

# [要件定義書]

## 1. システム名称

成績管理プログラム

## 2. 背景

教科書：「やさしいPython」にかかれている内容のまとめとして、初めてコードを自分で考え、形にするうえでわかりやすい題材を選んだ。

## 3. 目的

生徒の成績を、クラスごとのデータベースに入れ、管理し、入力後の確認・追加・変更・削除を可能とするコードを作成する。

## 4. 対象業務

- モジュールに分割できるものは分割する。
- 学級データ(クラスデータ)を各データベースで管理できるようにする。

## 5. 利用者

学習用のため、自分

## 6. 要求機能

### 0. メニュー

- 「1.データ確認」、「2.データ入力」、「3.データ削除」、「4.平均点グラフ」、「5.終了」の各機能を選択する

#### 1. データ確認

- 入力した学級名が存在する場合、データを一覧で表示

#### 2. データ入力

- 学級名を入力してデータが存在していない場合、学級の.dbを作成し、生徒のデータを入力する

- 学級名を入力してデータが存在している場合、データの追加・変更を行うか選択し、各処理を行う

#### 3. データ削除

- 学級を選択し削除するか、全データを削除するかを選択し、各処理を行う

#### 4. 平均点グラフ作成

- 各学級の平均点を計算し、学年ごとに一覧表とグラフを作成する

## 5.終了

- ・プログラムを終了する

# [基本仕様書]

## 0. メニュー処理仕様 (menu.py)

- ・「1.データ確認」「2.データ入力」「3.データ削除」「4.平均点グラフ」「5.終了」の選択
  - 「1.データ確認」選択時
    - 下記：1.データの確認処理仕様へ
  - 「2.データ入力」選択時
    - 下記：2.データの入力処理仕様へ
  - 「3.データ削除」選択時
    - 下記：3.データの削除処理仕様へ
  - 「4.平均点グラフ」選択時
    - 下記：4.平均点グラフ処理仕様へ
  - 「5.終了」選択時
    - プログラムを終了する

## 1. データ確認処理仕様 (menu.py→DataConfirmation.py)

- ・学級名を入力
  - 学級データが存在する場合
    - データを一覧表で表示
    - メニューに戻る
  - 学級名が存在しない場合
    - メニューに戻る

## 2. データ入力処理仕様 (menu.py→DataEntry.py)

テーブル構造

"CREATE TABLE product(No INT,name CHAR(20),score CHAR(20))"

- ・学級名を入力
  - 学級データが存在していない場合
    - 新しい学級の.db(テーブル)を作成
    - 追加する人数を選択
      - 生徒のデータを入力(No・名前・点数)
      - メニューに戻る
    - 学級データが存在している場合
      - データの追加・変更を行うか選択
      - 追加を選択した場合
        - 追加する人数を選択
          - 生徒のデータを入力(No・名前・点数)
          - メニューに戻る

### 3.データ削除処理仕様 (menu.py→DataDelete.py)

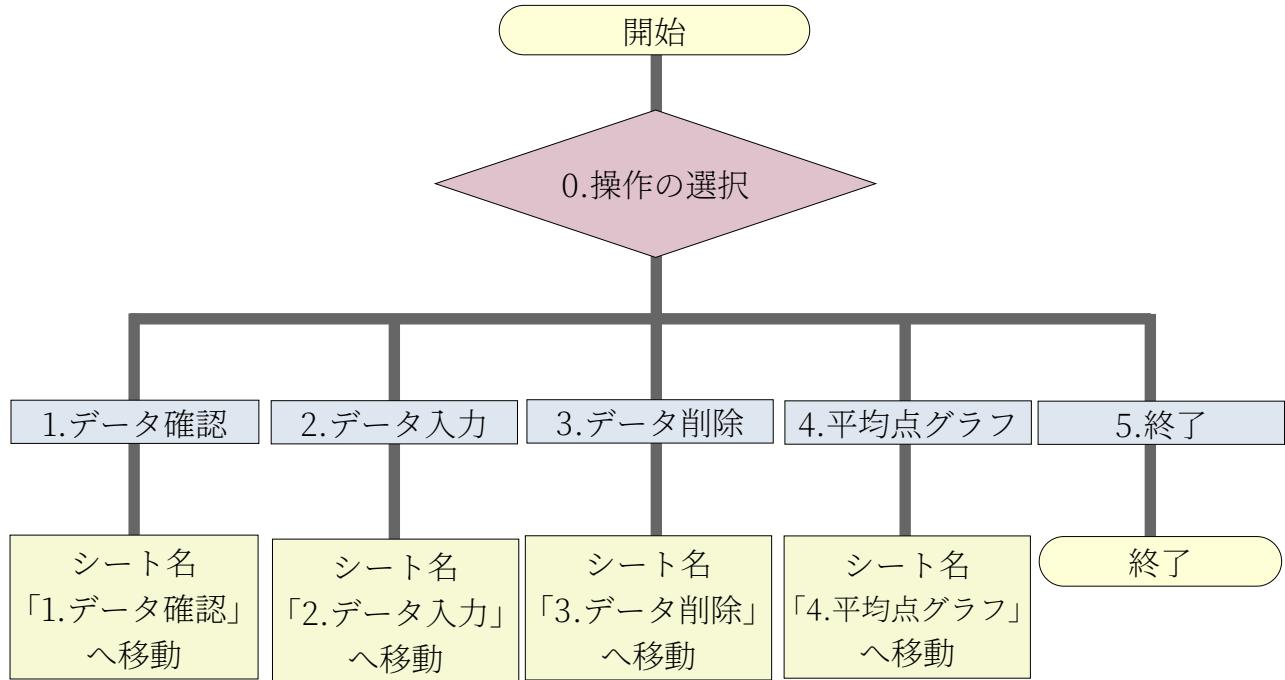
- ・学級を選択し削除するか、全データを削除するかを選択
  - 学級を選択し削除する場合
    - 学級名を選択
    - 削除の最終確認
    - 学級データ削除
    - メニューへ戻る
  - 全データの削除をする場合
    - 削除の最終確認
    - 全データの削除
    - メニューに戻る

### 4.平均点グラフ処理仕様 (menu.py→DataGraph.py)

- ・学年を選択する
  - 対象の学年があるか検索する
    - 対象の学年がある場合グラフを作成する。
    - メニューに戻る
  - 対象の学年がない場合
    - メニューへ戻る

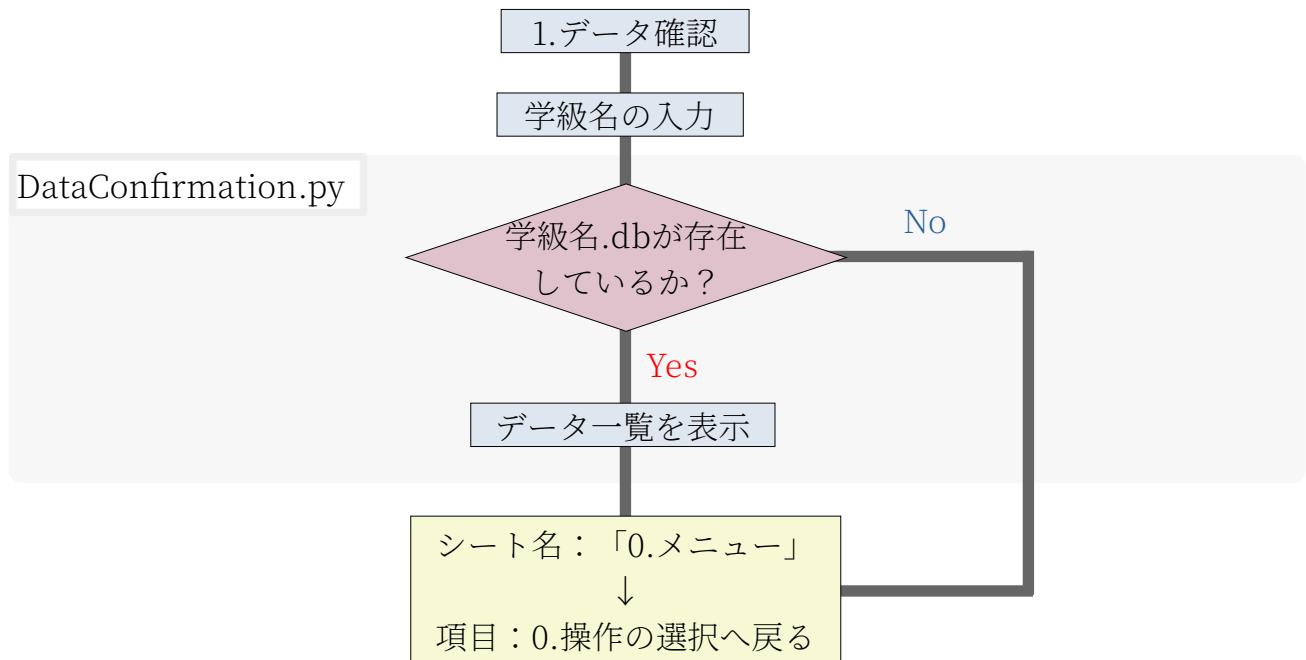
# [処理フロー図\_0]

## 0. メニュー処理仕様 (menu.py)



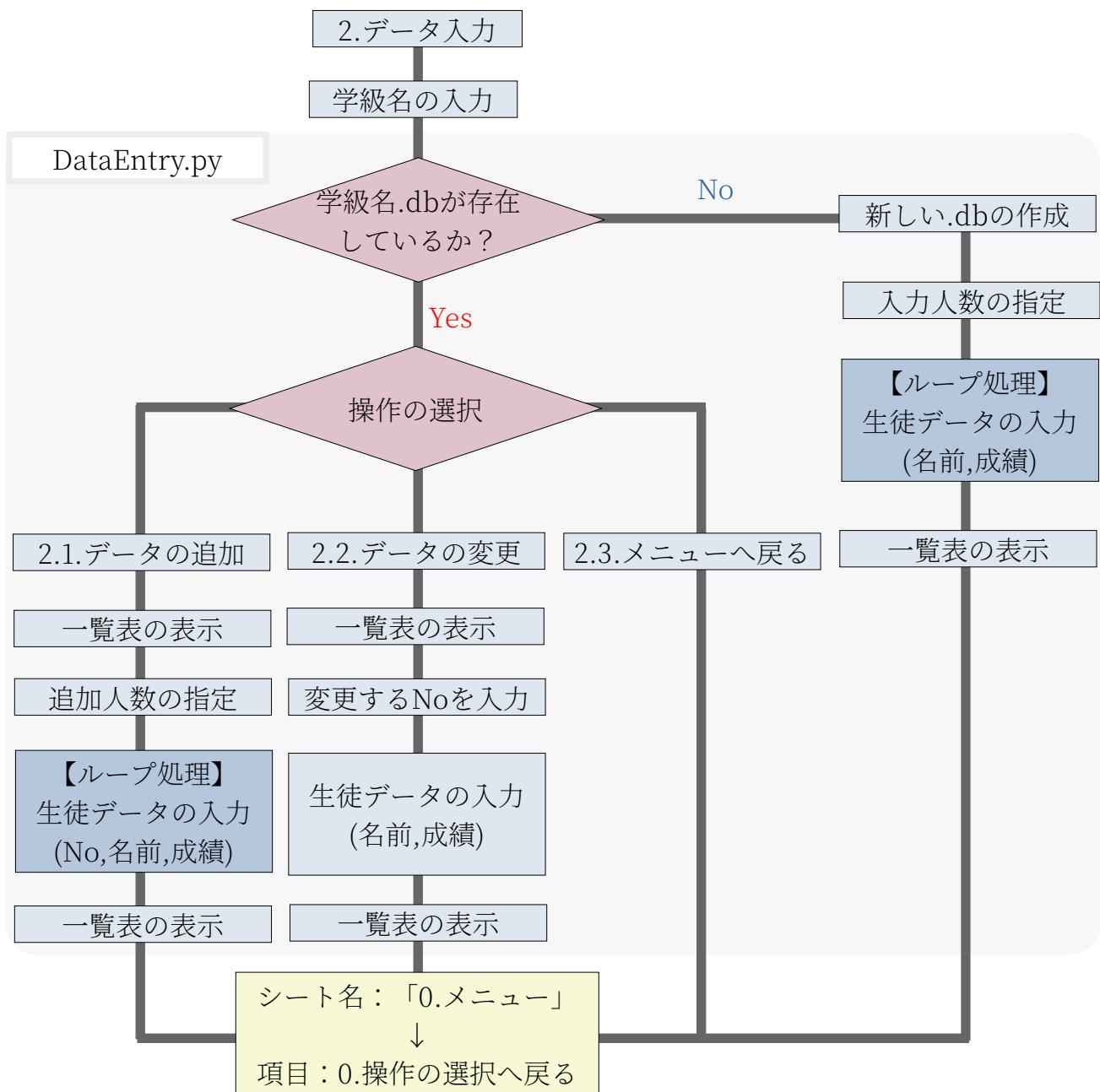
# [処理フロー図\_1]

## 1. データ確認処理仕様 (menu.py→DataConfirmation.py)



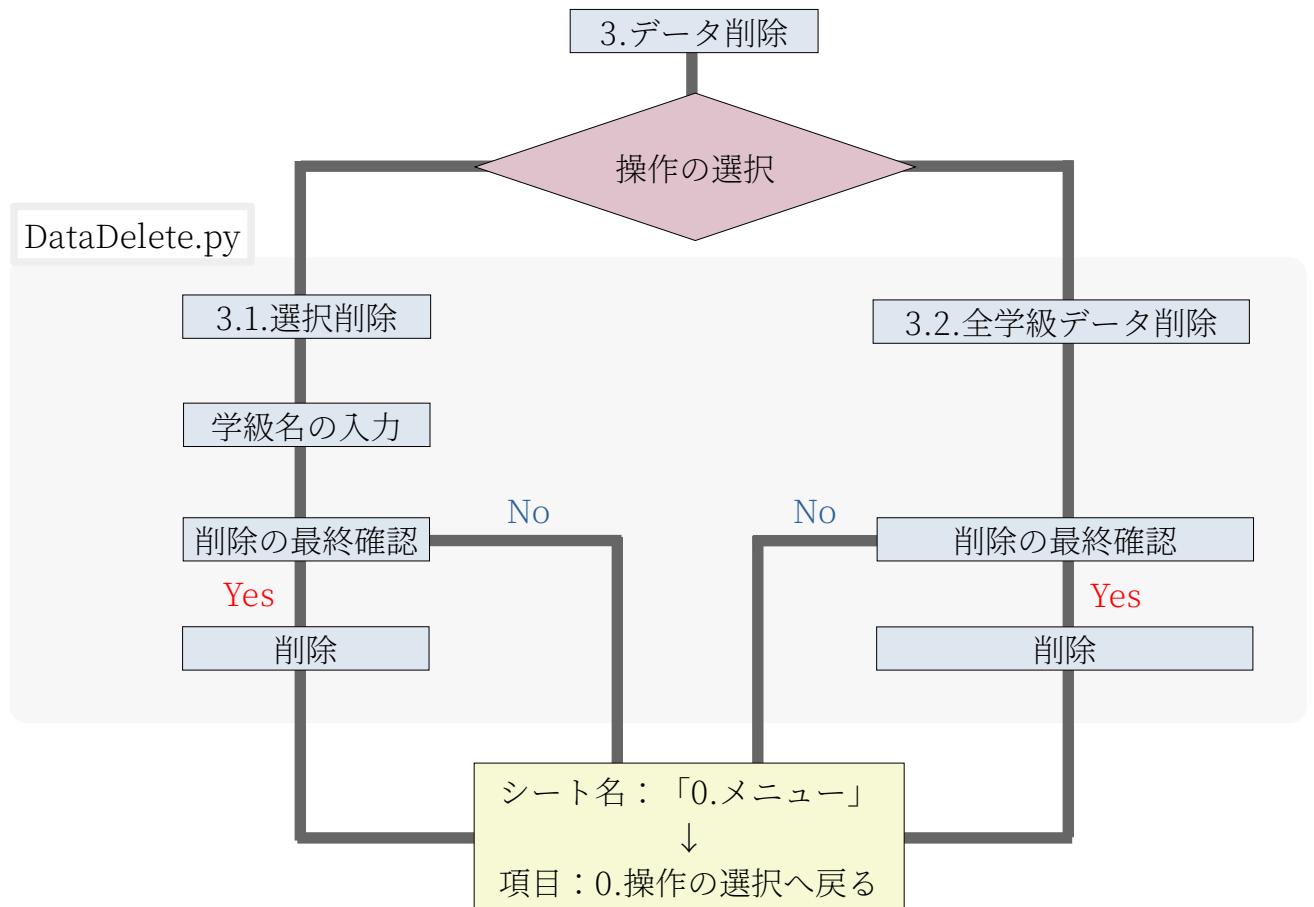
## [処理フロー図\_2]

### 2.データ入力処理仕様 (menu.py→DataEntry.py)



## [処理フロー図\_3]

### 3.データ削除処理仕様 (menu.py→DataDelete.py)



## [処理フロー図\_4]

### 4.平均点グラフ処理仕様 (menu.py→DataGraph.py)

