

# プログラミング演習2 課題 補足資料

---

第5週：連結リスト

# 練習課題5-1

## AddressDataクラス

---

- Point2Dクラス(練習課題2-7)などを参考にすればよい
- 変更点は
  - 問題文に示される通りの4つのフィールド変数を準備する
  - コンストラクタを、フィールド変数を初期化するように変更する
  - printメソッドも、指定されたフォーマットで表示するように変更する

# 練習課題5-2

## Cellクラス

---

- 第3週講義資料 pp. 28-29、教科書p.107を参考に
- ひな型は、p.29に示されている
  - p.29の例では、MyDataクラスをデータとして保持
  - 今回作りたいのは、作成したAddressDataクラスをデータとして保持するクラスなのでそこを変更する

# 練習課題5-3

## Addressクラス

---

- 第3週講義資料 pp. 30-31を参考に
- ひな型は、LinkedListクラスとして示されている
  - p.42も参考に、変数sizeを用意しておくとい

# 練習課題5-4,5-5

insertFirst, deleteFirst, displayListメソッド

---

- 第3週講義資料 pp. 32-37に実装例が示されている
- LinkedListクラスとしての例なので、Addressクラスへの対応のみ行えばOK

# 練習課題5-1,2,3,4,5 動作確認プログラムの例

以下のような  
出力が得られればOK

● 以下のようなプログラムで動作確認をすること

```
Address address = new Address();  
AddressData data = null;  
  
// 連結リストへデータを追加  
data = new AddressData("名前1", "なまえ1", "0001", "name1");  
address.insertFirst(data);  
data = new AddressData("名前2", "なまえ2", "0002", "name2");  
address.insertFirst(data);  
data = new AddressData("名前3", "なまえ3", "0003", "name3");  
address.insertFirst(data);  
  
address.displayList(); // 連結リストの表示  
System.out.println("-----");  
data = address.deleteFirst(); // リスト先頭からデータを削除  
address.displayList(); // 連結リストの表示  
System.out.println("-----");  
data.print(); // AddressDataクラスのprintメソッドの確認と  
// deleteFirstメソッドの戻り値の確認
```

```
Name : 名前3  
Kana : なまえ3  
Phone : 0003  
Mail : name3  
Name : 名前2  
Kana : なまえ2  
Phone : 0002  
Mail : name2  
Name : 名前1  
Kana : なまえ1  
Phone : 0001  
Mail : name1  
-----  
Name : 名前2  
Kana : なまえ2  
Phone : 0002  
Mail : name2  
Name : 名前1  
Kana : なまえ1  
Phone : 0001  
Mail : name1  
-----  
Name : 名前3  
Kana : なまえ3  
Phone : 0003  
Mail : name3
```

# 必須課題5-6

## getAddressDataメソッド

---

- 第3週講義資料 pp. 44-45を参考に
- ひな型はp.45に示されている
  - このひな形を利用する場合には、  
リストのサイズを記録するsize変数の値を  
適切に変更しておくこと(pp.42-43を参考に)
- current変数を、該当する要素を指すように移動している
  - まずheaderを代入して、先頭を指し、  
その後、for文内でcurrent.nextを指していくことで、該当要素へ移動

# 必須課題5-6

## 動作確認プログラムの例

---

- 練習課題5-1,2,3,4,5の動作確認プログラムに以下を追加して確認すること

```
        :  
// ここまで練習課題5-1,2,3,4,5の動作確認プログラム  
  
System.out.println("-- 5-6 --") ;  
data = address.getAddressData(2);  
data.print();
```

以下のような  
出力が得られればOK

```
-- 5-6 --  
Name : 名前1  
Kana : なまえ1  
Phone : 0001  
Mail : name1
```



# 必須課題5-7

## insertメソッド

---

- 第3週講義資料 pp. 46-47を参考に
- 処理の流れは、p.47に示されているので、これにのっとって実装すればよい
  - 先頭にデータを挿入の場合は簡単. insertFirstメソッドを使うだけ
- 先頭以外の場合
  - 1.「挿入する場所の1つ前の要素(A)を見つける」は、必須課題5-6と同様にすればよい
  - 2.「新たな要素(B)を作成」は、新たにCellのインスタンスを作成
  - 3.と4.については、その順序が重要  
(B)のnextが、(A)が元々指していたnext(C)を指すように変更した後、  
(A)のnextが(B)を指すように変更しなければいけない(p.46参考)

# 必須課題5-7

## 動作確認プログラムの例

- 必須課題5-6の動作確認プログラムに以下を追加して確認すること

```
:  
// ここまで必須課題5-6の動作確認プログラム  
  
Boolean result;  
AddressData newdata = null;  
  
System.out.println("-- 5-7 --");  
newdata = new AddressData("名前5","なまえ5","0010","name5");  
result = address.insert(newdata, 2);  
if(result==true)  
    System.out.println("insert 成功");  
else  
    System.out.println("insert 失敗");  
address.displayList(); // 連結リストの表示
```

5 を入力した場合も  
確認しておくこと  
(追加に失敗する)

以下のような  
出力が得られればOK

```
-- 5-7 --  
insert 成功  
Name : 名前2  
Kana : なまえ2  
Phone : 0002  
Mail : name2  
Name : 名前5  
Kana : なまえ5  
Phone : 0010  
Mail : name5  
Name : 名前1  
Kana : なまえ1  
Phone : 0001  
Mail : name1
```

# 必須課題5-8

## deleteメソッド

---

- 第3週講義資料 pp. 48-49を参考に
- 処理の流れは、p.49に示されているので、これにのっとって実装すればよい
  - 先頭のデータを削除の場合は簡単. deleteFirstメソッドを使うだけ
- 先頭以外の場合
  - ステップ1は、必須課題5-6と同様(p-1番目の要素を取得)
  - ステップ2は、p番目の要素を取得(=p-1番目の次)
  - ステップ3と4は、その順序が重要(p.48も参考に)

# 必須課題5-8

## 動作確認プログラムの例

- 必須課題5-7の動作確認プログラムに以下を追加して確認すること

```
:  
// ここまで必須課題5-7の動作確認プログラム
```

```
System.out.println("-- 5-8 --");  
data = address.delete(3);  
address.displayList(); // 削除後リストの表示  
if(data==null)  
    System.out.println("delete 失敗");  
else{  
    System.out.println("delete 成功");  
    data.print();  
}
```

5 を入力した場合も  
確認しておくこと  
(削除に失敗する)

以下のような  
出力が得られればOK

```
-- 5-8 --  
Name : 名前2  
Kana : なまえ2  
Phone : 0002  
Mail : name2  
Name : 名前5  
Kana : なまえ5  
Phone : 0010  
Mail : name5  
delete 成功  
Name : 名前1  
Kana : なまえ1  
Phone : 0001  
Mail : name1
```