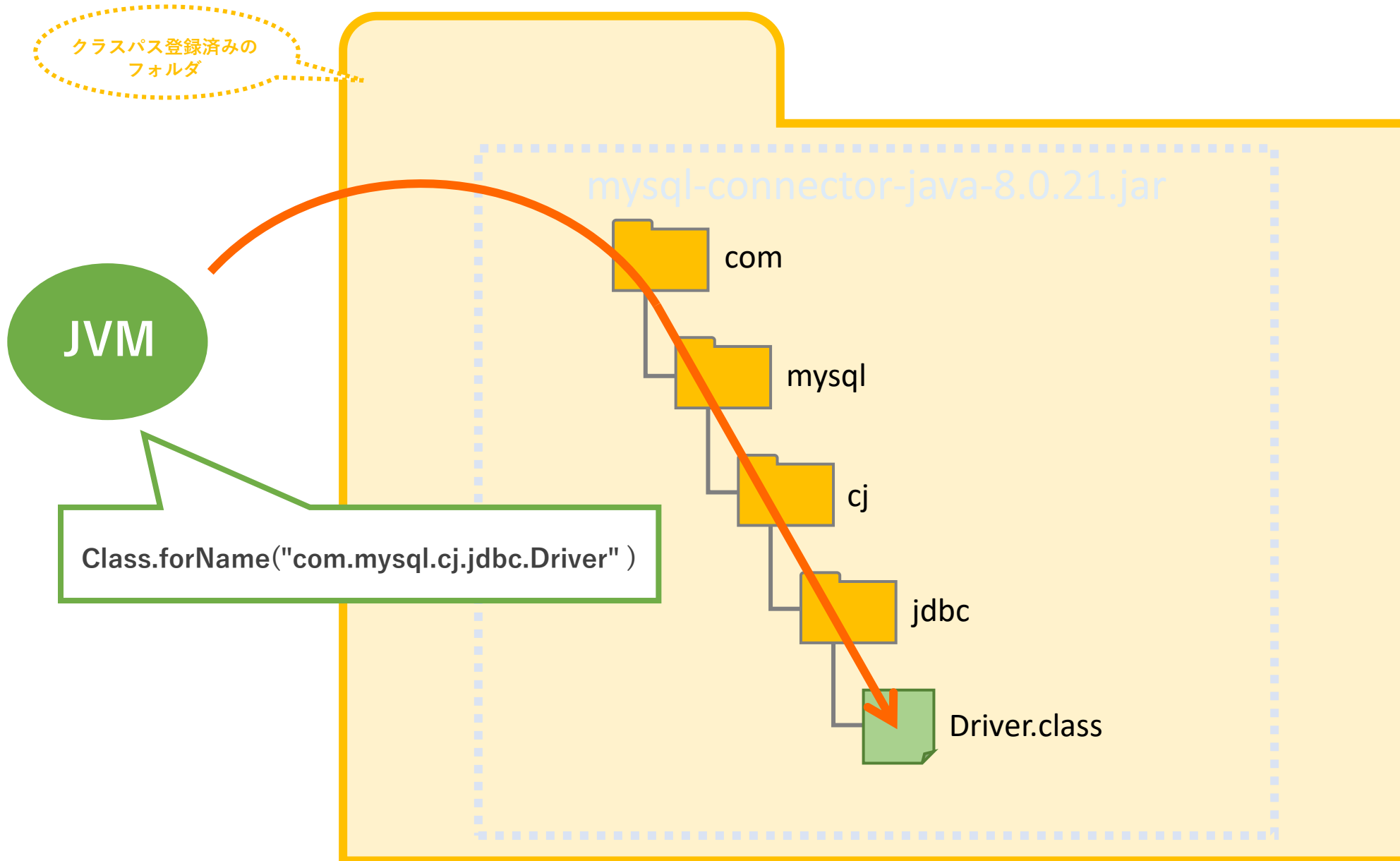


# ～JARファイル～

クラスパス登録済みの  
フォルダ

mysql-connector-java-8.0.21.jar

# ～JARファイル～

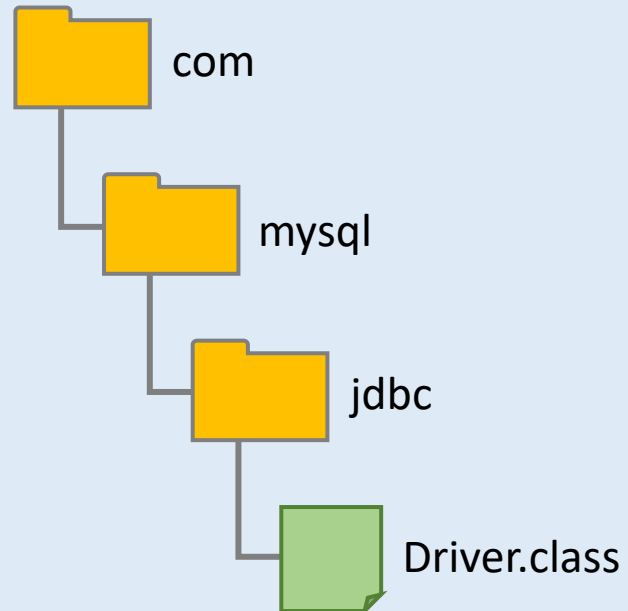




# ～JARファイル～

mysql-connector-java-5.x.xx

※バージョンが5系の場合



## ～データベース接続の5ステップ～

## ステップ② 接続の確立

```
Connection    con = null ;    // Connection (DB接続情報) 格納用変数
```

```
//接続先のデータベース←
//※データベース名が「test_db」でない場合は該当の箇所を変更してください←
String jdbcUrl      = "jdbc:mysql://localhost/test_db?characterEncoding=UTF-8&serverTimezone=JST&useSSL=false";←

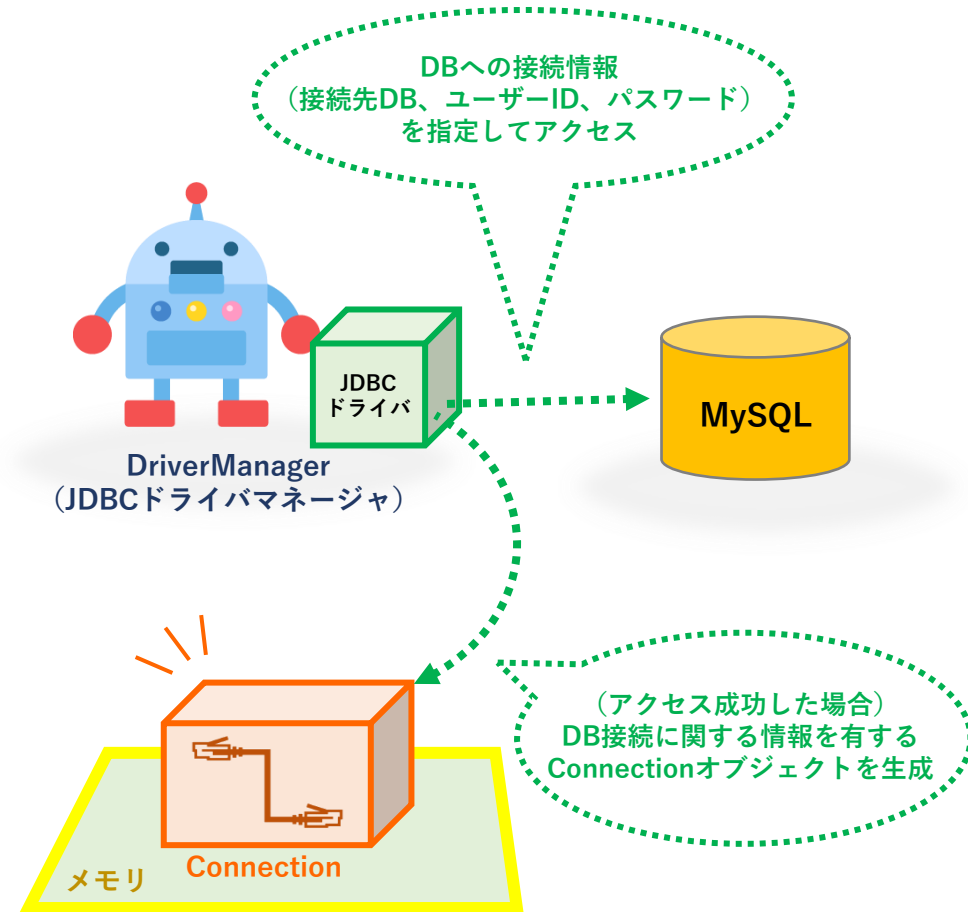
//接続するユーザー名←
//※ユーザー名が「test_user」でない場合は該当の箇所を変更してください←
String userId       = "test_user";←

//接続するユーザーのパスワード←
//※パスワードが「test_pass」でない場合は該当の箇所を変更してください←
String userPass     = "test_pass";←
```

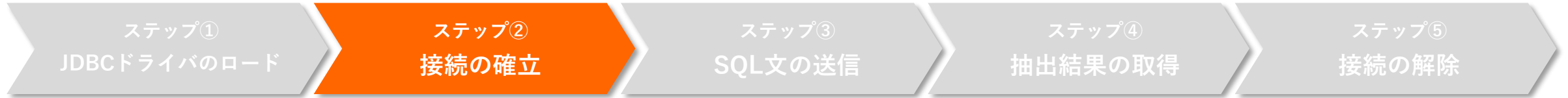
```
//-----␣↵
// ②接続の確立（Connectionオブジェクトの取得）␣↵
//-----␣↵
con = DriverManager.getConnection(jdbcUrl, userId, userPass);␣↵
```

## DriverManagerクラス

- ・ JDBCドライバマネージャのクラス。JDBCドライバにアクセスしてRDBMSを操作するための様々な機能が提供されている。
- ・ getConnectionメソッドでRDBMSに接続し、接続に成功した場合はDB接続に関する情報を有するConnectionオブジェクトを戻り値として返す。引数にDBへの接続情報（接続先DB、ユーザーID、パスワード）を指定することで接続が実行される。



## ～データベース接続の5ステップ～



```
//接続先のデータベース<
//※データベース名が「test_db」でない場合は該当の箇所を変更してください<
String jdbcUrl    = "jdbc:mysql://localhost/test_db?characterEncoding=UTF-8&serverTimezone=JST&useSSL=false";<
```

接続先  
データベース名

通信における  
文字コード

通信における  
標準時刻

暗号化通信  
するかどうか