最速でJavaエンジニアになって稼ぐ

2018/10/17

人生逃げ切りオンラインサロン内

はじめに

学ぼうとする人とそうでない人の差は大きいです。
学ぼうとしているだけで値千金なのです。
レベルが違うのです。上位の人間なのです。
というぐらいの自信を持って取り組んでいただければと思います。

勉学においては「根拠のない自信」こそが最も重要です。
どれだけ教える側に知識や技術があっても、それがないと教えても無駄になるからです。

「勉強しに来てるけど、私は天才」ってぐらいが丁度良いと思います。

あ、でも「自信」がある事と「謙虚さ」は共存出来るので、そこは気を付けて。

目次的なやつ

- 講師紹介
- そもそも何するの？
- 最速は何をもって最速なの？
- 何故Java……(RubyとかPHPやれよ)
- カリキュラムの簡単な説明
- 開発環境を用意してみる
- HelloWorld実装してみる

講師紹介

- Javaプログラマー。
- 実は元インフラ屋さん。
- 最近は殆どリーダーかPM(プロジェクトマネージャー)。
- 月単価は100万以上。現在は130万。
- PHPとかJSとかAWSとかも出来る。やりたいかどうかは別。
- プログラミング技術は中の上ぐらい。
- でも実際はこう。上>>>>>>>>>>>中>下。辛い。
- りゅうけんさんの元現場リーダー。
- ゲームがめっちゃ好きで、ゲームする為に仕事してる感ある。

そもそも何するの？

2019年1月末までに
Javaエンジニアになる。

どうやったらなれるの？

友情・努力・勝利


友情
勝利

- 「エンジニア」というものを理解する。
- 理解する為のステップを踏んでいく。
- ステップを踏む中で「技術」を習得する。
- 「技術」は別に「プログラミング」の事ではない。
- 「エンジニア」を理解し、「技術」を学び、それらを組み合わせた「成果」の「見える化」をする。

「エンジニア」というものを
理解する。

- 「エンジニア」は「プログラムが書ける人」の事ではない。
- 「エンジニア」を辞書通りに引くと「機械・電気などの技師。更に広く、工学者や技術者」になる。
- 我々は「技術者」になろうとしている。しかも「ITシステム」のという枕詞付。
- 「ITシステム技術者」→「システムエンジニア(SE)」的な。
- 色々割愛すると、SEに求められている「技術」は「ITシステムを依頼した人の期待通りに完成させる技術」
- 「期待通り」がミソ。「最初に決めたことや依頼した通り」に完成させる必要は全くない。
- ここらへんは「期待値コントロール」とか言われたりもする。

つまり「エンジニア(SE)」の中に「プログラマー」や「インフラエンジニア」とかがいて、その人たち全員の目指しているところは「ITシステムを依頼した人の期待通りに完成させる」事。

理解する為のステップ

- 「技術」の内訳を知る。まずは知識から。
- その技術が一体何に使われるもので、それの何がシステムを期待通りに完成させるのに必要なのか。
- プログラムは何に必要？　インフラは？　本当にそれだけ？
- お客さんと話をする。やる事を決める。やらない事を決める。決める為の段取りは？　プレゼン手法？　会話？　全部「技術」
- 技術は特化する事も出来る。「分業」という方法もある。1人で全部やってもいい。しかし素人にはオススメできない茨の道(フルスタックエンジニア)
- 特化した時に名乗るのが「Javaプログラマー」とか「インフラエンジニア」
- 特化してるからといって「それしかできない」わけではないのがややこしい。

プログラムが書けない
「エンジニア」なんてたくさんいる。

でも「プログラム」という「技術」について何も知らない「エンジニア」は「エンジニア」ではない。

だって目的は「依頼した人の期待通りにシステムを完成させる事」だから。
しかもそれを「最速で仕上げ、最高品質で、最低のコスト」でやるのが目標だから。

極端な話、「プログラム」を書かなくても「依頼した人の期待通りに完成させる事」が可能なら、コストのかかる「プログラム」なんてなくてもいい。より良い方法を提案すれば良い。

それをする為に「技術」が必要。
「技術」の内訳や「知識」が必要。

技術とは

- 「プログラミング」や「設計力」「デザイン」「プランニング」など様々。
- どの「技術」が「好き」なのかも人によって様々。
- なので「私はエンジニアです」って言われても、何してる人かはわからない。
- しかし「Javaエンジニアです」って言われても、よくわからない。むずい。
- 「プログラミングが得意で設計も好きなJavaエンジニアです」って言われても、わかるようでわからない。むずすぎる。
- 「技術」っていうのは大抵の能力を指すことが出来るので、かなり抽象的で、しかも人によって習熟度が異なるので「Yes or No」みたいな判断も出来ない。
- だから「見える化」が大事。何が出来て、何ができないかの「見える化」

見える化って何？

- 自分が「出来る事」と「出来ない事」を表明する。
- 特に「出来ない事」が重要。
- 「数値」や「画像」で出すと非常に効果的。
- 実際、サロン内でも年収とか月収とか出す方が色々伝わりやすい。
- では「技術」の「見える化」ってどうする？
- 一番簡単なのは「作ったモノ」を公開すること。
- Webサービス、業務システム、ちょっとしたツール、botなどなど
- 「作ったモノ」があるのとないのとでは天と地の差がある。
- じゃぁ見せられるようになんか適当に作ってしまおうという話。

最速は何をもって最速なの？

稼ぎ始めるまでの速度。
プログラミングスクールでは
「プログラミング」が主になっちゃう。
ここでは「技術」を学んで「エンジニア」になって、しかも就
職支援もやれる範囲でやるよ、という話なので

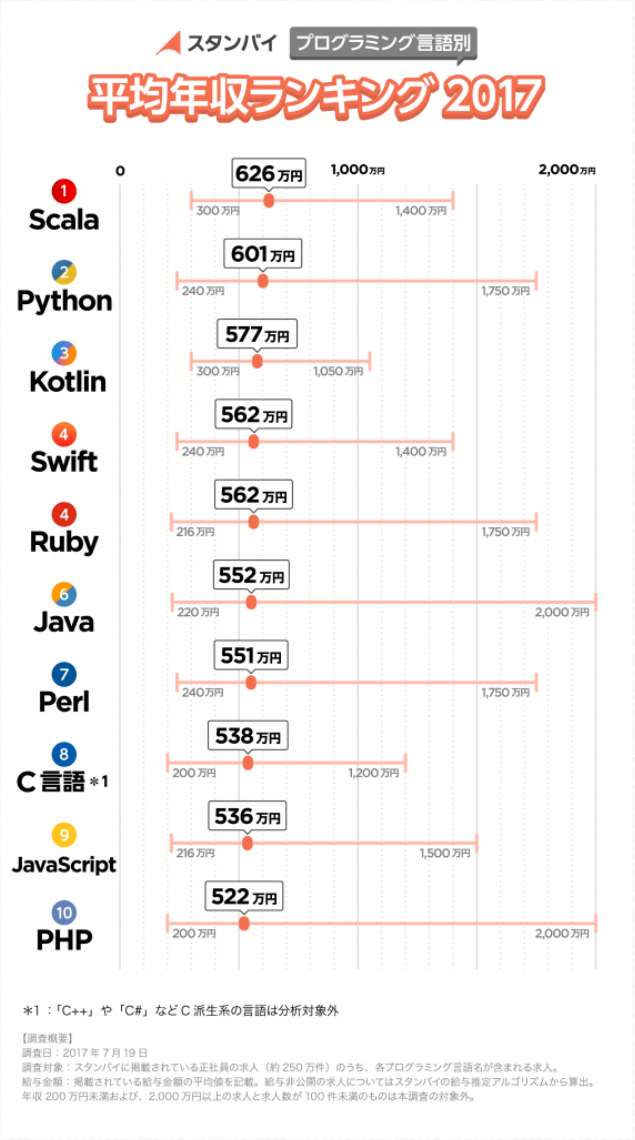
「最速」

多分

何故Java

- お堅い「技術」の1つだから。「Java」に拘りは全くない。
- 技術は時間の流れと共にどんどん変わっていくので、常に勉強し続ける必要がある。勉強をやめた瞬間に「技術者」ではなくなってしまうという問題もあるので、なるべく楽する為に、お堅い技術を学ぶ。
- お堅いからって別に勉強しなくて良いわけではない。でも大分楽。
- 昔からあるので、仕事の量もやっぱりそれなりに多い。
- Java(JVM)をベースに考えられた新しい言語っていうのも沢山ある。
- JavaはクソだけどJVM(Javaが動いている元みたいなもの)は最高っていう風潮もある。
- 後は単純に私がJava以外は教えられるレベルにない。
- 静的型付け言語最高。動的型付けとか無理。

ちなみに言語別平均年収



1位のScalaはJVMを使った言語
3位のKotlinもJVMを使った言語

つまりトップ3に2つもJavaが入ったようなもん
（大きな語弊がある）

見てほしい一つのポイントとしてはJavaの最高値。
2,000万とかいる。
PHPも凄い。ちなみにJavaよりPHPの方が技術者多いので、どうしても平均は低くなる傾向にあるけど、それでも高い水準を維持してるのでPHP熱い。

やっぱりPHPとかRubyがいい

迫さんのSkill Hacksやろう

PHPは知らない

カリキュラムの簡単な説明

1〜4週目まで

- 基本的な開発手順や開発ツールの使い方に慣れてもらう。
- Javaのインストールから開始。
- ありがちな「テキストエディタ」で動かす、みたいなことはしない。
- 細かいプログラムの挙動もそこまで追わない。良く使う構文については説明。
- コードはがんがん書くけど、書く内容の細かい部分は自力で調べる。
- 勿論、講師に聞く、っていうのも「自力」の一部。
- なので、実際にコードを書きながら学習していく。座学 → 実装 ではなく
  実装 → 座学ぐらいの勢い。まずは書く。まずは作る。ブログとかと一緒。
- 簡単な業務を模したアプリケーションを作る。別に細かい機能は作らない。

5〜8週目まで

- 引き続きアプリケーションの開発。
- 自分が作ったコードを「テスト」するという事を学ぶ。
- テストのやり方、自動テストの手法や考え方。
- 何故テストをするのか。テストがない時の苦しみを味わう。
- リファクタリングについても学ぶ。
- これは自動テストがないとぶっちゃけやるのは難しいので、テスト作ってから。
- 端的に言うと、作ったアプリケーションを「より良く」「品質を高める」ような事をがっつりやる。

9〜12週目まで

- 作ったテストをいちいち自分で実行しなくても出来るようにする。
- CI/CDと呼ばれるような技術を学ぶ。
- ここらへんからコードを書くよりも、画面見ながら色々設定したりする。
- AWSも使う。基本的な使い方がわかってるとだいぶ違う。
- 開発手法についてもここで学ぶ。効率の良い開発の仕方とは。
- 画面から動かすようなアプリケーションにするので、画面操作を自動化したテストとかも作る。ここらへんはやってみるとプチ感動があるかもしれない。
- 見えるってのは本当に大事。

13〜14週目まで

作ったアプリをリリースして終わり！

開発環境を用意してみる

1. Javaをダウンロードしてインストール
2. Eclipseをインストール

終わり

それだけではわからぬ

調べてみよう！
それでもわからなかったら……

講師に聞く

or

https://techacademy.jp/magazine/9039

既に存在しているものがあるなら使う。
大事。

HelloWorld実装してみる

開発環境と一緒でまずは調べる。
わからなければ、講師に聞く。

https://qiita.com/FukuharaYohei/items/bb7b31283efe8730d2fa

いんたーねっとってべんり

- HelloWorldの実装はプログラムをやる上で大事なことが結構詰まってる。
- packageやclass、メソッドの作成や引数、アクセス修飾子。
- System.out.printlnといった呼び出し方。
- 標準出力。
- ここに手を加えてif文やfor分を入れるだけで十分遊べる。というわけで入れてみよう。

- HelloWorld出力が出来たなら、それを10回表示させてみよう。
- 10回表示出来たなら、引数に「Hello」を渡したら「World」を表示するようにしてみよう。

時間内に出来なかった人

それが普通です。
わからないのが当たり前。
出来ないのが当たり前。
なので、講師を有効活用しましょう。
今なら現役月収130万の講師が無料です。

おしまい