● JKad23D「3 で割った余り」

入力された整数を 3 で割ったときの余りを表示する処理を、switch 文を使って作成せよ。

リスト１「3 で割った余り（if 文版）」

```
import java.util.Scanner;

public class JKad23D {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("正の整数を入力してください>");
        int n = in.nextInt();
        n %= 3;
        if (n == 0) {
            System.out.println("割り切れました！");
        } else if (n == 1) {
            System.out.println("余りは1です！");
        } else if (n == 2) {
            System.out.println("余りは2です！");
        }
        in.close();
    }
}
```

switch 文に直すこと

課題完成時の画面①

```
正の整数を入力してください>123
割り切れました！
```

課題完成時の画面②

```
正の整数を入力してください>10
余りは1です！
```

課題完成時の画面③

```
正の整数を入力してください>23
余りは2です！
```

● JKad23C「あなたのクラスは？③」（JKad09X の switch 版）

クラス名を選択すると、それぞれのコースを表示する処理を、switch 文を使って作成せよ。仕様は以下の通り。なお、1～3 以外を入力したときは「わからないのね！」と表示すること。

入力	クラス	コース
1	IE	4 年制コース
2	SK	3 年制コース
3	SE	2 年制コース

課題完成時の画面①

```
わたしはスーパーティーチャー、ECC エクセレントよ！
あなたのクラスを教えてね！
（1：IE、2：SK、3：SE、それ以外：わからない>1
4 年制コースなのね！
```

課題完成時の画面②

```
わたしはスーパーティーチャー、ECC エクセレントよ！
あなたのクラスを教えてね！
（1：IE、2：SK、3：SE、それ以外：わからない>4
わからないのね！
```

● JKad23B「春夏秋冬！」

「月」(1～12) を入力するとそれぞれの月に応じて以下のように表示する処理を作成せよ。

月	表示
3、4、5	春です！いい季節ですね！！
6、7、8	夏です！暑いですね！！
9、10、11	秋です！食べ物がおいしいですね！！
12、1、2	冬です！寒いですね！！
それ以外	月は 1 月～12 月までです！

課題完成時の画面①

```
いま何月ですか？（1～12）＞6
夏です！暑いですね！！
```

課題完成時の画面②

```
いま何月ですか？（1～12）＞13
月は 1 月～12 月までです！
```

● JKad23A「今日は何曜日？」

「月」(1～12) と「日」(1～31) を入力すると曜日を表示する処理を作成せよ。なお、1 月 1 日は土曜日、2 月は 28 日とする。また範囲外の値の入力（例えば「月」に 13 を入力）は想定しないものとする。

ヒント：入力された日付が 1 月 1 日から何日目（1 月 1 日が 1 日目）にあたるのかがわかれば曜日もわかる。

課題完成時の画面①

```
「月」を入力してください>1
「日」を入力してください>1
1 月 1 日は土曜日です！
```

課題完成時の画面②

```
「月」を入力してください>6
「日」を入力してください>30
6 月 30 日は木曜日です！
```

● JKad23S「じゃんけん（switch 版）」

「あなた」と「コンピュータ（CPU）」がじゃんけんをする処理を作成せよ。なお、勝敗判定は switch 文を使って行うこと。

課題完成時の画面①

```
じゃんけんをします！
何の手を出しますか？（0：グー、1：チョキ、2：パー>0
あなたはグーを出した！
ＣＰＵはチョキを出した！
あなたの勝ち！
```

課題完成時の画面②

```
じゃんけんをします！
何の手を出しますか？（0：グー、1：チョキ、2：パー>1
あなたはチョキを出した！
ＣＰＵはチョキを出した！
あいこだ！
```

課題完成時の画面③

```
じゃんけんをします！
何の手を出しますか？（0：グー、1：チョキ、2：パー>2
あなたはパーを出した！
ＣＰＵはチョキを出した！
あなたの負け！
```

● JKad23X「ストップウォッチ！」

ストップウォッチの状態遷移を作成せよ。状態は「停止中（STOP)」「計測中（TIME)」「一時停止（PAUSE)」の３つとし、操作は「1：スタート/ストップボタン」「2：クリアボタン」の２つとする。状態遷移表は以下の通り。

状態	1：スタート/ストップ	2：クリア
停止中（STOP) ※初期状態	「動き出しました」と表示して 計測中（TIME）へ遷移する	「その操作は無効です」 「ストップウォッチは止まっています」と表示する
計測中（TIME) ※時間は実際には計測しない	「一時停止します」と表示して 一時停止中（PAUSE）へ遷移する	「その操作は無効です」 「時間を計測中です」と表示する
一時停止（PAUSE)	「計測を再開します」と表示して 計測中（TIME）へ遷移する	「計測をストップしてタイムをクリアします」 と表示して、停止中（STOP）へ遷移する

なお、時間は実際には計測せず、状態遷移のみを作成すること。また、マイナスの値が入力されたときはプログラムを終了するものとし、これら以外の値が入力されたときは現在の状態を維持するものとする。

課題完成時の画面

```
どうしますか？（1：スタート/ストップ、2：クリア）>1
動き出しました！                                          ← 停止中から計測中へ

どうしますか？（1：スタート/ストップ、2：クリア）>1
一時停止します！                                          ← 計測中から一時停止へ

どうしますか？（1：スタート/ストップ、2：クリア）>1
計測を再開します！                                        ← 一時停止から計測中へ

どうしますか？（1：スタート/ストップ、2：クリア）>2
その操作は無効です！                                      ← 計測中なので「2：クリア」は無効
時間を計測中です！

どうしますか？（1：スタート/ストップ、2：クリア）>1
一時停止します！                                          ← 計測中から一時停止へ

どうしますか？（1：スタート/ストップ、2：クリア）>2
計測をストップしてタイムをクリアします！                  ← 一時停止から停止中へ
                                                            タイムは実際には計測していないので
                                                            「クリアします」は表示のみ

どうしますか？（1：スタート/ストップ、2：クリア）>2
その操作は無効です！                                      ← 停止中なので「2：クリア」は無効
ストップウォッチは止まっています！

どうしますか？（1：スタート/ストップ、2：クリア）>3      ← イレギュラーな値は無効
ストップウォッチは止まっています！                          （現在の状態を維持）

どうしますか？（1：スタート/ストップ、2：クリア）>-1     ← マイナスの値で終了
終了します！
```