

3.2 Java コレクション API

演習問題

一. リストの利用

時間: 15 分

- 生徒クラス Student を定義してください。各生徒は名前 (name)、年齢 (age)、学籍番号 (id) を持っています。
- 次のような生徒のデータをリストに格納し、このリストを繰り返し、各生徒の個人情報を出力してください:

名前	年齢	学号
Alice	19	110120
Bob	21	110121
Carol	20	110122
Dave	19	110123

二. データ構造の選択

時間: 20 分

- 問題 1 の生徒を、以下の機能を持つデータ構造に格納したい:
 - 新入生データの追加。
 - 生徒の情報を指定された名前から取得。
- データを保存するために、どのデータ構造を選択しますか?
- 質問 1 の生徒を選んだデータ構造に追加し、`main()` メソッドで上記の機能が満たされるかどうかを確認してください。

三. 重複な文字

時間: 25 分

- メソッド `twiceChar` を実装しなさい。このメソッドは `char` 型の配列を受け入れます。配列の中、2 回以上出現した文字があります。そのような文字を返したい。

Example ✓

`twiceChar(new char[] {'a', 'b', 'c', 'a'})`
を呼び出したら `'a'` が返されます。

- ヒント: `Map` または `Set` を使用してみてください。

四．リストの整列

時間： 15 分

1. 問題 1 のリストを、年齢が小さいものから大きいものまでの順に並べ替えてください。
2. 各生徒の順位と情報を出力してください。

五. スタックの実装

時間: 30 分

- Java に付属する `Stack` クラスを使わずに、以下のスタックの基本機能を提供する `MyStack` クラスを実装してください:
 - プッシュ `push(item)`
 - ポップ `pop()`
 - スタックのサイズを計算 `size()`
- 最初は、整数だけを格納できるようなものから始めてみてください。 `MyStack` クラスを使用して、さまざまな種類のデータを保存したい場合は、どうすればよいですか?
- ヒント: `List` で実装してみましょう。