

## 総合復習 2

### 問題 1 クラス図からソースプログラムの導出 (提出 DepartmentStore.java)

クラス図 DepartmentStore からソースプログラムを導出しなさい.

DepartmentStore
<u>printName():void</u> <u>getFloor(goods:String):int</u> <u>getGoods(floor:int):String</u>

## 問題2 おみくじ (提出 Omikuji.java)

このプログラムは、おみくじを引いて、その結果を表示するものです。(1)～(3)に従って作成しなさい。

- (1) クラス図 Omikuji からソースプログラムを導出しなさい。

Omikuji
<u>getMessage(name:String):String</u> <u>getMessage(number:int):String</u> <u>getRandomMessage():String</u> <u>consult(name:String):void</u>

- (2) 動作確認用プログラム OmikujiTest.java と OmikujiMain.java を作成しなさい。

実行イメージ

```
>java OmikujiTest
Pro Taro さんの今日の運勢は
大吉
吉です
```

```
1 // 00AJ00 電大太郎
2 public class OmikujiTest{
3     public static void main(String[] args){
4         String welcome = Omikuji.getMessage("Pro Taro");
5         System.out.println(welcome);
6         String fortune = Omikuji.getMessage(0);
7         System.out.println(fortune);
8         String message = Omikuji.getRandomMessage();
9         System.out.println(message);
10    }
11 }
```

### 実行イメージ

```
>java OmikujiMain
Pro Taro さんの今日の運勢は吉です
```

```
1 // 00AJ00 電大太郎
2 public class OmikujiMain{
3     public static void main(String[] args){
4         Omikuji.consult("Pro Taro");
5     }
6 }
```

(3) API仕様 Omikuji を満足するように Omikuji.java を実装しなさい。

API仕様 Omikuji	
getWelcomeMessage	<p>メッセージを返却します。</p> <p>引数 (name:String) は名前です。</p> <p>名前と「さんの今日の運勢は」の文字列を連結し、返却します。</p>
getMessage	<p>おみくじの番号からおみくじ（文字列）に変換し、返却します。</p> <p>おみくじの番号 (number:int) とおみくじの関係は、以下の通りです。</p> <p>0:大吉, 1:中吉, 2:小吉, 3:吉, 4:末吉, 5:凶, 6:大凶</p>
getRandomMessage	<p>おみくじの結果を返却します。</p> <p>乱数によっておみくじの番号（0～6の整数）を生成します。</p> <p>次におみくじの番号からおみくじの文字列（「大吉」等）に変換します。</p> <p>おみくじの文字列と「です」を連結し、返却します。</p>
consult	<p>おみくじを引き、その結果を出力（表示）します。</p> <p>引数 (name:String) は名前です。</p> <p>おみくじの結果を生成します。</p> <p>名前と「さんの今日の運勢は」とおみくじの結果を出力（表示）します。</p>

※ RandomNumber クラス を利用するには以下の記述が必要です。

```
import jp.tdu.util.RandomNumber;
```

### 問題3 あっちむいて・ホイ (提出 Acchimuite.java)

このプログラムは、あっちむいて・ホイのゲームです。(1)～(3)に従って作成しなさい。

- (1) クラス図 Acchimuite からソースプログラムを導出しなさい。

Acchimuite
<pre> getArrow(number:int):String printIntroduction():void getPlayerNumber():int getComputerNumber():int getMessage(player:int,computer:int):String getResult(player:int,computer:int):String hoi():void </pre>

- (2) 動作確認用プログラム AcchimuiteTest.java と AcchimuiteHoi.java を作成しなさい。

実行イメージ

```

>java AcchimuiteTest
0: ↑
1: →
2: ↓
3: ←
番号を入力してください: 0
player: 0
computer: 1
message: You: ↑ vs →: Com

```

```

1 public class AcchimuiteTest{
2     public static void main(String[] args){
3         Acchimuite.printIntroduction();
4         int player = Acchimuite.getPlayerNumber();
5         System.out.println("player:" + player);
6         int computer = Acchimuite.getComputerNumber();
7         System.out.println("computer:" + computer);
8         String message = Acchimuite.getMessage(player, computer);
9         System.out.println("message:" + message);
10    }
11 }

```

### 実行イメージ

```

>java AcchimuiteHoi
0: ↑
1: →
2: ↓
3: ←
番号を入力してください: 2
You: ↓ vs ↓:Com
You Win!!

```

```

1 public class AcchimuiteHoi{
2     public static void main(String[] args){
3         Acchimuite.hoi();
4     }
5 }

```

(3) API仕様 Acchimuite を満足するように Acchimuite.java を実装しなさい。

API仕様 Acchimuite	
getArrow	<p>番号を矢印（文字列）に変換し、返却します。</p> <p>引数 (number:int) は番号です。</p> <p>番号と矢印の関係は、以下の通りです。</p> <p>0 : ↑, 1 : →, 2 : ↓, 3 : ←</p>
printIntroduction	<p>番号と矢印の関係、入力を促すメッセージを出力（表示）します。</p> <p>以下の文字列を出力（表示）します。</p> <p>0 : ↑</p> <p>1 : →</p> <p>2 : ↓</p> <p>3 : ←</p> <p>番号を入力してください:</p>
getPlayerNumber	<p>キーボードからプレイヤーの手（番号 0～3）を入力させます。</p> <p>その番号を返却します。</p>
getComputerNumber	<p>乱数を使用して、番号を返却します。番号は 0～3 の整数です。</p>
getMessage	<p>プレイヤーとコンピュータの矢印等を連結した文字列を作り、返却します。</p> <p>引数 (player:int) はプレイヤーの番号です。</p> <p>引数 (computer:int) はコンピュータの番号です。</p> <p>「You:」, player の矢印, 「vs」, computer の矢印, 「: Com」 の文字列を連結し、返却します。</p>
getResult	<p>勝敗結果を文字列として、返却します。</p> <p>引数 (player:int) はプレイヤーの選んだ番号です。</p> <p>引数 (computer:int) はコンピュータの選んだ番号です。</p> <p>player と computer が一致した場合は、文字列「You Win!!」を返却します。</p> <p>それ以外は、文字列「You Lose!!」を返却します。</p>
hoi	<p>あっちむいて・ホイをします。</p> <p>（他の振る舞いを呼び出して実現します。）</p> <p>(1) 番号と矢印の関係、入力を促すメッセージを出力（表示）します。</p> <p>(2) プレイヤーの番号を入力します。</p> <p>(3) コンピュータの番号を決めます。</p> <p>(4) プレイヤーとコンピュータの選んだ矢印等を出力（表示）します。</p> <p>(5) 勝敗結果を出力（表示）します。</p>

※ KeyboardReader クラス を利用するには以下の記述が必要です.

```
import jp.tdu.util.KeyboardReader;
```

※ RandomNumber クラス を利用するには以下の記述が必要です.

```
import jp.tdu.util.RandomNumber;
```