

# JDBC 理解度テスト②プログラム作成問題(全 3 問:60 点)

テスト範囲： 当スクールの MySQL、Java データベース連携 (JDBC プログラミング) テキスト

実施日： 年 月 日 ( 曜日 )

資料作成者：神田 IT スクール コンテンツ開発部

開始時間： 時 分 終了時間： 時 分

テスト制限時間： 1 時間 30 分

タイピングチェック制限時間： 10 分

氏名：

会社名：

タイピングレベル：

<<テストの流れ>>

**※eclipse を使用して、プログラムを作成してください。**

**※テキストは使ってよい。インターネットは使ってはいけません。**

**※制限時間は厳守してください。**

**※取得した PDF ファイルを「Adobe Acrobat Reader DC」で開いてください。**

Step1 ■配布されたテスト資料が 6 ページで構成されているか確認します。

Step2 ■上記で記入する項目を記入したかどうか確認してください。

☐ 「実施日」を記入しました。

☐ 「開始時間」「終了時間」を記入しました。

☐ 「氏名」を記入しました。

☐ 「会社名」を記入しました。

Step3 ■Eclipse のプロジェクトエクスプローラーのファイルは全てバックアップを取り、削除します。

Step4 ■プロジェクト「jdbc\_test\_kanda.zip」を以下 URL からダウンロードし、解凍・インポートします。

<http://kanda-it-school-square.com/?wpdmdl=4585>

Step5 ■プロジェクトを「jdbc\_test\_名字のローマ字」にリファクタリングしてから問題を解き、プログラムを完成させます。

Step6 ■制限時間は厳守してください。

すべてのプログラムが完成、もしくは制限時間に達したら作業を止めます。

※ただし、残り数行で書き終わる等、目途が立っている場合は最後まで仕上げていただいて構いません。

処理の組み立て方について検討もつかず、分からない時は、潔く諦めて提出するようにしましょう。

Step7 ■「タイピングレベル」の確認をします。

『腕試しレベルチェック』のページを開き、レベルチェックを行って下さい。(制限時間：10 分)

<https://www.e-typing.ne.jp/roma/check/>

Step8 ■レベルチェック実施後、結果が出たら、本紙上部の「タイピングレベル」に結果を記入して下さい。

※ テスト実施の度に、レベルの確認を行います。

★研修期間中、あと 1 回、レベルの確認があります。

今回のレベルを上回るように、継続して日々練習を行ってください。

※ 到達目標レベルは、「A-」以上です。

Step9 ■レベル確認が終わったら、本紙を講師に提出してください。

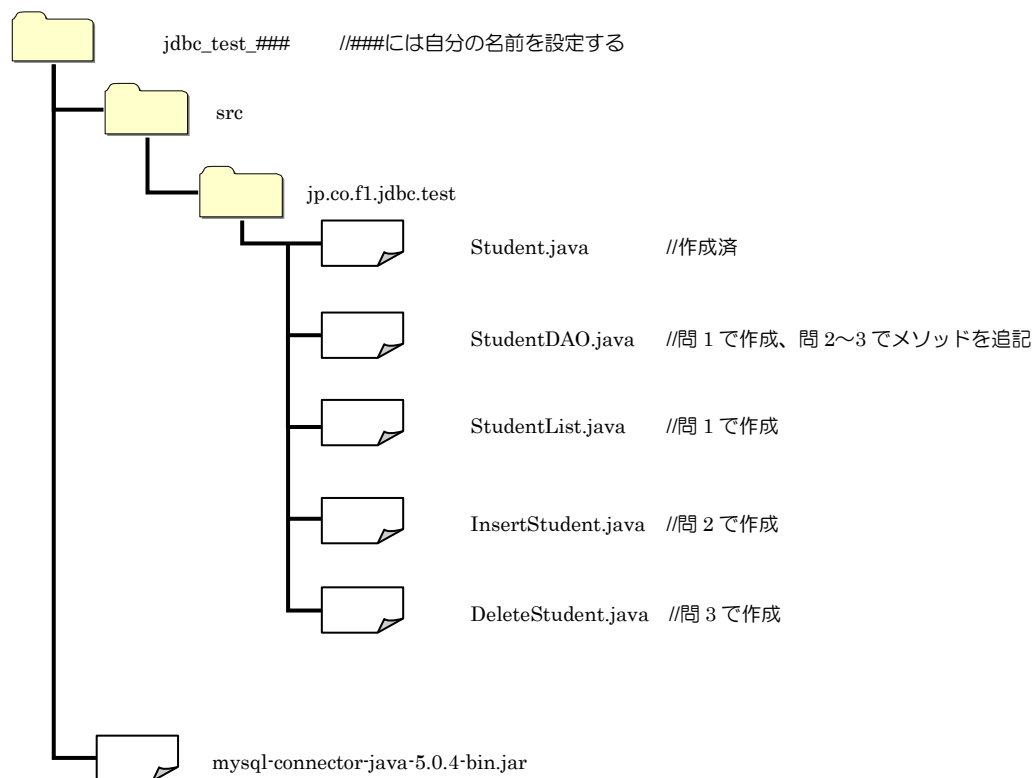
※ 作成したプロジェクトは zip で圧縮し、提出フォームから提出して下さい。

※ 記入内容に漏れがないか確認してから本ファイルを上書き保存し、

プロジェクトと同じフォルダに格納して zip で圧縮し、提出フォームから提出して下さい。

## <<プログラムのフォルダ構成>>

作成するプログラムのフォルダ構成は以下のようになります。



## <<前提>>

JDBC ドライバーは必ず指定の場所に格納し、ビルド・パスを追加してください。

**問1** 以下仕様と実行結果から **StudentDAO.java**・**StudentList.java** を作成しなさい。

#### ソースコード

- ①ソースフォルダ : jdbc\_test\_名字のローマ字/src
- ②パッケージ名 : jp.co.fl.jdbc.test
- ③名前 : StudentDAO、StudentList

※ファイルを作成する階層は p2 を参照してください。

また、JDBC ドライバーがビルド・パスに追加されていることを必ず確認してください。

#### 概要

students テーブルのデータを全件取得し、コンソール上に出力するプログラム。

#### 単元

JDBC プログラミングの準備、JDBC プログラミング、DAO パターン

#### 作成仕様

##### ■DB 情報

- ・MySQL を使用
- ・ユーザー名 : root
- ・パスワード : root123
- ・データベース名 : testdb
- ・テーブル名 : students
- ・下記の SQL 文を実行し、データベースとテーブルを作成してください。

DROP DATABASE testdb;      ※この SQL 文については実行時にエラーが出ても無視してください

```
CREATE DATABASE testdb;
```

```
USE testdb;
```

```
CREATE TABLE students (  
    sid int(11) primary key,  
    name varchar(20),  
    id int(11) unique,  
    score int(3)  
);
```

```
INSERT INTO students (sid, name, id, score) VALUES (1, 'E', 105, 80);
```

```
INSERT INTO students (sid, name, id, score) VALUES (2, 'C', 102, 70);
```

```
INSERT INTO students (sid, name, id, score) VALUES (3, 'A', 101, 90);
```

```
INSERT INTO students (sid, name, id, score) VALUES (4, 'D', 104, 90);
```

```
INSERT INTO students (sid, name, id, score) VALUES (5, 'B', 103, 80);
```

## ■Student クラスの概要

※ダウンロードした `Student.java` ファイルの仕様は以下の通りです

- ・メンバ変数: `sid`、`name`、`id`、`score`    ※カプセル化が適用されています
- ・引数のないコンストラクタ (`id` を 999、`score` を 0 で初期化しています)

## ■StudentDAO クラスの概要

- ・DB 接続用の情報を定数として定義

### @getConnection メソッド

- ・引数    なし
- ・戻り値   `Connection` 型の接続情報
- ・DB へ接続し、戻り値として接続情報を返す。

※テキストを参照して作成をしてください。

### @selectAllStudents メソッド

- ・引数    なし
- ・戻り値   `Student` 型の `ArrayList`
- ・`students` テーブルから `id` の昇順で全件データを取得する
- ・取得したデータは `Student` 型の `ArrayList` に格納し、戻り値として返す。

## ■StudentList クラスの概要

### @main メソッド

- ・実行結果と同じ表示になるように、処理を作成する。

## 実行結果

```
学生 A の id は 101、成績は 90 点
学生 C の id は 102、成績は 70 点
学生 B の id は 103、成績は 80 点
学生 D の id は 104、成績は 90 点
学生 E の id は 105、成績は 80 点
```

問2 以下の仕様と実行結果から **StudentDAO.java**・**InsertStudent.java** を作成しなさい。

### ソースコード

①ソースフォルダ	: jdbc_test_名字のローマ字/src
②パッケージ名	: jp.co.fl.jdbc.test
③名前	: StudentDAO、InsertStudent

### 概要

問1で作成した StudentDAO に insert メソッドを追加して、  
キーボードで入力した情報を students テーブルに登録するプログラム。

### 単元

JDBC プログラミング、DAO パターン

### 作成仕様

#### ■StudentDAO クラスの概要

##### @insert メソッド

- ・引数 Student 型の objStu
- ・戻り値 int 型の count
- ・引数で受け取った objStu の各要素のデータを students テーブルに登録するクエリを発行する。
- ・登録した件数を戻り値として返す。

#### ■InsertStudent クラスの概要

##### @main メソッド

- ・students テーブルに登録する情報をキーボードで入力する  
(全て nextLine()メソッドを使って String 型として入力する)
- ・入力した情報を Student 型のオブジェクトにセットする。この時、必要に応じて int 型にキャストする。
- ・実行結果と同じ表示になるように、処理を作成する。

### 実行結果

※>以降の情報は適宜キーボードで入力すること

```
sid を入力>6
name を入力>F
id を入力>106
score を入力>60
```

```
1 件の登録が完了しました。
```

**問3** 以下の仕様と実行結果から **StudentDAO.java**・**DeleteStudent.java** を作成しなさい。

#### ソースコード

- |          |                            |
|----------|----------------------------|
| ①ソースフォルダ | : jdbc_test_名字のローマ字/src    |
| ②パッケージ名  | : jp.co.fl.jdbc.test       |
| ③名前      | : StudentDAO、DeleteStudent |

#### 概要

問1で作成した StudentDAO に delete メソッドを追加して、  
キーボードで指定した情報に紐付くデータを students テーブルから削除するプログラム。

#### 単元

JDBC プログラミング、DAO パターン

#### 作成仕様

##### ■StudentDAO クラスの概要

###### @delete メソッド

- ・引数 int 型の id
- ・戻り値 int 型の count
- ・引数で受け取った id と同じ id を持つデータを students テーブルから削除するクエリを発行する。
- ・削除した件数を戻り値として返す。

##### ■DeleteStudent クラスの概要

###### @main メソッド

- ・students テーブルから削除する id をキーボードで入力する
- ・実行結果と同じ表示になるように、処理を作成する。

#### 実行結果

※>以降の情報は適宜キーボードで入力すること

削除する id>105

id : 105 のデータを削除しました。

※存在しない id を入力して削除件数が 0 件だった場合は、以下のメッセージが表示されること

削除する id>111

id : 111 のデータは存在しませんでした。

以上、お疲れ様でした。