

● JKad03D 「エスケープシーケンス」 (JKad03D0.java のコードをコピーして作成)

課題フォルダにファイル「JKad03D0.java」(リスト 1) が準備されている。これを実行すると **JKad03D0 実行時の画面 (Before)** のように表示される。以下の指示に従って**課題完成時の画面 (After)** のように表示されるようにプログラムを作成せよ。なお、ファイルは JKad03D.java を作成し、そこに JKad03D0 の main メソッドをコピーして修正すること。

リスト 1: 「エスケープシーケンス」 (ファイル「JKad03D0.java」)

```
public class JKad03D0 {  
    public static void main(String[] args) {  
        // こんにちは!  
        System.out.println("こんにちは! 今日よい天気ですね!");  
        // これから"Java 言語"を学習します!  
        System.out.println("これから Java 言語を学習します!");  
        // 変数 a の値の表示  
        int a = 5;  
        System.out.println("変数 a の値:");  
        System.out.println(a);  
        // 変数 b の値の表示  
        int b = 10;  
        System.out.println("変数 b の値:");  
        System.out.println(b);  
        // a+b の結果表示  
        System.out.println("a+b の結果:");  
        System.out.println(a + b);  
    }  
}
```

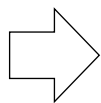
「こんにちは!」と「今日よい天気ですね!」との間に改行を入れる。ただし println を 2 つに分けてはいけない。

「Java 言語」をダブルクォーテーション (") で囲む。

変数 a、b、a+b の値を改行せずに表示する

JKad03D0 実行時の画面 (Before)

```
こんにちは! 今日よい天気ですね!  
これから Java 言語を学習します!  
変数 a の値:  
5  
変数 b の値:  
10  
a+b の結果:  
15
```



課題完成時の画面 (After)

```
こんにちは!  
今日よい天気ですね!  
これから"Java 言語"を学習します!  
変数 a の値: 5  
変数 b の値: 10  
a+b の結果: 15
```

● JKad02C1 「算術演算の短縮表現」(JKad02B.java のコードをコピーして作成)

課題フォルダに JKad02B 「四則演算、余りは 3」の完成版相当のプログラム(ファイル「JKad02B.java」)が準備されている。四則演算と余りの演算を短縮表現に変更せよ。

課題完成時の画面 (JKad02B と同じ)

```
変数 x の値を表示します！
5
変数 x に 14 を足します！
19
変数 x から 3 を引きます！
16
変数 x に 7 をかけます！
112
変数 x を 10 で割ります！
11
変数 x を 4 で割った余りを求めます！
3
```

● JKad03C2 「インクリメントとデクリメント」

リスト 1 のコードを入力・実行してインクリメントとデクリメントの動作確認をせよ。

リスト 1: インクリメントとデクリメント (ファイル「JKad03C2.java」)

```
public class JKad03C2 {
    public static void main(String[] args) {
        int i = 0;
        System.out.println("現在の i の値" + i);
        System.out.println();
        // インクリメント
        System.out.println("i++で表示" + (i++));
        System.out.println("現在の i の値" + i);
        System.out.println();
        System.out.println("++i で表示" + (++i));
        System.out.println("現在の i の値" + i);
        System.out.println();

        // デクリメント
        System.out.println("i--で表示" + (i--));
        System.out.println("現在の i の値" + i);
        System.out.println();
        System.out.println("--i で表示" + (--i));
        System.out.println("現在の i の値" + i);
        System.out.println();
    }
}
```

課題完成時の画面

現在の i の値	0
i++で表示	0
現在の i の値	1
++i で表示	2
現在の i の値	2
i--で表示	2
現在の i の値	1
--i で表示	0
現在の i の値	0

● JKad03B 「ようこそ！ECC コーヒーへ」

あこがれの「ECC コーヒー」で食事をした！食事内容は以下の通り。支払金額を計算する処理を作成せよ。なお、可能な限り算術演算の短縮表現を使うこと。

食事内容（値段は税抜き、これに 10%の消費税が加算される）

- ・ コーヒー 290 円
- ・ ドーナツ 250 円
- ・ チキンサラダ 380 円
- ・ チーズケーキ 420 円

課題完成時の画面

```
public class JKad03B {
    public static void main(String[] args) {
        final int COFFEE = 290;    // コーヒーの値段
        final int DONUT = 250;     // ドーナツの値段
        final int SALAD = 380;     // サラダの値段
        final int CAKE = 420;      // ケーキの値段
        必要に応じて変数宣言を追加する。

        System.out.println("ようこそ！ECC コーヒーへ");
        System.out.println("こちらでお召し上がりですね！");
        System.out.println("———");
        各メニューの単価と小計を表示する処理を作成する。
        System.out.println("———");
        消費税と支払金額（税込み）を計算して表示する。
        System.out.println("ありがとうございました！");
    }
}
```

変数宣言の前に final を付けると後から値を変更できなくなる（つまり定数）。一般に定数の名前は大文字で記述する。

課題完成時の画面

```
ようこそ！ECC コーヒーへ
こちらでお召し上がりですね！
———
コーヒー      290 円   小計：290 円
さくらドーナツ 250 円   小計：540 円
チキンサラダ   380 円   小計：920 円
チーズケーキ   420 円   小計：1340 円
———
消費税が 134 円で支払金額は 1474 円になります！
ありがとうございました！
```

各メニューの金額と小計を表示。最後に消費税（10%）を計算して加算する。

● JKad03A「2の段を表示しよう！」

かけ算の「2の段」を表示する処理を作成せよ。なお、仕様は以下の通り。

- 変数*i*をかける数（かけ算の後ろ側の数値）、変数*j*をかけられる数（かけ算の前側の数値）とする。

```
int i;           // かける数（かけ算の後ろ側の数値）
int j;           // かけられる数（かけ算の前側の数値）
```

- 使用できる `System.out.println` はひとつのみとする。
- 表示する数値の前にはタブを入れる。

課題完成時の画面（各数値の前にタブを入れる）

2	4	6	8	10	12	14	16	18
---	---	---	---	----	----	----	----	----

● JKad03S「九九を表示しよう！」

JKad03A を参考に九九を表示する処理を作成せよ。

課題完成時の画面

1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81

● JKad03X「九九を表示しよう！（メソッド編）」（教科書 P.112～P.114）

以下のメソッドを作成し、このメソッドを使って九九を表示する処理を作成せよ。教科書 P.112～P.114、および課題フォルダに準備されているファイル「JKad03X0.java」を参考にすること。

書式	処理
<code>public static void printMulTable(int j)</code>	引数 <code>j</code> で指定された段を表示する。

課題完成時の画面

(JKad03S と同じ)