

Olá!

<https://linktr.ee/devmagro>

Eu sou o Lucas Carrilho, vulgarmente conhecido como @devmagro, sou desenvolvedor java e go. Fiz este guia para ajudar iniciantes em migração de carreira / iniciando na área!

*Este guia foi montado com base em **MINHA** trajetória estudando java e migrando de carreira, com base nisso, eu elenco skills técnicas (hard skills) para buscar uma vaga de estágio ou Jr, com foco em **JAVA**.*

*Lembrando que não basta somente isso, ter skills de **relacionamento** (busque por soft skills no google) é essencial e são muito bem valorizadas em entrevistas de emprego em todos os níveis, portanto, não se prenda somente no conteúdo que será exposto aqui.*

No meu canal do youtube @devmagro, você irá encontrar também vídeos montando LinkedIn e Currículo, recomendo dar uma olhadinha lá!

SUMÁRIO

1 - INTRODUÇÃO, REGRAS E ERROS.....	3
1.1 ERROS E REGRAS.....	3
ERRO 1 - VIRAR O DEV PATO.....	3
ERRO 2 - FALTA DE FOCO E CONSTÂNCIA.....	4
ERRO 3 - FALTA DE UM MENTOR.....	4
ERRO 4 - VIVER DE GPT + CONTROL C e V.....	4
REGRAS 1 - EU NÃO TENHO CULPA DA SUA VIDA.....	5
REGRAS 2 - NÃO FAÇA GRADUAÇÃO SEM ESTUDAR POR FORA.....	5
REGRAS 3 - EU NÃO VOU VENDER SONHOS.....	6
REGRAS 4 - PROPAGUE O BEM.....	6
2 - INÍCIO DOS ESTUDOS.....	9
2.1 - NOÇÕES INTRODUTÓRIAS.....	9
2.2 - APRENDA GIT E USAR ARQUIVOS .MD para ORGANIZAR O QUE VOCÊ ESTUDOU!.....	9
3 - CURSOS PARA SEGUIR.....	10
3.1 - YOUTUBE (DevDojo).....	10
3.2 - UDEMY - NÉLIO ALVES.....	12
4 - DESAFIO - SISTEMA DE CADASTROS!.....	13
Passo 1.....	13
Passo 2.....	14
Passo 3.....	14
Passo 4.....	14
Passo 5.....	15
Passo 6.....	15
Passo 7.....	15
5 - APRENDENDO SQL + BANCOS DE DADOS.....	17
5.1 -> INTERFACE PARA VER O BANCO DE DADOS.....	17
6 - MAVEN + CONEXÃO COM DB.....	18
6.1 - O QUE É O MAVEN?.....	18
6.2 - COMO CRIAR PROJETOS MAVEN?.....	18
6.3 - FINALIZANDO TEORIA JAVA + CONEXÃO DE APLICAÇÃO + ASSUNTOS FINAIS.....	18
7 - PRIMEIRA API REST.....	20
7.1 O QUE É O SPRING?.....	20
7.2 OPÇÕES DE CURSOS COM SPRING.....	20
8 - DESAFIO - VERSÃO 2.....	21
9 - TÓPICOS EXTRAS (Teoria).....	22
10 - COMO EU VIRO JÚNIOR?.....	23
11 - CONSIDERAÇÕES FINAIS.....	25

1 - INTRODUÇÃO, REGRAS E ERROS

Seguindo meu guia, você irá avançar com base em alguns níveis de conhecimento:

- 1 - iniciante (**aqui**)
- 2 - mediano
- 3 - quase nível estágio
- 4 - estágio
- 5 - juninho

Lembrando que **NÃO BASTA** ficar fazendo control c + v o tempo todo ou viver consultando o chatGPT, ou fazer exatamente o exercício que o curso propaga, busque se **desafiar**, fazer coisas **diferentes** pra praticar, inovar na resolução dos exercícios.

Além disso, busque treinar soft skills e outras habilidades que também serão necessárias pro dia a dia como desenvolvedor e relacionamento interpessoal.

Aprender a **procurar vagas**, aprender a passar em **entrevistas** de **RH** também são coisas **MUITO** necessárias, ok?

1.1 ERROS E REGRAS

ERRO 1 - VIRAR O DEV PATO

Caso seu foco NÃO seja backend com Java, a dica de ouro é pra você **NÃO** tentar ser fullstack ou aprender tudo ao mesmo tempo, foque somente em 1 coisa, por exemplo: Opções: (C#, FrontEnd com React + básico de node + TS, Go, DevOps)

Dito isso, hoje o que eu mais vejo é uma galera começando com HTML, CSS + JS, não é ruim, mas caso vá por este caminho, foque no frontend (caso você goste). Por outro lado, se você gostou do backend, o que eu mais indico seria o Java ou C#. Caso goste de análise de dados, machine learning, eu indicaria por exemplo Python.

Este guia é baseado **EM JAVA**, mas aplica-se para C# de uma forma bem parecida.

Portanto, escolha seu foco e vá até um ponto de estar fazendo bons projetos com ele, procure um mentor na área e peça para que ele o avalie e julgue seu desempenho nos projetos próprios.

Então, nada de começar com JS, depois ir pro Java, depois pro Python ok? Foco até o fim.

ERRO 2 - FALTA DE FOCO E CONSTÂNCIA

Mantenha-se em um estudo de forma constante, mesmo que estude pouco tempo por dia, faça isso **TODO DIA**. Isso te ajuda a memorizar e criar uma adaptação à rotina muito boa!

Não posso te prometer que você vai virar um dev java jr em 6 meses, afinal, hoje em dia não basta **SÓ** saber fazer código, mas sim, ter todo um conjunto de habilidades que o mercado pede, porém, **ACREDITAR** no processo e trabalhar duro e de forma **disciplinada (traduzindo, fazer repetidas vezes todo dia)**, te traz resultados de forma bem rápida.

Até hoje, como eu disse, já falei com diversas pessoas iniciando na área, conheci alguns que com 6 meses estavam empregados, com 4, com 1 ano, 3 anos, mas todos persistiram até chegarem lá. No final dá certo se você se dedicar, eu garanto.

ERRO 3 - FALTA DE UM MENTOR

Resolvi adicionar este ponto porque vejo muita gente sem um **direcionamento**, eu mesmo não tenho tempo suficiente para analisar e fazer code reviews orientando a galera (ocasionalmente eu tento fazer), este guia serve para te orientar, mas um mentor faz isso de forma mais detalhada.

O que eu indico? Procure um mentor com experiência na área e, de preferência, **de mesma stack** que você. Essa pessoa poderá lhe indicar a melhor forma de estudar determinado assunto, tirar suas dúvidas e lhe dar aulas particulares de determinado tema.

Onde encontrar? Em fóruns da linguagem, grupo de telegram entre outros, além de comunidades como SouJava por exemplo.

ERRO 4 - VIVER DE GPT + CONTROL C e V

Quando estiver estudando, **NUNCA** pule os exercícios, sempre tente resolvê-los ao máximo sem copiar, tente entender o que se pede e busque soluções aplicando o conhecimento aprendido, ficar apenas copiando e colando código não vai te fazer aprender, pensar e resolver problemas vai.

Além disso, hoje temos o CHAT GPT e milhões de outras ferramentas que podem SIM te ajudar, mas eu **não indico** para pessoas que estão começando.

Sim, vários resolveram meu desafio, me chamaram pra mostrar e quando fui ver inúmeros trechos que as pessoas simplesmente não sabiam responder ou pra que servia, era nítido o uso de chatGPT em diversas partes do código. Logo, **eu não indico o uso no começo devido a forma indiscriminada que é utilizado por iniciantes**, mas posteriormente pode ser de grande ajuda.

REGRA 1 - EU NÃO TENHO CULPA DA SUA VIDA.

💡 Ahhh Lucas, mas eu trabalho de 8-18, chego em casa vou malhar e só me sobram 1 hora do dia.

- Vire fisiculturista, já que é mais fácil não faltar a academia do que estudar. Seu foco é lá, não em estudar.

💡 Ahh Lucas, mas eu estudo na faculdade, mas eu arrumo casa e blá blá blá.

- Um sábio uma vez me disse:

“Cada cão que lamba sua caceta.”

Eu infelizmente não tenho controle sobre sua rotina e dia a dia, o que eu indico é (infelizmente e **não indicado por muito tempo**):

Sacrifique seu sono (durma 6:30 por dia) e foque em maximizar o maior tempo útil do seu dia (estude as 5 ou 6 da manhã, estude algo no almoço ou no busão).

Saudável? **NÃO !**

Recomendo? **NÃO !**

Mas cada um sabe da sua realidade, apenas dei uma sugestão de uma forma que pode ajudar **temporariamente**.

REGRA 2 - NÃO FAÇA GRADUAÇÃO SEM ESTUDAR POR FORA.

DEVO FAZER GRADUAÇÃO? **SIM**, e, de preferência, tecnólogo EAD.

Devo focar em estudar para graduação ou por fora?

- **Por fora, sempre.**

A graduação te permite realizar **ESTÁGIO**, e, hoje, esta é a forma + fácil de se iniciar na área, porém, você **PRECISA** de estar tecnicamente + preparado, utilizando assim a graduação para obter o estágio, mas **não** como fonte principal de conhecimento, ok?

💡 Lucas, fiz graduação mas não estagiei. O que eu faço?

Pós graduação EAD e busque estágio, ou JR (**Hard**)

REGRA 3 - EU **NÃO** VOU VENDER SONHOS

O mercado de T.I **não está mais com o boom da pandemia**. Ele **NÃO** VAI PAGAR **5 mil** inicial para júnior, muito provável que você **não** irá conseguir um estágio / vaga de jr em menos de 1 ano, então **NÃO VENHA** querer viver um mundo mágico.

A área é boa? **Sim**.

Está aquecida? **Sim**, mas não é um mundo dos sonhos.

Como eu citei, não basta só saber teoria, estude ela **SIM**, mas saiba ter outras skills além do código ok? Isso facilita e muito a busca de uma oportunidade.

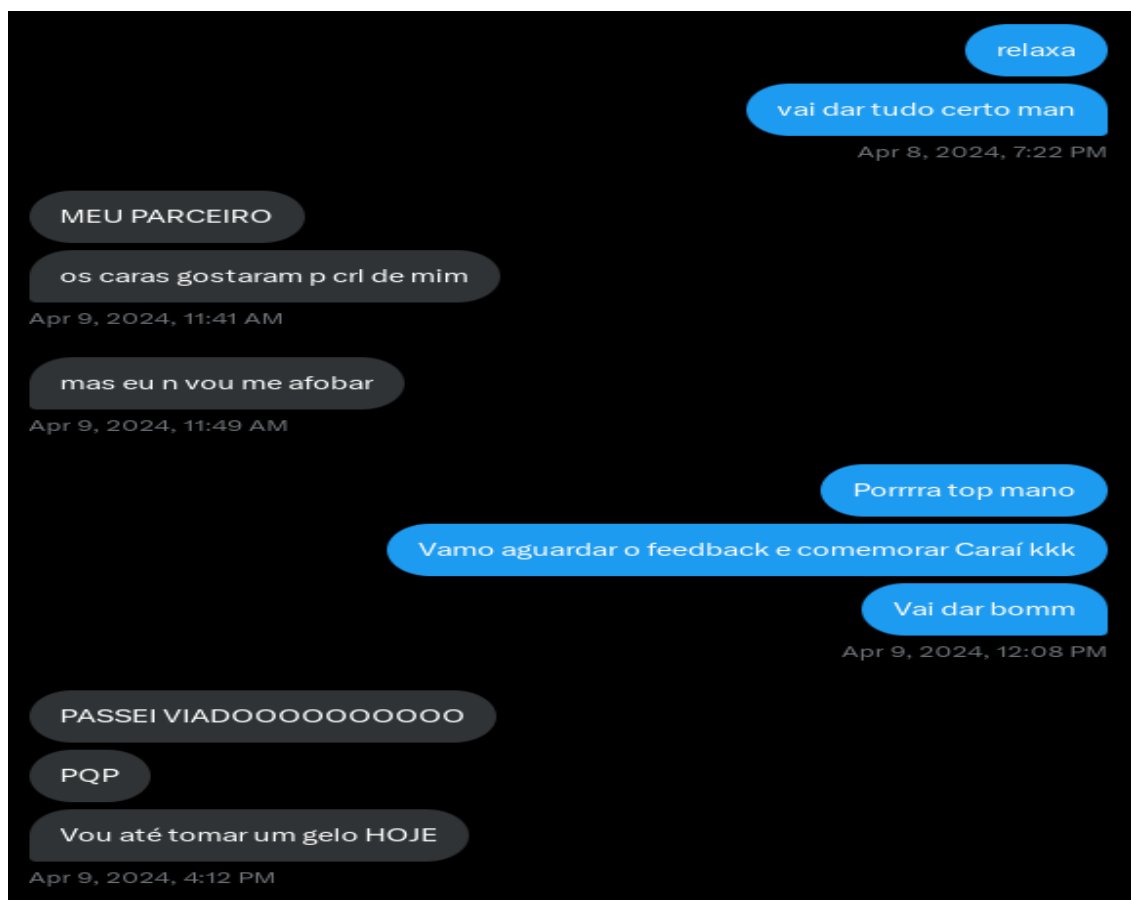
Salário médio de estagiário -> 1000 - 1500 + refeição

Salário médio de jr -> 2.2 - 3k + benefícios.

REGRA 4 - **PROPAGUE O BEM**

Se você cumprir tudo aqui, conseguir sua vaguinha etc, **faça o que eu fiz aqui por você pra alguém ok?** Eu faço isso porque me faz bem e porque já fui muito ajudado um dia.

Isso aqui é o que me motiva a continuar:



Acabei de ir em uma call com o @devmagro de Aprox 1 hora que me esclareceu bastante coisa, o cara é top clã, e obg novamente Man ❤️ ❤️

entrevista foi massa, to na ansiedade agora aguardando resposta

Apr 5, 2024, 11:50 AM

fofoca ce pode mandar 10 minuto que eu vou ouvir
KKKKKKKKKKKKKKKKKKKKKKKKKK

CARALHO FODAAAAAAA

Porra que ótimo mano, acertamo observabilidade

EXCELENTE MANOOO

Apr 5, 2024, 11:51 AM

mano papo reto, nossa entrevista foi um hack brabo
kkkkkkkkkkkkkkkk

po, fiquei feliz demais mano agora a ansiedade que ta matando

Tô tentando estudar mas ta foda kkkkk

↩ Replying to

Porra que ótimo mano, acertamo observabilidade

Demais mano, estudei umas ferramentas como grafana, prometheus e vi os pilares

Apr 5, 2024, 11:53 AM

VAI BEBER CARAI HOJE É SEXTOU

Apr 5, 2024, 11:53 AM

Contexto de microserviços vs monolítico



Fala irmão, demorou mas consegui o estagio em javinha e ainda foi numa empresa grande, to felizao kkkkkkk

tava procurando pra junior também mas to com pressa não

E vi que voce vai dar palestra no workshop, brabo demais mano, vou ir assistir

o evento

Apr 10, 2024, 3:50 PM

Fala mannnn

Porrran nego, que notícia boa

Foca em evoluir agora, uns 6-8 meses

Vamo lá cara, me procura lá, trocar uma resenha tbm

Tá em qual companhia?

Apr 11, 2024, 7:37 AM

Tô estudando desde o começo do ano passado mas comecei a candidatar em dezembro agora

Fiz umas 250 candidaturas kkkkkkk



Tenho vários desses na DM, isso aqui é só pra provar que pra maioria **falta** direcionamento, apenas. Você consegue com teu esforço e é isso.

Agora se tu não quiser acreditar e ficar nessa baboseira de falta de vagas e **mimimimi**, faz o que tu quiser, não sou teu pai nem mãe.

Dito isso, vamos ao que interessa:

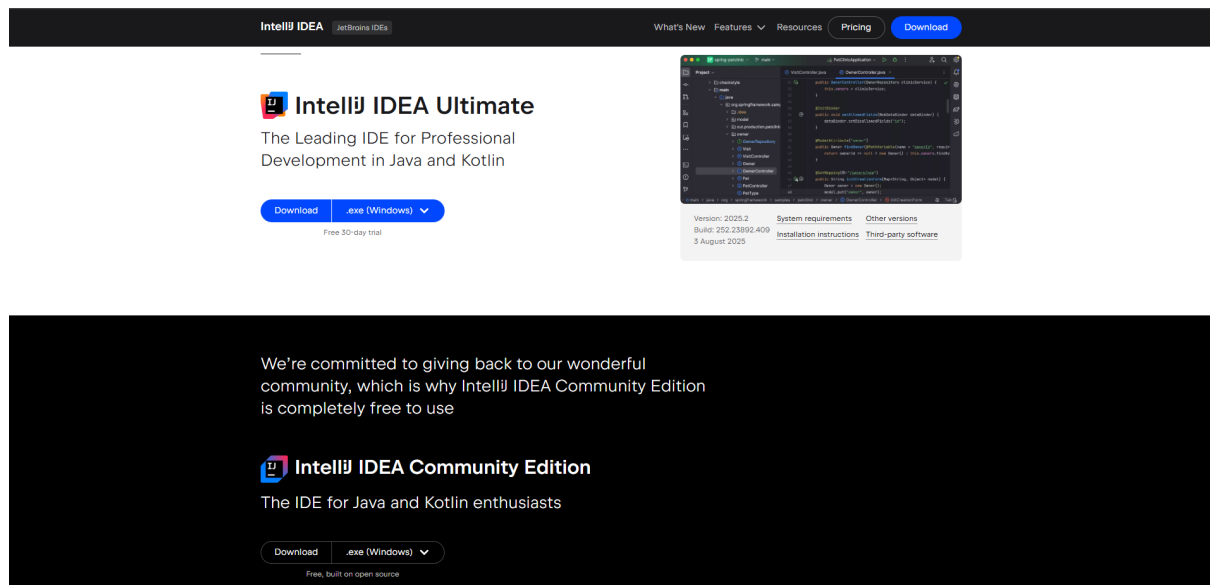
2 - INÍCIO DOS ESTUDOS

2.1 - NOÇÕES INTRODUTÓRIAS

Instale o IntelliJ community:

<https://www.jetbrains.com/idea/download/?section=windows>

No link abaixo, role até a parte final da página, e faça download da Community Edition:



Assista essa noção introdutória da loiane:

<https://www.youtube.com/watch?v=mu2ti43cgwc&list=PLGxZ4Rq3BOBq0KXHsp5J3PxyFaBIXVs3r&index=6>

ASSISTA **APENAS ESSA INTRODUÇÃO**, é o que eu indico, mas se você curtiu.. pode seguir por ela também, mas eu indico youtube o **DevDojo (seção 3.1, logo a frente)**.

2.2 - APRENDA GIT E USAR ARQUIVOS .MD para ORGANIZAR O QUE VOCÊ ESTUDOU!

Bom, agora que você aprendeu a fazer o mínimo de código em Java, antes de propriamente começarmos, eu recomendo que você aprenda a usar o básico do Git, que será sua ferramenta de versionamento e gerenciamento de **CÓDIGO**, além de utilizar para mostrar projetos e suas criações para outras pessoas.

O que eu indico -> Um simples arquivo .md para poder **ANOTAR** o que você estudou e organizar o conteúdo das aulas para você mesmo não perder nada.

EXEMPLO:

[Curso de Git para iniciantes - Aula 1](#)

3 - CURSOS PARA SEGUIR

Neste ponto, haverá vários caminhos a seguir, aqui eu vou listar 3 deles, indico você ir pelo **YOUTUBE , GRATUITO.**

Nélio / Alura SÃO BONS também, mas **EU** indico **YOUTUBE.**

Recomendações de canais de suporte:

- 1 - <https://www.youtube.com/@kipperdev> - Fernanda Kipper
- 2 - <https://www.youtube.com/@loianegroner> - Loiane Groner
- 3 - <https://www.youtube.com/@MichelliBrito> - Michelli Brito

3.1 - YOUTUBE (DevDojo)

Link com a playlist inteira do curso de java desse cara **FODA:**

 00 - Maratona Java Virado No Jiraya - Bem-vindo

Recomendo você ir até a **aula 95 DO CURSO.**

PAUSA!!!!!! - > FAÇA A LISTA DE EXERCÍCIOS ABAIXO E SUBA NO GIT!!!

<https://www.dio.me/articles/lista-de-exercicios-para-treinar-logica-de-programacao>

Depois de retornar, fazer a aula **96 até 143.**

PAUSA!!!!!! - > FAÇA O DESAFIO DÁ PAGINA 13 - SISTEMA DE CADASTROS E SUBA NO GIT!!!

Após o desafio completo, subido no GIT, pule para a aula **159 até 217.**

Neste ponto após a aula 217, comece a aprender SQL (página 17) e siga o guia em diante de lá.

FINAL -> treinar!

<https://www.dio.me/articles/lista-de-exercicios-para-treinar-logica-de-programacao>

Faça esses exercícios da lista, suba no git e mande pra mim na DM, assim posso te ajudar com o que precisar!

Isso servirá pra pegar prática na lógica de programação

APÓS A LISTA CONCLUÍDA, vá para o desafio.

3.2 - UDEMY - NÉLIO ALVES

Anteriormente, era um curso que eu indicava.

Após feedbacks de alguns alunos próximos, notou-se que alguns deles não gostaram dos exemplos em aulas (trigonometria) e alguns conteúdos desnecessários.

Não quer dizer que o curso em si é ruim, conheço pessoas que fizeram sua caminhada até a primeira oportunidade por lá, porém, eu não fiz pra opinar, isso é com base em feedbacks que eu recebi e dúvidas que eu tirei.

<https://www.udemy.com/course/java-curso-completo/?couponCode=ST6MT42324>

O que eu indico?

Faça até a aula TRABALHANDO COM ARQUIVOS (feche este tópico)

Depois, lista de exercícios + desafio a seguir.

Lista:

<https://www.dio.me/articles/lista-de-exercicios-para-treinar-logica-de-programacao>

Pode pular:

JOGO DE XADREZ

PARTE 2 - NÉLIO

Feito o desafio, eu não indico concluir o curso do nélio nesse instante, recomendo seguir o caminho comum (aprender banco + persistência via youtube + spring - > TÓPICO 5 DESTE GUIA) mas você pode optar também por continuar com o nélio caso você goste, ok?

Neste caso, pule também o tópico sobre **interface gráfica** com Java.

4 - DESAFIO - SISTEMA DE CADASTROS!

O desafio foi todo refatorado e está neste link:

[LINK GIT](#) - *Desafio sistema de cadastros*

 Desafio Cadastro Antigo - Para quem quiser ver a versão antiga.

PARABÉNS!!!

Se você cumpriu o desafio, me mostre!! Vou adorar ver a abordagem realizada!

Neste ponto, caso você tenha entendido bem os conceitos vistos até aqui, acredito que você estaria em um nível MEDIANO. Sabendo o básico do java bem aprendido e tendo projetinhos pequenos de estudos. Continue em frente!

5 - APRENDENDO SQL + BANCOS DE DADOS

Banco de dados é um assunto diário mais que necessário, dito isso, precisamos que você aprenda com consistência devido conteúdo, dito isso, eu indico que:

5.1 -> INTERFACE PARA VER O BANCO DE DADOS

Recomendo antes de tudo, baixar o DBeaver para poder conectar-se ao banco de dados.

<https://dbeaver.io/download/>

5.2 - CURSO DE SQL COMPLETO

E aqui, teremos um curso de SQL completo com base no MYSQL, porém o conteúdo pode ser utilizado para outros bancos relacionais:

 Curso SQL Completo 2023 [Iniciantes] + Desafios + Muita Prática

6 - MAVEN + CONEXÃO COM DB

6.1 - O QUE É O MAVEN?

Antes de tudo, estude o que é o MAVEN:

<https://www.treinaweb.com.br/blog/introducao-ao-maven-aprenda-como-criar-e-gerenciar-projetos-java>

6.2 - COMO CRIAR PROJETOS MAVEN?

Entendido o conceito do MAVEN, aprenda a criar um projeto JAVA usando-o.

 Como criar um projeto Maven do zero no IntelliJ | DICAS DE DEV

6.3 - FINALIZANDO TEORIA JAVA + CONEXÃO DE APLICAÇÃO + ASSUNTOS FINAIS

Depois, retornaremos ao dev dojo, mais precisamente na aula:

[246 - Padrões de Projeto pt 01 - Builder](#)

Segue a playlist com as aulas:

<https://www.youtube.com/playlist?list=PL62G310vn6nFIsOCC0H-C2infYgwm8SWW>

Basicamente, pegamos uma introdução de padrões de projetos e aprenderemos a conectar uma aplicação com banco, persistências, operações CRUD, testes unitários e outros assuntos importantíssimos!

Neste ponto, assista a aula 246, até o final, que é a aula 285 (devdojo)

PARABÉNS MAIS UMA VEZ!

**PARA MIM NESTE PONTO, VOCÊ
CHEGOU AO NÍVEL PRÉ ESTAGIÁRIO!**

(espero que você ainda não esteja viciado 100% em café)

7 - PRIMEIRA API REST.

Neste ponto, já sabemos o que é o maven, já sabemos conectar no banco de dados, sabemos o que são as operações CRUD, agora, aprenderemos a ferramenta + utilizada com java, o Spring e suas features.

7.1 O QUE É O SPRING?

<https://www.treinaweb.com.br/blog/o-que-e-spring-framework>

7.2 OPÇÕES DE CURSOS COM SPRING.

UDEMY - Está incluso em qualquer curso java, incluindo o do Nélio.

GRATIS-  Spring Boot 3 | Curso Completo 2023

Além disso, nesse ponto, há diversas opções de curso para criar API REST via youtube ok?

8 - DESAFIO - VERSÃO 2

Bom, neste ponto, não há muito o que desafiar ou criar novas ideias, o que faremos aqui, é **REUTILIZAR O PRIMEIRO DESAFIO**.

Porém, dessa vez, iremos implementar o mesmo sistema de cadastro, porém sem .TXT.

O que você irá fazer?

Refatorar, conectando a um banco de dados DIFERENTE do que você aprendeu na aula de persistência em banco.

Depois, criará todas as operações CRUD para este processo, criará todas as entidades que criamos antes (perguntas, alunos, respostas) e terá integração com banco de dados.

ALÉM DISSO, é NECESSÁRIO que tenhamos testes unitários na aplicação (são muito bem valorizados, julgo que seja um conteúdo essencial).

Além disso, temos que tratar as exceções de nossa aplicação, deixar rotas bem definidas, responder pro usuário com o status HTTP correto ok?

PARABÉNS MAIS UMA VEZ!

PARA MIM NESTE PONTO, VOCÊ CHEGOU AO NÍVEL ESTAGIÁRIO (conhecimento técnico)!

MAS...

Lembre-se, cada vaga varia, cada vaga pode pedir de uma forma diferente (pra mais ou pra menos), eu considero este ponto uma quantidade IDEAL de conhecimento adquirido para poder aplicar pra vagas de estagiário (considerando que a pessoa tenha praticado bastante, não tenha vivido de control c +v e saiba falar sobre o que codou, e tenha também colocado alguns projetinhos em seu git).

Além disso, não basta saber somente código, alguns tópicos também são importantes de serem estudados, e vamos abordar isso a seguir.

9 - TÓPICOS EXTRAS (Teoria)

9.1 - SOLID

<https://medium.com/desenvolvendo-com-paixao/o-que-é-solid-o-guia-completo-para-você-entender-os-5-princípios-da-poo-2b937b3fc530>

<https://www.youtube.com/watch?v=ZYCABdRwRNk>

9.2 - Pilares da POO

<https://www.devmedia.com.br/os-4-pilares-da-programacao-orientada-a-objetos/9264>

📺 Os Pilares da Orientação a Objetos: Encapsulamento #01

9.3 - Arquitetura - Model - View - Controller (MVC)

<http://www.linhadecodigo.com.br/artigo/2367/abordando-a-arquitetura-mvc-e-design-patterns-observer-composite-strategy.aspx>

📺 Arquitetura MVC - Funcionamento real com explicação fácil

9.4 - Perguntas teóricas sobre o contexto do JAVA.

<https://www.turing.com/pt/interview-questions/java>

<https://www.geeksforgeeks.org/java-interview-questions/>

<https://codegym.cc/pt/groups/posts/pt.78.as-21-principais-perguntas-da-entrevista-sobre-java>

<https://www.guru99.com/pt/java-interview-questions-answers.html>

10 - COMO EU VIRO JÚNIOR?

10.1 - DIFERENÇA DO JÚNIOR PRO ESTAGIÁRIO

Normalmente, em termos de níveis, seja júnior, pleno, senior, o que vai te diferenciar é a MELHORIA dos aspectos em gerais, **normalmente** um júnior vai ter um conhecimento técnico + aprofundado, terá também um conceito da regra de negócio mais bem entendido, e, principalmente, softskills mais bem desenvolvidas, além de autonomia na tomada de decisões e vivência de cotidiano em programação, e daí por diante.

Portanto, não há uma regra bem definida do que estudar para chegar no JÚNIOR, algumas empresas, vão considerar o conteúdo que citei até AQUI como suficiente pra júnior, outras mais, outras menos.

O que vou falar a seguir são melhorias que acredito que pra um backend faça mais sentido, não considero que sejam TODAS obrigatórias, mas acho que conhecimento nunca é demais, e, além disso, avaliando diversas vagas para desenvolvimento backend, é **COMUM** que sejam cobrados conhecimentos adicionais.

Agora falando o famoso PORTUGAL:

IMAGINA QUE TU TÁ NUM JOGO DE RPG, CHEGOU NO LEVEL 50 E AGORA VAI TER QUE ESCOLHER UMA PROFISSÃO PRA “ESPECIALIZAR”

Ou seja, não há certo ou errado, há o que cada um escolheu aprofundar +, e isso pode agradar vaga X OU Y de acordo com sua preferência.

O que eu indico? Aprofundar ao máximo em algo que o “core” do java costuma pedir, depois você decide de acordo com teu gosto, ok?

10.2 - SUGESTÕES DE ESTUDO

Nesse tópico, vou elencar implementações que acho que seriam boas para serem feitas no seu projeto anterior, lembrando, não indico criar 1 separado pra cada, mas ir aprimorando o que já foi feito ok?

10.2.A - > DOCKER

Entender o conceito, aplicações e uso cotidiano, uma ideia é rodar a sua aplicação web com ele, um banco de dados, um frontend qualquer com ele

10.2.B -> FERRAMENTA DE DOCUMENTAÇÃO

Descrever documentação de cada endpoint da sua aplicação é algo também muito bem visto, aprender o básico de HTML e criar um README pro seu projeto agradável é ótimo, isso somado a uma boa documentação de cada endpoint da api com o uso de uma ferramenta como, por exemplo o Swagger.io é uma boa pedida.

10.2.C -> BANCO noSQL (não relacional)

Este conteúdo é constantemente cobrado nos dias atuais, tente entender a finalidade de usar bancos não relacionais, tente implementar um mongodb na sua aplicação pode ser um ótimo desafio.

10.2.D -> CACHE DE APLICAÇÃO

Entender como o cache funciona nas aplicações web e o ganho (e custos) por trás dessa forma pode ser muito útil no dia a dia, além de claro, ser mais um conhecimento adquirido.

10.2.E -> SEGURANÇA E AUTENTICAÇÃO

Que tal se aprofundar um pouco nesses conceitos? Implemente algum sistema de autenticação aos usuários da sua aplicação. No Spring do Java, temos o módulo Spring Security, que é bem interessante de ser visto, estudado e implementado.

10.2.F -> CLOUD

Assunto essencial para boa parte das vagas backend, nesse ponto, recomendo escolher a AWS (por deter maior parte do mercado e ter + recursos), e implementar via Spring Cloud utilizações e integrações de alguns serviços famosos como:

S3, filas SQS (outro conceito FODA), LAMBDA, SNS, System Manager, Ec2

10.2.G -> CI / CD

É algo muito utilizado nos dias atuais que garante a entrega de forma integra nos deploys de aplicações cotidianos, neste ponto, você pode usar o Actions do github, FLUX, Jenkins entre outros.

10.2.H -> Outra STACK

Neste ponto, aprender outra linguagem pode ser algo vantajoso, mas acho que isso só deve ser feito quando faz sentido pro que você busca, se você já está como estagiário java, e vê que a empresa utiliza frontend com React, talvez seja uma boa pedida. Ou se utilizam mobile android, talvez kotlin seja melhor.

Dito isso, não há uma regra clara do que se deve estudar, eu indico para frontend com Java sempre Angular, e se for backend Kotlin, por serem linguagens que normalmente são cobradas juntas, mas não recomendo ir por este caminho antes de estar empregado.

11 - CONSIDERAÇÕES FINAIS.

O que vai te fazer ser Junior? Saber tudo o que você aprendeu até aqui, comprovar isso com PROJETOS, e SEMPRE APRIMORAR os projetos com novos conhecimentos e aprender a integrar com soluções CLOUD entre outros, há MUITA coisa que pode ser feita e aprimorada e isso levará tempo, além é claro, de softkills.

Ser júnior, não é só ter o conhecimento técnico, mas também ter a habilidade de lidar com pessoas e equipes, saber ter AUTOGERENCIAMENTO para receber as tarefas, estimá-las e concluí-las. Ser dev não é só escrever código, ser dev é principalmente saber lidar com pessoas e SEMPRE continuar buscando conhecimento, busque ser alguém que você gostaria de trabalhar junto!

Outros assuntos como estrutura de dados, algoritmos, modelagem etc, que normalmente são vistos na faculdade **TAMBÉM** são importantes, mas no momento aqui, o foco será em ENCURTAR a caminhada para o caminho de virar estagiário / jr.

Acredito firmemente que para galgar degrau maior que este, será necessário SIM ter conhecimento da base da programação, mas é muito mais confortável estudar essas coisas com dinheiro no bolso e trabalhando na área do que LISO.

Indicação de grupo para encontrar oportunidades:

[Telegram: Contact @vagastalentsjuniors](https://t.me/vagastalentsjuniors)

Por fim, eu desejo toda sorte do mundo pra você e que você NÃO PARE nunca de estudar e aprimorar esse nosso desafio, ele servirá como um EXCELENTE PORTFÓLIO.

O que precisar, pode me chamar, meus links de contato estão a seguir no meu linktree

<https://linktr.ee/devmagro>

Bjos do Magro pessoal!

“Maior é aquele que se propaga.”