第 54 回検定(H28.01.31)全商情処プロ部門 Java 2 級　解説

【 ７ 】

概 要

大問 1 題 (小問 5 題) で出題されることとなった 2 回目の検定問題。オーソドックスなオブジェクト指向の問題で、なおかつ問題となっている箇所の難易度が低い。支店ごとに Siten クラスのオブジェクトを生成する。そのオブジェクトに配列を持たせて集計して表示する、というプログラム。

詳 細

処理内容もプログラムも問題も非常に簡単な問題である。解き方は以下の通りです。

1　まずは、いつも通りクラスとメソッドをチェックすること。Siten クラスがあって、その中にコンストラクタがあって、syukei メソッドと hyoji メソッドがあって…と。目立たせるためにクラス名やメソッド名を枠で囲ったり、線を引いたりしよう。

2　次からは処理の流れを追っかけよう。main メソッド内でインスタンス化している箇所を見つけよう。Siten クラスを new している所です。ここでオブジェクトが作られるので、プログラムを読み解く上でヒントになります。えっと、main メソッドの 3 行目の new…ではなくて、5 行目の new がインスタンス化ですね！3 行目は配列生成の new ですよ。インスタンス化と違うよ！同じ new ですが間違わないように。

3　インスタンス化されると、処理がコンストラクタに飛ぶので Siten クラスのコンストラクタを見てみよう。コンストラクタはクラス名と同じ、public Siten(String tenmei) の部分だ。

4　次は main メソッド内のインスタンスメソッドをチェックしよう。インスタンスメソッドは、この解説の 1 でチェックした通り、syukei メソッドと hyoji メソッドの 2 つですね。メソッドを使用する際は「.」(ドット) で区切られているので見つけやすい。「○○○○.syukei(○○○○)」と「○○○○.hyoji()」の場所を見つけてチェック！ここで処理が飛ぶからね。ちなみに「○○○○.hyoji()」は問題文中に存在しません、、、ってことは、この時点で(5)の答えがわかってしまいました^^

処理の流れをまとめると、①main メソッドが実行される。②new Siten(tenmei[p]) の箇所でクラス Siten のコンストラクタ public Siten(String tenmei) が実行される。③main メソッドに戻って st[scode].syukei(ucode, ukin) の箇所で public void syukei(○○○○) が実行。④最後の while 文の中で hyoji() が実行！　そして main メソッド終了！ちょっと次のページで 1 問ずつ詳しく見ていきましょう。

（１） while の中の継続条件です。(１)の条件を満たしている間は繰り返す、ということです。なお、この while 文は 3 行しかありません。これは典型的な線形探索、ラインサーチです。線形探索は「A と B が同じになるまで繰り返す」、つまり「A と B が違う間は繰り返す」というものです。A や B が何なのか、ということは処理条件 5 に書いてあります。線形探索を理解していれば簡単ですよ。

（２） 左辺の変数は、その名の通り sokei。つまり総計です。総計には売上金額をプラスします。

（３） for の中身のこの部分は、変数の宣言と初期値の設定です。main メソッドの最初の方にも for 文があるので、この場所が何を問うているかは分りやすいですね。答えとなる使われている変数も(3)の右側を見たら分るでしょう。

（４） printf 関数の中なので、(4)には変数が入ります。ここの内容が printf 関数で表示されます。(4)は un[n]と heikin の間にあり 2 番目に表示される項目です。実行結果を見ると 2 番目は「売上金額計」です。売上金額を合計している変数（or 配列）を考えるとわかりますね。

（５） 前のページでも書いていますが(5)の前に「.」(ドット)があります。ドットがある、ということは(5)にはメソッド名が入ります。メソッドは「オブジェクト名.メソッド名」というルールで記述します。

※この問題は（１）～（５）全て比較的簡単です。しかし、オブジェクト指向そのものが難しいので、こういう比較的簡単な問題でクラスやメソッド等の理解を深めましょう。この問題を通して単に問題が解けるようになるだけでなく、そもそもインスタンス化とは何か、コンストラクタとは何か、等を理解しましょう。よくわからない場所は積極的に先生に尋ねましょう。

解答

（１）	uc[m] != ucode
（２）	sokei + ukin
（３）	n = 0
（４）	syu[n]
（５）	hyoji()