

2. 難しいか？易しいか？

一口に「易しい」と言い切るのは憚られる。

Railsの基礎を理解していれば解ける問題になっているが、幅広い範囲からポツポツと出題される感じなので、バッチリ満点が取れるかというと、そうでもない。

イチから体系的に学んでいれば点はとりやすいと思うが、つまみ食いしながら速習した人には少しつらいかもしれない。

3. Railsの細かい仕組みを問う問題群

最も印象的だったのは、Railsの細かい仕組みを頭で覚えておく必要がある、ということだ。

vimなどでちょいちょいってコードスニペットを呼んできて、頭で覚えることを放棄して楽しんでいるプログラマは痛い目を見るだろう（いうまでもなくわたしのことだ）

細かい部分が出るから、一生懸命に勉強しても点は上がらないかもしれない。出題傾向を把握した上で、あとは自分の経験と頭だけが頼り。

ちなみに、出題傾向はITトレメの問題集で確認できる。ただし実際の試験は、もう少しハイレベルな問題・選択肢なので、これが十分に解けたからといって高をくくらないこと。

4. Rspecの問題が「出ない」

RubyやRailsではおなじみのユニットテストに関する問題も出る。

...出るには出るのだが、test/unitがベースになっており、人気（少なくともわたしのまわりでは）のRspecは出題範囲外である。

test/unitもRspecも基本は似通っているとはいえ、Rspec派にはあまりやさしくない。

5. 結局のところ

ActiveRecordのリレーションの仕組みや、データベースのテーブル・カラムとの対応、ルーティングによるRESTfulな設計、ActiveSupportに含まれているメソッドに関する内容は必ず押さえるべき。ただ、このあたりは普段からRailsを書いていると自然と身につくものだ。

なので、出題傾向を掴んだあとは、やはりスキルのみが生きてくるような問題となっていると思う。