

## オブジェクト指向プログラミング 第11回

### マウスリスナー List インターフェース

担当：高橋、佐藤聖也

#### サンプルコードリスト

1. **MyMenuFrame02.java**

前回配付した MyMenuFrame02.java と同一。

2. **MyMenuFrame03.java**

MyMenuFrame02.java を継承して、メニューバーにカスケードメニューおよびチェックボックスメニューを追加する。ただし、カスケードメニューとチェックボックスメニューのアクションリスナーを、それぞれに対応する内部クラスを作成して処理する。

3. **MyMouseSample01.java**

MouseMotionAdapter を継承した内部クラスによって時々刻々マウスの位置を読み取り、それに合わせてパネル上に縦横の直線を表示させる。

4. **ArrayListSample01.java**

ArrayList クラスによるリストの作成例 ( 1 )。

- Generics ( < ... > の部分 ) の使用に注意する。
- List インターフェースによるインターフェース型変数 ( List<String> ar = new ArrayList<String>(); ) に注意する。インターフェース型変数とすることで、コンストラクタを変更するだけでデータ構造を変更できるようになる例えば ( List<String> ar = new LinkedList<String>(); とするなど )。

5. **ArrayListSample02.java**

ArrayList クラスによるリストの作成例 ( 2 )。

~~6.~~ **ArrayListSample03.java**

for 文ではなく、Iterator インターフェースを使ってリストの表示を行う。

7. **ArrayListUsage01.java**

ArrayList クラスにおけるメソッドの使用例。

8. **VectorSample01.java**

Vector クラスの使用例。Vector クラスは ArrayList クラスとほとんど同一だが、Vector クラスはマルチスレッドに対応している。

~~9.~~ **StackSample01.java**

Stack クラスの使用例。Stack クラスは、Vector クラスを拡張してスタックを管理できるようにしたクラスである。

10. **LinkedListSample01.java**

LinkedList クラスの使用例。ArrayList クラスや Vector クラスは配列をベースにしているので、要素を追加または削除するたびに配列の一部の要素を移動しなければならない。これに対し、リンクドリストは次の要素への参照を保持するので、リンクの操作によってデータの挿入や削除を行うことができる。

~~11.~~ **FileInformationDisplay02.java**

File クラスを用いた、ファイル情報の取得例。

### 課題 1

MyMouseSample01.java にコードを加筆し，マウスをクリックするごとにパネル上のマウスの位置と，何回目のクリックであるかを標準出力するプログラム

MyMouseSample02\_00\*\*000.java を作成しなさい．ただし，ファイル名の00\*\*000は自分の学籍番号とする．

ヒント：コンストラクタに `addMouseListener(new MyMouseListener());` を追加するとともに，内部クラスとして `MouseAdapter` クラスを継承した `MyMouseListener` クラスを作成し，`mouseClicked` メソッドを実装する．

### 課題 2

上の課題 1 で作成した MyMouseSample02.java にコードを加筆し，マウスをクリックするごとに，クリックした位置を中心とする半径 10 ピクセルの円がパネル上に表示されていくプログラム MyMouseSample03\_00\*\*000.java を作成しなさい．ただし，円座標の保持を保持するために `LinkedList` クラスを用いなさい．また，ファイル名の00\*\*000は自分の学籍番号とする．

#### 提出の際の注意点：

- ・ コンパイルすると実行できる形式で提出すること。
- ・ 学籍番号のフォルダに提出ファイルを全て入れ、そのフォルダをzipファイルにして提出すること。

提出先：<https://tdu.app.box.com/f/027a511786fd4bd2b5a7897dcf4f46e1>

提出期限：7/26 23:59