

第 51 回検定(平成 26 年 9 月 28 日) 全商情報プロ部門 Java 1 級解説

【 7 】

概要

ShopInfo をたくさんインスタンス化し、売上データはそのインスタンスに持たせて最後に並べ替え。線形探索や挿入ソートが出現する。オーバーロードも出てきており味わい深い。オブジェクト指向を使ったオーソドックスな問題。

詳細

最初の `while` で店舗データを読み込んで `ShopInfo` クラスをインスタンス化。これで `ShopInfo` クラスのインスタンスが沢山できる。そのインスタンスの参照を配列 `shop` に格納。

次の `while` で売上データを読み込んで `ShopInfo` クラスのインスタンスにデータを集計。インスタンス自身が変数を持っているので、そこにどンドンプラスしていく。この辺りが「ザ・オブジェクト指向」となっている楽しい箇所。

最後の `while` はプログラムの実際の動作のメイン部分です。と、このように `while` がそれぞれ何をしているのか理解すると、物事がシンプルになります。こうやって概略をつかんだ上で、細かい内容を見ていくことが勉強全般のコツでもありますよ。

(1) がある場所は、メソッドの名称のままゲッターメソッドです。しかも `int getCode()` という風に「`int`」の変数を返すことになっています。こうやって「`void`」ではなく「`int`」や「`String`」と付いているメソッドには、必ず `return` が必要です。さらにゲッターメソッドなので、名称そのままの変数を返すわけです。ゲッターメソッドということが理解できていれば簡単に解けますね。

(2) は難しい。まずはここで並べ替えをしている、ということに気づくことが第 1 条件。それは `sort` というメソッド名から気づきやすいかな。第 2 条件として並べ替えには「バブルソート、選択ソート、挿入ソートがある」と知っていること。そしてその特徴のどれに合うか考えるのだ。そうすると (2) の外側の `for` の継続条件が長ったらしくて、これは挿入ソートだと気づきます。ということで、挿入場所までずらしていく処理が答えになる。これは挿入ソート自体を理解していないと解けません。とてもこの文章だけでは説明できないので、わからない人は自分で挿入ソートの基本を学びなおして下さい。

(3) はサービス問題です。`while` の中の命令たちをぼんやりと眺めると解けるのではないのでしょうか。1 級の `Java` は難しいので、こういう問題は確実に点をゲットするんだ。

(4) がオブジェクトオブジェクトしているオブジェクト指向的なワクワクする問題。ただ、(4) の 2 行上を見てみると、なんとなく答えが推測できる。できる、けれどもここはオーバーロードしているメソッドを呼び出すので難しい問題です。

`ShopInfo` クラスの `calcTotal` メソッドは 2 つあります。1 つは引数が 1 つのメソッド、もう 1 つは引数が 2 つのメソッドです。このように、引数の数に応じて呼び出すメソッドが自動的に変わる仕組み

みをオーバーロードと言います。オーバーライドではなくてオーバーロードですよ。似ているけれど間違わないように。

(4) は `discount == 0` の `else` で呼び出されるので、つまりディスカウントしているということ。その場合は、割引金額を計算しないといけない、、、ということは割引金額を引数として渡さないといけない。計算するためには金額を渡さないと、メソッド側は計算できませんからね。この「引数として渡す」という考え方、メソッドに「渡す」という考え方、これは重要ですので理解ませう。

(5) も迷うところです。でも、よくよく考えるとソートしていないんですよね。これに気づく人は最初に概略を把握している人でしょう。「ソートしなくちゃいけないのに、よくよく考えたらソートしていない。こりゃー、(5) はソートしかないっしょ」と気づいて5点ゲットです。ただ、`sort` メソッドは `int number` で引数を受け取っています。ということは `sort` メソッドさんに、何かしらの値を渡してあげないといけません。大切なことなので何度も言いますが、この、メソッドに引数を「渡す」という考え方を十分理解して下さい。

※処理条件には、「`ShopInfo` クラスをインスタンス化し、配列 `shop` に記憶する。」と書かれているが、厳密には配列 `shop` には、`ShopInfo` クラスのインスタンスの場所を指し示す「参照」が記憶されている。インスタンスそのものが記憶されているわけではない。この「参照」ということを意識することは、今後ステップアップしたい人には重要なのだ。

解答

(1)	<code>return code</code>
(2)	<code>shop[m + 1] = shop[m]</code>
(3)	<code>index = index + 1</code> (別解 <code>index++</code>)
(4)	<code>shop[j].calcTotal(sales, discount)</code>
(5)	<code>sort(number)</code>