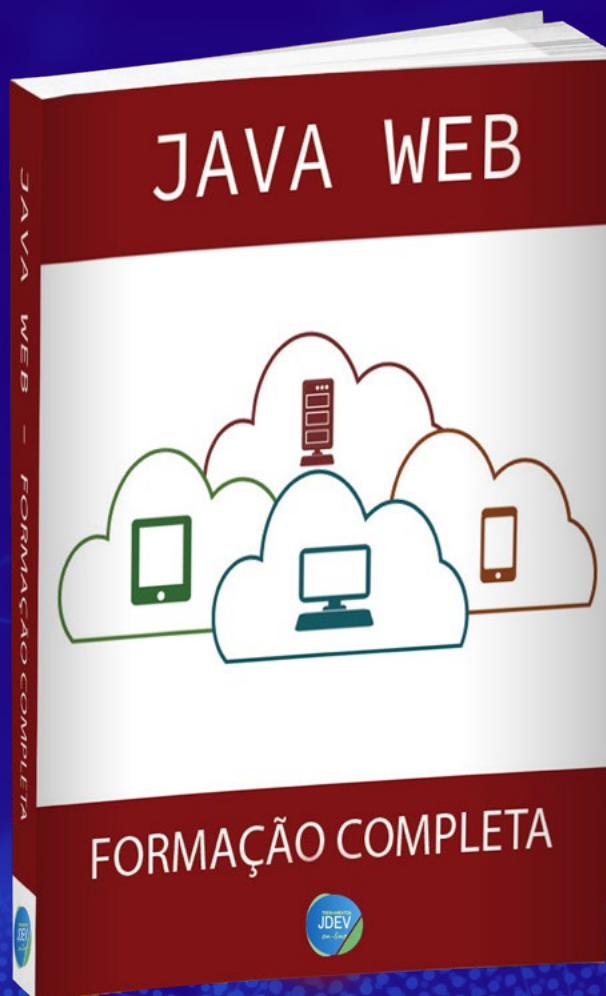


# Livro gratuito de JAVA WEB

**Curso mais  
completo do Brasil**



**Passo a passo  
para ser programador  
Java Web**

A profissão de programador Java Web é enorme e assim como você eu fiquei sem saber por onde seguir! E com esse e-book vou te ensinar o caminho correto para ser um programador.

# Sumário

Alex Fernando Egidio | 3

Compartilhe e me siga nos canais de conteúdo | 3

Projetos que você aprenderá a criar do ZERO | 4

Frameworks usados nos projetos | 5

Perguntas frequentes (FAQ) | 6

Ficou com dúvidas? | 7

Agora eu vou mostrar para você qual os caminhos corretos para ser um programador profissional. | 8

Aula Gratuita → Como funciona um sistema Desktop? | 9

Tenha muita calma em todas as horas | 9

Estudar, pesquisar e aprender sem parar | 10

Aula Gratuita → Como funciona um sistema Web? | 11

Foque muito na Lógica de Programação | 12

Foque no aprendizado de algoritmos | 13

Seja especialista em apenas uma linguagem | 15

Você precisa ter FOCO e objetivo | 16

Abuse do aprendizado em programação orientada a objetos (POO) | 19

Agora vamos falar do assunto mais importante. Por onde começar no Java Web? | 20

Nossa, como aprenderei tudo isso? | 25

Depoimentos dos alunos felizes | 26

JavaScript não é Java | 27

Cascading Style Sheets (CSS) | 29

Frameworks MVC em Java Web | 31

Frameworks ORM Hibernate e JPA | 32

Hibernate | 33

Banco de dados e SQL | 36

Segurança em aplicações Java Web | 37

Frameworks de Componentes e MVC | 38

O que é JSF? | 38

O que é PrimeFaces? | 39

Agora você irá conhecer cada aula do curso | 43

## Alex Fernando Egidio

Autor, criador e fundador do JDev Treinamento TI. Atua no mercado como desenvolvedor e engenheiro de sistemas em Java é apaixonado por desenvolvimento web em Java e pela “mágica” que seus frameworks trazem para o dia a dia de nós desenvolvedores. Com mais de 10 anos de experiência ajuda programadores do Brasil todo a se tornarem profissionais.

## Compartilhe e me siga nos canais de conteúdo



Página Oficial | Grupo Oficial | Perfil Pessoal



Canal Youtube Oficial



Perfil Profissional



Java Avançado Cursos



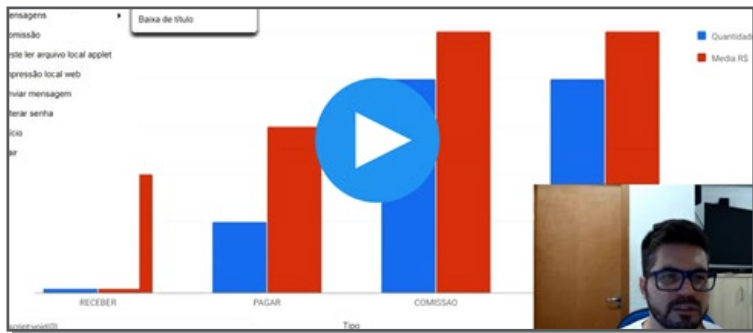
**sim! Iniciar o curso agora mesmo**



**clique aqui!**

## Projetos que você aprenderá a criar do ZERO

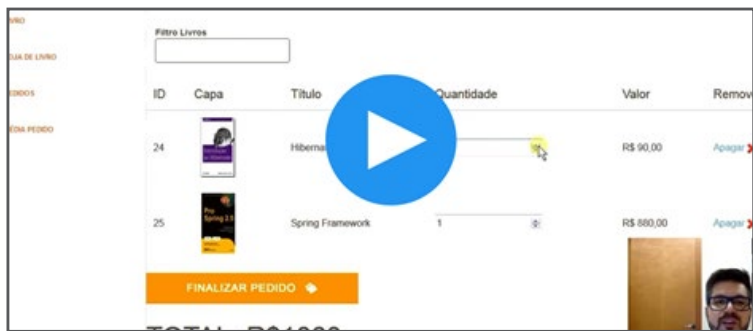
#1 – JavaServer Faces + PrimeFaces + Hibernate e muitos mais.



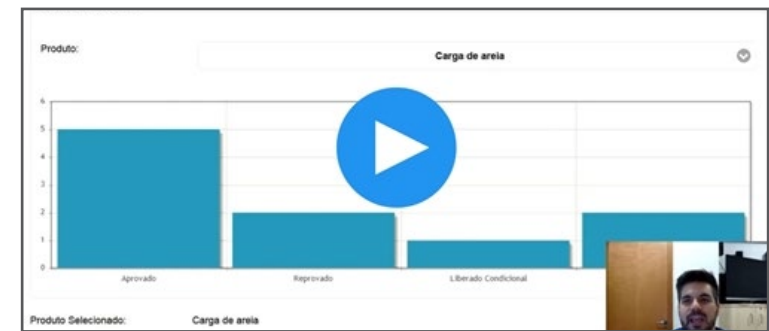
3# – Sprin MVC + JSP + Hibernate e muito mais.

A screenshot of a web application interface for reservations. It includes a form with fields for 'Professor', 'Curso', 'Disciplina', 'Laboratorio', 'Horario', and 'Data'. There are buttons for 'Reservar' (green), 'Limpar' (red), and 'Todas Reservas' (blue). A large blue play button is overlaid on the form. A small video feed of a person is visible in the bottom right corner.

#2 – AngularJS + Spring RESTful + Hibernate e muito mais.



4# – PrimeFaces Mobile + JSF + Spring + Hibernate e muito mais.



## Frameworks usados nos projetos

Spring Framework	Facelets
Spring Security	Hibernate
JSF 2.2	JPA (Java Persistence API)
PrimeFaces	Hibernate Envers
iReport	Flyway
jQuery	JavaEE
PrimeFaces Mobile	BootsStrap
Google Charts	SQL
AngularJS	CSS + JavaScript
Eclipse	PostgreSQL
Ajax	Java Avançado



Imagina você aprendendo todos esses frameworks queridos pelo mercado e tudo em um só curso.



**Quero assistir ao vídeo agora!**

## Perguntas frequentes (FAQ)

### 1. O curso tem duração limite?

Não! Você tem acesso vitalício ao curso e pode assistir as aulas no seu próprio ritmo, como achar melhor, também é possível assistir as aulas quantas vezes forem necessárias. você é livre! Você terá acesso e suporte sempre que precisar.

### 2. Como eu acesso o curso?

O curso é totalmente online, assim que é efetuado a inscrição a plataforma te envia automaticamente o acesso para seu e-mail e você já começa a estudar em alguns minutos.

### 3. Eu posso baixar as aulas?

Sim, você pode assistir a todo o curso baixando as aulas para seu computador e podendo assistir sem ter acesso à internet.

### 4. O Certificado tem custo?

Não, o certificado está incluso no valor da inscrição e você imprimirá o certificado sem nenhum custo a mais.

### 5. Como obtenho suporte a dúvidas?

Na área on-line de estudos possuí em baixo de cada aula os campos para perguntas e respostas, onde você postara a sua dúvida e logo responderei e ajudarei você exclusivamente e individualmente.

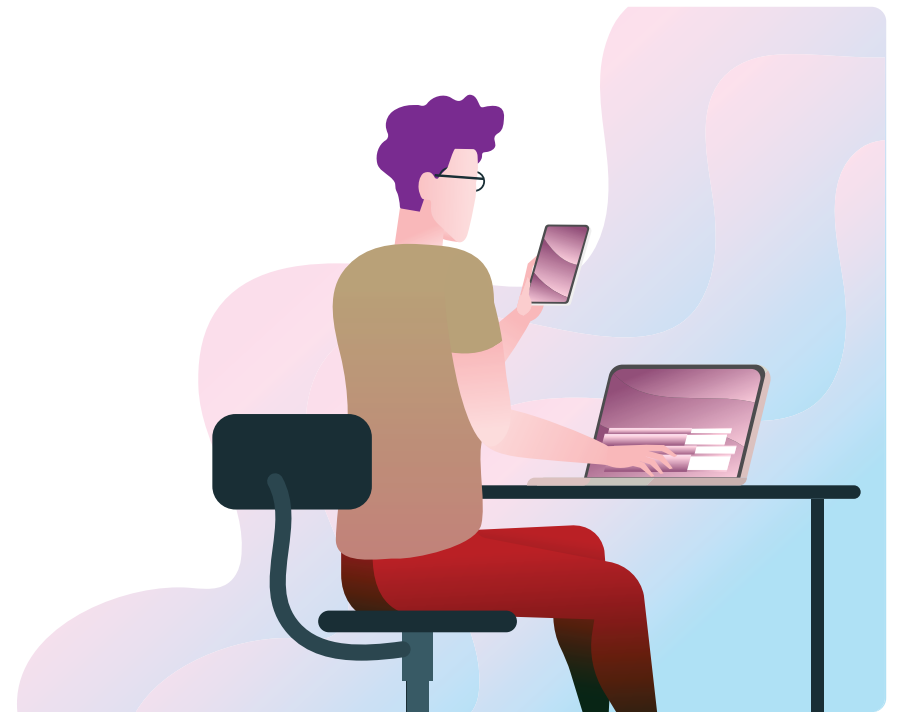


## 6. As atualizações do curso tem custo?

Não, você se matriculando agora no curso todos os vídeos que serão adicionados e todo o conteúdo que for incluído e atualizado você terá acesso sem custo nenhum.

## 7. Esse curso é pra mim? Qual é o requisito para estudar?

Se você é estudante de java, está na faculdade ou mesmo estudando por conta própria para se atualizar e já possui uma boa noção em orientação a objetos e/ou já consegue criar um cadastro pequeno em java desktop, sim esse curso é pra você.



## Ficou com dúvidas?



[javaavancado@javaavancado.com](mailto:javaavancado@javaavancado.com)



[alex.fernando.egidio@gmail.com](mailto:alex.fernando.egidio@gmail.com)

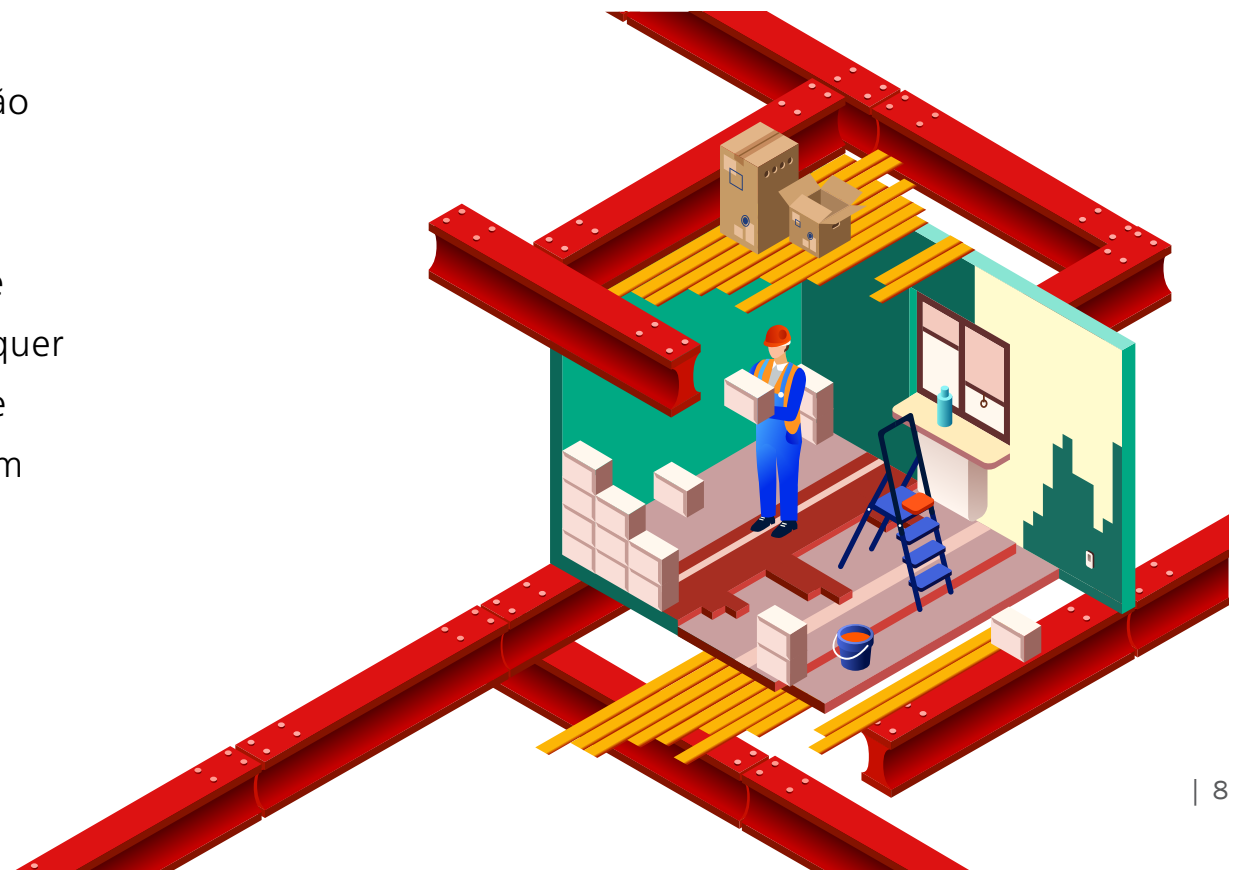


(45) 9 9979-5800

## Agora eu vou mostrar para você qual os caminhos corretos para ser um programador profissional.

A primeira regra que você deve seguir é NÃO PULAR ETAPAS, programador é uma profissão como qualquer outra e exige esforço, dedicação e amor pelo que se faz, sentir prazer em ver o resultado final e saber que foi você que criou todo aquele sistema lindo e com um monte de funcionalidades legais que impressionam qualquer um e satisfaz o cliente 100 % porquê o cliente final e o resultado final é o mais importante em nossa profissão.

Quando falo não pular etapas imagina uma parede de tijolos e se faltar tijolos essa parede desaba ou até mesmo é impedida de ser construída. Essa é a ideologia principal, temos que construir toda a nossa base bem sólida, forte e fiel para poder ter uma construção da nossa profissão perfeita.





## Aula Gratuita — Como funciona um sistema Desktop?

Como funciona um sistema desktop?. Um sistema desktop funciona diferente de um sistema web é claro, mas você sabe como?. Então **CLIQUE AQUI** para assista a aula.



## Tenha muita calma em todas as horas

Eu não tenho como negar e não vou mentir, durante os mais de 10 anos da minha carreira de programador eu já me vi muito irritado, estressado, desanimado, muito e muito furioso mesmo com o código que não conseguia resolver ou entender o que o mesmo fazia dentro do sistema, sem saber qual o caminho seguir.

Mas felizmente chegou o dia em que amadureci profissionalmente e pessoalmente e isso me fez ver e aprender que quanto mais preocupado e/ou estressado eu ficava menos eu progredia como desenvolvedor, se estivesse estressado não conseguia pensar e raciocinar para programar a lógica e depois escrever o código Java.

Então o segredo pra ser um ótimo desenvolvedor é ter calma, paciência e dedicação.

## PERDER A PACIÊNCIA É PERDER A BATALHA

MAHATMA GANDHI



## Estudar, pesquisar e aprender sem parar

Em nossa área de programação é muito comum aprendermos por demanda, isso quer dizer que vamos aprendendo de acordo com a necessidade e com os problemas e desafios que vamos encontrando no decorrer da carreira.

Até hoje me deparo com problemas e acabo tendo que pesquisar pela solução em nosso querido Google, hoje alias está muito mais fácil encontrar soluções para os problemas porque na minha época de iniciante era tudo muito mais difícil, muitos dos problemas eu apenas conseguia resolver em sites do exterior, estudando a documentação do Java ou mesmo descobrindo por conta própria usando a raça e a coragem, testando, retestando o código por dezenas de vezes e por dias até achar a solução.

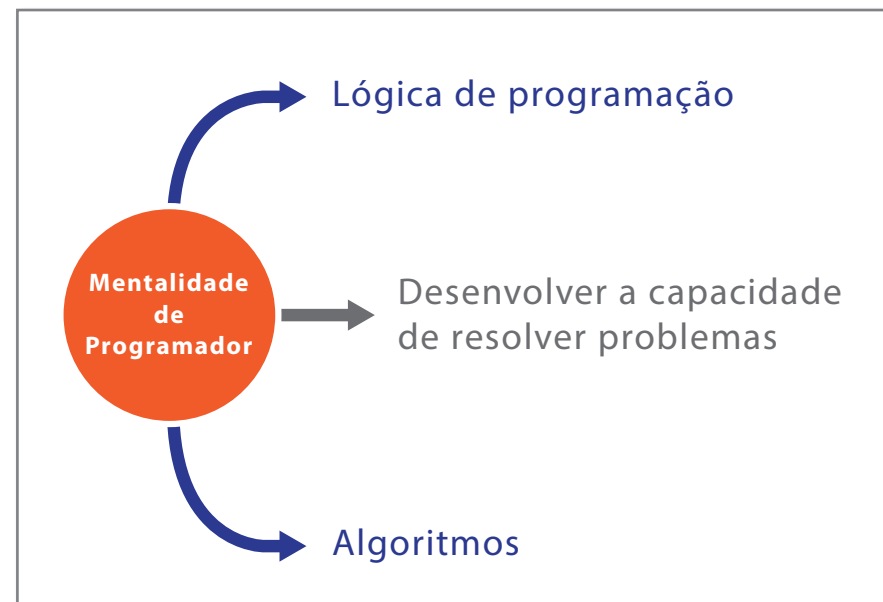
## Aula Gratuita — Como funciona um sistema Web?

Como funciona um sistema web?. Um sistema web funciona diferente de um sistema desktop é claro, mas você sabe como?.

Então **CLIQUE AQUI** para assista a aula.



## Mentalidade de programador



Primeiro de tudo antes de aprender uma linguagem de programação devemos adquirir a mentalidade de programador ou pensar e entender como conversar e dar instruções ou comando para o computador realizar as tarefas do jeito que nós

queremos e assim ter um resultado final correto para um determinado trabalho do dia a dia.

Nesse processo inicial as duas coisas mais importantes são a lógica de programação e o entendimento de algoritmos.



## Foque muito na Lógica de Programação

Lógica de Programação é a técnica de desenvolver sequências lógicas para atingir um determinado objetivo. Essas sequências lógicas são adaptadas para linguagem de computador pelo programador a fim de produzir software com determinada linguagem que em nosso caso é Java.

Uma sequência lógica é denominada algoritmo. Então podemos dizer em linguagem mais coloquial, que um algoritmo é uma sequência de passos para atingir um determinado objetivo. Como podemos ver a lógica de programação trata basicamente de construir algoritmos que serão transformados em programas de computador.

Saber lógica de programação é saber o melhor jeito de escrever um código, para o computador interpretar corretamente. É saber se comunicar com a máquina a partir de uma linguagem seja lá qual for.

Quem quer começar uma carreira como programador deve primeiro aprender lógica de programação. Uma base sólida em lógica garante ao iniciante conhecimentos suficientes para aprender qualquer linguagem de programação.



## Foque no aprendizado de algoritmos

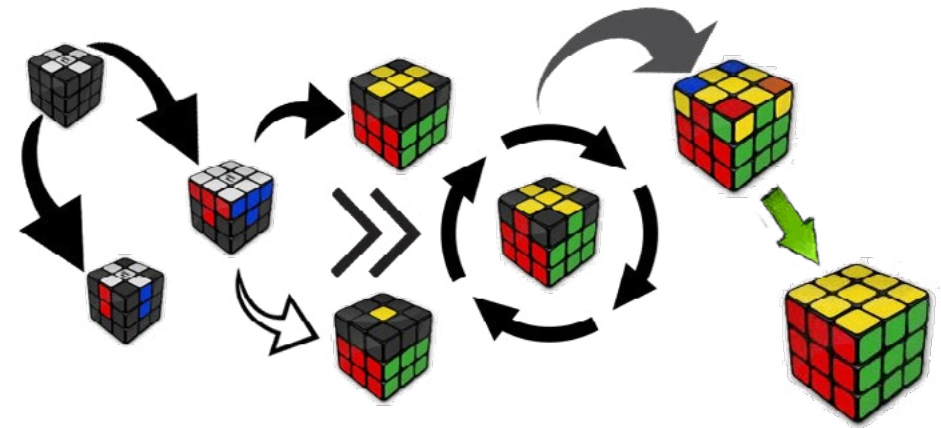
Um algoritmo nada mais é do que uma receita que mostra passo a passo os procedimentos necessários para a resolução de uma tarefa. Ele não responde a pergunta “o que fazer?”, mas sim “como fazer”. Em termos mais técnicos, um algoritmo é uma sequência lógica, finita e definida de instruções que devem ser seguidas para resolver um problema ou executar uma tarefa.

Embora você não perceba, utiliza algoritmos de forma intuitiva e automática diariamente quando executa tarefas comuns. Como estas atividades são simples e dispensam ficar pensando nas instruções necessárias para fazê-las, o algoritmo presente nelas acaba passando despercebido. Por exemplo, quando precisa trocar uma lâmpada, você:



Início:

Verifica se o interruptor está desligado;  
Procura uma lâmpada nova;  
Pega uma escada;  
Leva a escada até o local;  
Posiciona a escada;  
Sobe os degraus;  
Para na altura apropriada;  
Retira a lâmpada queimada;  
Coloca a lâmpada nova;  
Desce da escada;  
Aciona o interruptor;  
    Se a lâmpada não acender, então:  
        Retira a lâmpada queimada;  
        Coloca outra lâmpada nova;  
    Senão  
        Tarefa terminada;  
        Joga a lâmpada queimada no lixo;  
        Guarda a escada;  
Fim;



Então acima acabamos de ver um exemplo de algoritmo escrito em nossa linguagem que falamos, mas o importante é entender que nesse processo podemos definir como o sistema irá se comportar e a partir daí transformamos o algoritmo em um processo dentro de um sistema com alguma linguagem que no nosso caso é Java.

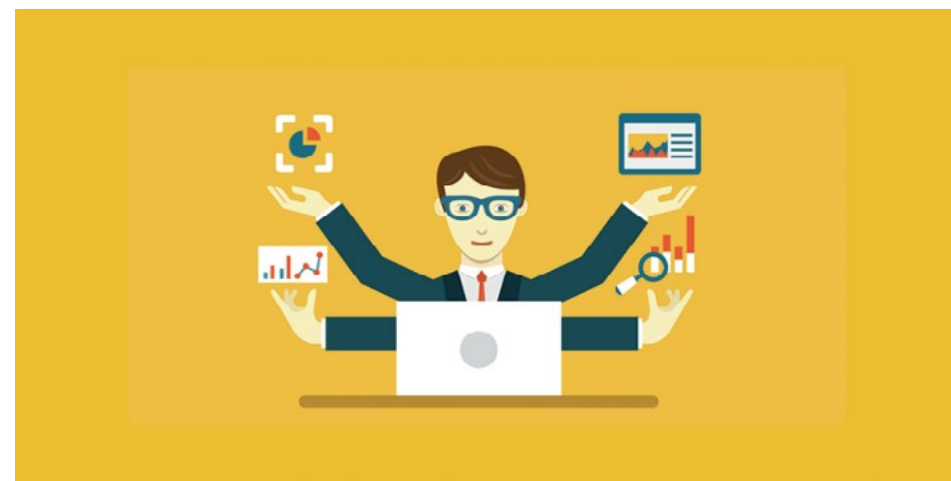


## Seja especialista em apenas uma linguagem

Sim, isso mesmo não acredite nas coisas loucas que você encontra nos fóruns e blogs que existem por aí à fora nesse mundão da internet dizendo que programador tem que saber várias linguagens 2, 3, 4, não isso não é verdade.

Sabe como você sempre terá vagas de trabalhando batendo a sua porta? Quando se tornar especialista no que faz e quando falo isso é ser muito bom mesmo em programação e focado em uma linguagem, só pra lembrar eu estou a 10 anos trabalhando com Java e vou falar a verdade para você já criei dezenas de sistemas para muitas empresas e tenho a certeza que não usei nem ao menos 50% de todo o poder e utilidade que a plataforma Java oferece.

Agora imagina se eu fosse aprender várias linguagens ao mesmo tempo? Com certeza viraria uma bagunça na minha mente e eu não seria especialista apenas saberia um pouco de cada linguagem, mas só para lembrar que sendo especialista você será procurado pelas empresas e terá sucesso em sua carreira de programador, então para concluir tenha foco em uma linguagem e a estude em todas as suas melhores partes.



## Você precisa ter FOCO e objetivo

Quando eu falo FOCO como eu citei acima o foco deve ser na linguagem que você escolheu para sua carreira, dentro da linguagem de programação serão escolhidos frameworks, ferramentas, plataforma de desenvolvimento (IDE), banco de dados e principalmente o seu objetivo o que você quer alcançar?

1. Ser um programador Desktop?
2. Atuar no [desenvolvimento Web](#)?
3. Criar aplicativos [Mobile](#)?
4. Ser desenvolvedor [front-end](#)?
5. Ser desenvolvedor [back-end](#)?
6. Ser um desenvolvedor [full-stack](#)?
7. Ser um profissional [cloud computing](#)?
8. Ser um profissional de [banco de dados](#)?
9. Ou todas as 8 áreas juntas?

Calma, tudo de uma vez não dá não, lembre do que já falamos → Não pule etapas!

Vou citar um exemplo, o **Spring Framework** é o maior framework Java do mundo, ele abrange todos os módulos de desenvolvimento é o ecossistema completo tendo todas as ferramentas e módulos que auxiliam o desenvolvedor Java a criar aplicações com agilidade, qualidade, performance, mas você deve estar se perguntando onde entra a parte do FOCO.

Aqui que está a coisa mais importante, o **Spring Framework** é o mais usado pelas empresas de desenvolvimento de sistemas, praticamente 90% das vagas de empregos para Java pedem obrigatoriamente conhecimentos aprofundados nele e, com certeza, você não aprende em 1 semana, 1 mês ou 5 meses.

Aqui vai a verdade nua e crua doa a quem doer, para se aprender e dominar muito bem um frameworks gigantesco como esse é preciso de em média 2 anos para se sentir confortável e desenvolver com tranquilidade sem contar o tempo para aprender programação básica.

Então é preciso foco, escolher o que aprenderá levando em consideração o que o mercado mais usa, quantidade de vagas de emprego e qualidade em desenvolvimento, escolhendo as ferramentas agora entra o foco nos estudos porque nesse ponto você já sabe que sendo especialista na linguagem e nas melhores ferramentas usadas pelo mercado de trabalho você abrirá as portas para sua carreira e se tornando um profissional de verdade seguindo uma linha de estudo com muito foco sabendo onde quer chegar.

O foco é mais importante ainda quando começamos a se deparar com muitos problemas o que é normal em qualquer linguagem, ferramenta e framework de desenvolvimento, cabe a você pesquisar, aprender e solucionar, a última coisa que deve pensar é partir para outro framework por causa de um problema ou outro, tudo tem solução você tem que aprender a usar, entender o frameworks pra resolver problemas e entregar o sistema para o cliente.

Você acha mesmo que tudo que irei te ensinar no meu curso de **Formação em Java Web**

eu aprendi em 1 ano, 2 anos ou pior ainda 6 meses? Nunca, todo o conhecimento que estou passando levaram 10 anos pra serem adquiridos e eu juro que quando entrei pra estudar programação eu não sabia o que era Java e demorei 2 anos pra fazer o primeiro cadastro

em um sistema e olha só, hoje sou especialista e estou aqui ensinando tudo a você dê uma forma prática, fácil e diferente, entregando todo meu conhecimento passo a passo em cima de todas as dificuldades que já passei e sei que você também passará e com o meu curso tudo ficará mais fácil.

Lembre-se, FOCO, dedicação, paciência, estudo e assim você terá sucesso como eu.



## Abuse do aprendizado em programação orientada a objetos (POO)

A programação Orientada a objetos (POO) é uma forma especial de programar, mais próximo de como expressaríamos as coisas na vida real do que outros tipos de programação.

Com a POO temos que aprender a pensar as coisas de uma maneira distinta, para escrever nossos programas em termos de objetos, propriedades, métodos e outras coisas que veremos rapidamente para esclarecer conceitos e dar uma pequena base que permita soltarmos um pouco com este tipo de programação.

Muitas linguagens estão se tornando orientadas a objetos e pegando mesmo caminho que o Java, mas o Java já nasceu orientado a objetos? Sim, isso é verdade o que podemos ver é que conceitos que existem a mais de anos no Java estão sendo implementados agora em novas linguagens ou atualizações de linguagens antigas mas que são forte no mercado como o Groovy é igual ao Java.

Uma coisa é, Java já nasceu estando muitos anos-luz a frente de qualquer outra linguagem existente até o momento.

Uma coisa interessante é que você se tornando ótimo em orientação a objetos fica muito mais fácil você entender outras linguagens que estão surgindo no mercado.



E Java e sua orientação a objetos só é aprendida colocando a mão na massa mesmo, trabalhando ou mesmo faça um desafio a você mesmo, coloque como objetivo para você aprender a criar um sistema como esses aqui do vídeo → **CLIQUE AQUI PARA ASSISTIR.**

## Agora vamos falar do assunto mais importante. Por onde começar no Java Web?

Você está na faculdade e o professor começou a falar de Java Web e um monte de coisas que não entram em sua cabeça e te deixa perdido sem rumo e o pior ainda, quando busca ajuda nos fóruns e **grupos do Facebook** a confusão só aumenta com a enxorada de informações desconstruídas e confusas que cada pessoa fala uma coisa diferente e só piora a situação.

Pois é, eu acompanho vários grupos e fóruns e vejo que as respostas em grande maioria só confunde mais ainda a pessoa que está buscando o caminho correto para aprender Java Web e entrar no mercado de trabalho.



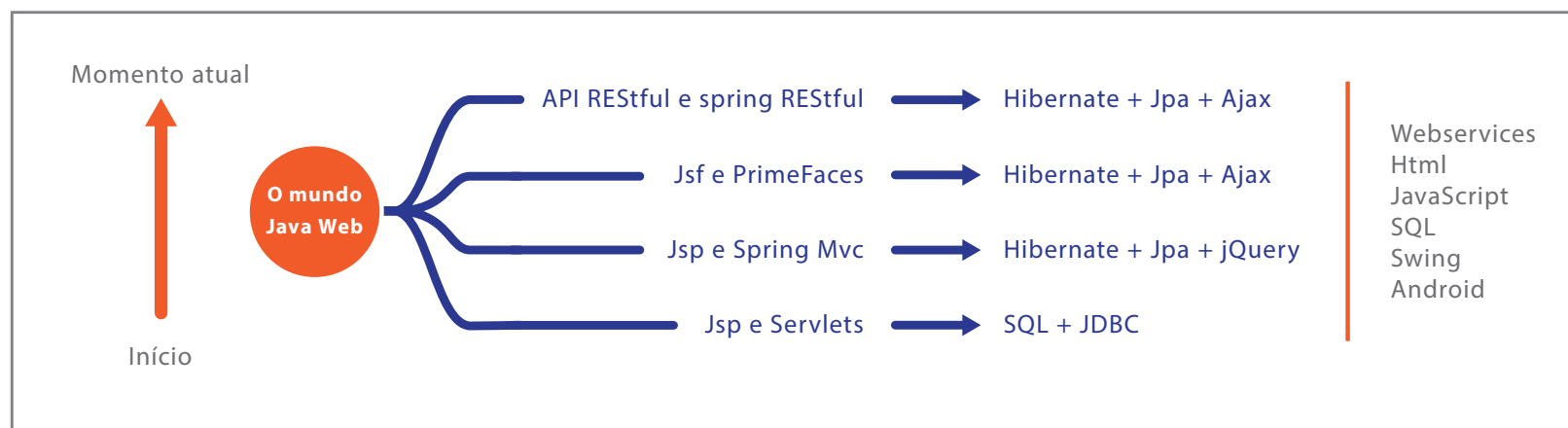
Haaaa! E imagina a sorte que você tem por ler esse texto. Eu tive sorte o suficiente pra trabalhar em várias empresas do Brasil e pegar projetos em todas as plataformas do Java e assim nesses mais de 10 anos de carreira eu aprendi degrau por degrau os pilares para ser programador sênior em Java Web.

E eu percebi que a minha dificuldade era a mesma de muitos e por influência da vida eu consegui aprender todos os pilares necessários e sempre tive em minha mente que um dia criaria meu site (**que você pode conhecer CLICANDO AQUI**) e um

curso para ensinar todo mundo de uma forma fácil e do jeito que eu aprendi.

Cansei de vasculhar blog, sites e baixar projetos que nunca funcionavam e de repente eu conseguia criar aquele código mas não tinha muito a ver com as coisas na internet então sempre me vinha a mente que disponibilizaria todo esse conhecimento para fazer os outros sofrerem menos nessa carreira.

Olhe esse mapa mental simplificado abaixo:



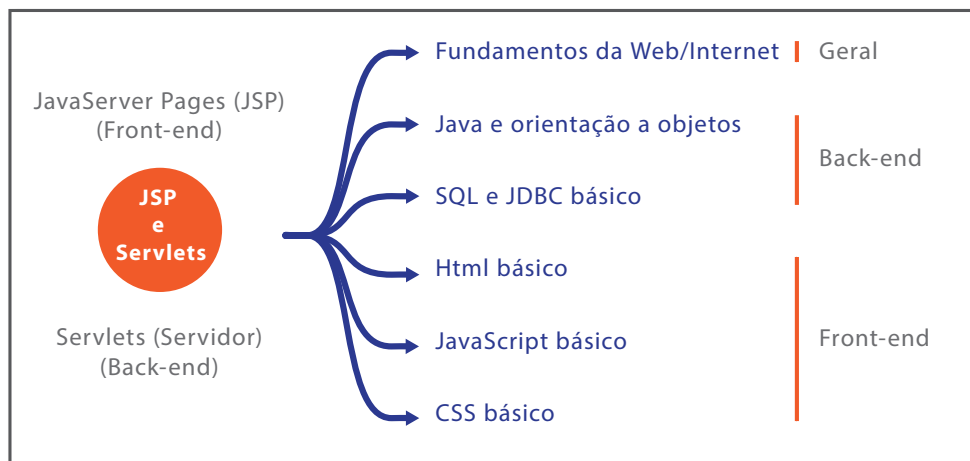
Nós temos 4 grandes módulos mostrando a evolução de baixo para cima que você deve aprender:

1. A linha azul que representa JavaServer Pages (JSP) e Servlets que é a base em Java Web e o que for aprendido aqui será levado para sempre na sua carreira é neste ponto que construiremos a sua base.
2. A linha verde-escuro que representa JavaServer Pages (JSP) e Spring Framework com arquitetura MVC e contém também com o famoso jQuery que nos dá o poder do Ajax e também o Hibernate com JPA que nos dá o grande poder de persistência com banco de dados diminuindo a complexidade do SQL.
3. A linha verde-claro representa os frameworks baseados em componentes e feitos para criar aplicações ricas e com alta produtividade sendo eles JavaServer Faces (JSF) e sua biblioteca principal o PrimeFaces.
4. E na linha amarela representa as APIs RESTful que são tecnologia de ponta onde aplicações nessa estrutura pode ser reusadas e seus do back-end pode ser usando para desktop, apps mobile e navegadores web, neste momento o uso do JSON explode mundialmente.

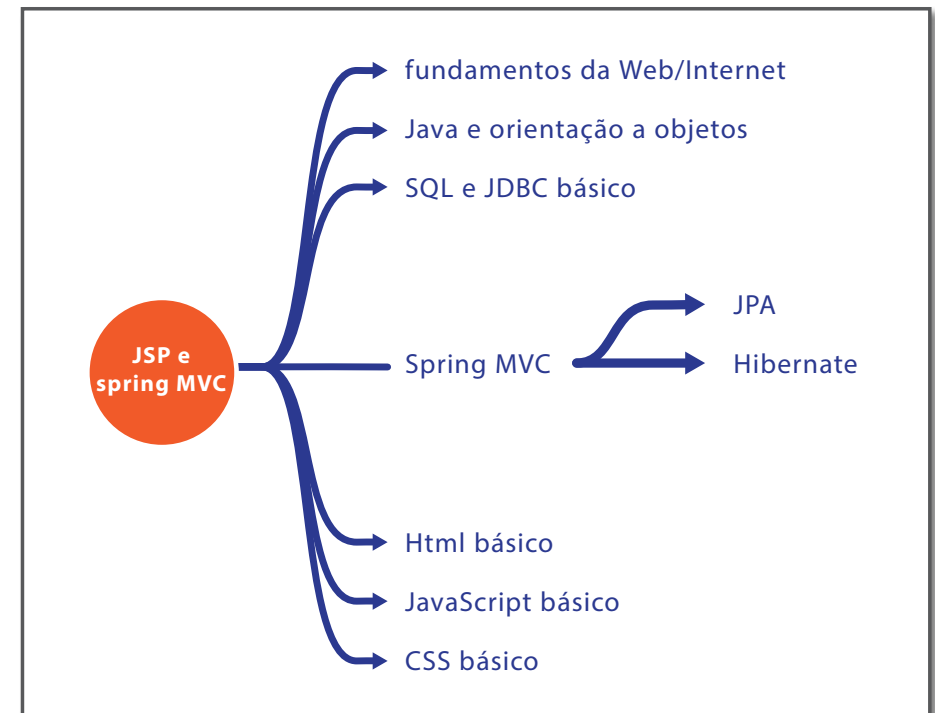
E podemos ver que **html, css, javascript, sql e webservices** seguem acompanhando toda essa evolução desde do início e assim continuará porque são a base para se aprender qualquer frameworks web, lembra da base, não pular etapas e construir a base.

Cada módulo citado acima são arquiteturas diferentes, estruturas de projetos e desenvolvimentos e isso influencia muito no modo de pensar em como programar a aplicação e como criar as funcionalidades dentro dela e o mais importante ainda como fazer os frameworks se conectarem e trafegarem dados da tela para o banco de dados e vice-versa.

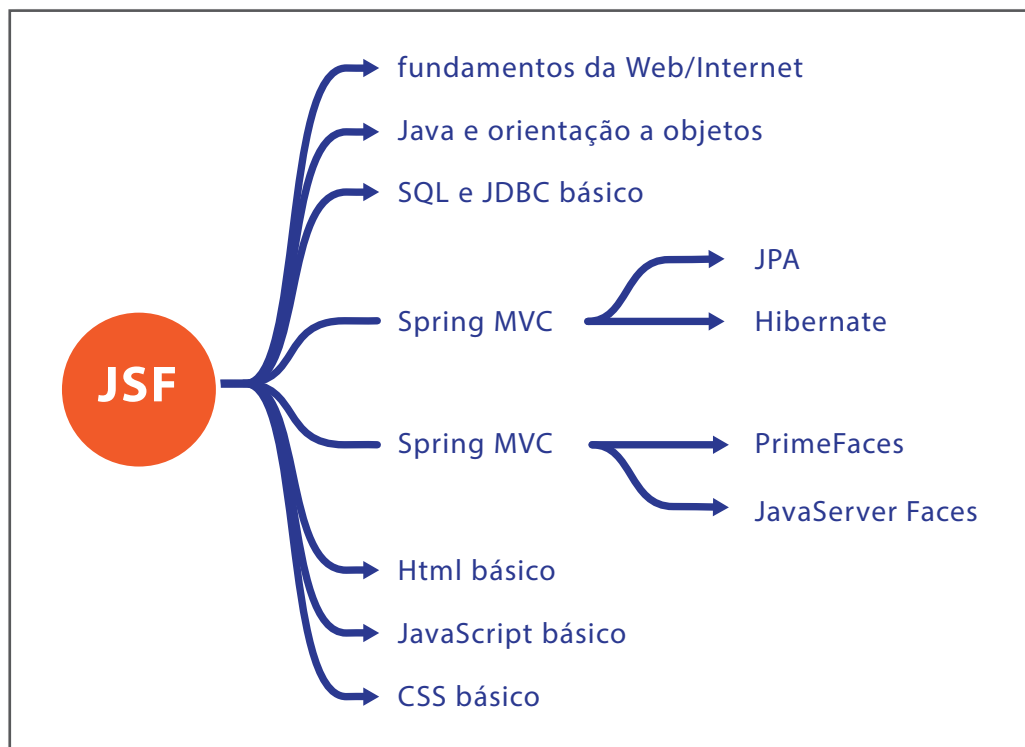
## Modelo mental → JavaServe Pages (JSP)



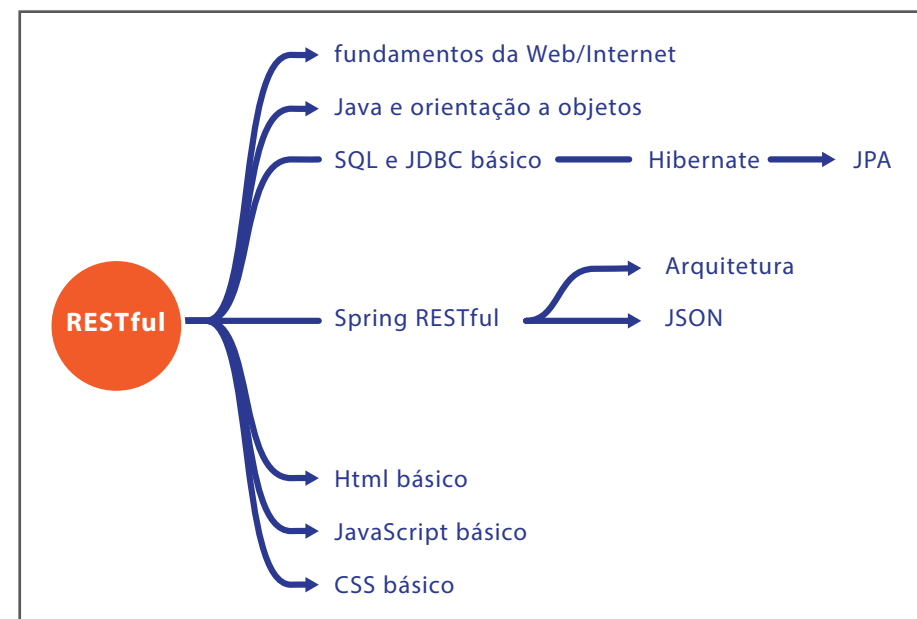
## Modelo mental → JavaServe Pages (JSP) e Spring MVC



## Modelo mental → JavaServer Faces e frameworks



## Modelo mental → Rest e RESTful



## Nossa, como aprenderei tudo isso?

Muito simples, lembra no texto acima que eu disse que é muita sorte você estar lendo esse texto? Então foi para isso que eu criei o maior curso de Java Web do Brasil e tenho muito orgulhos disso.

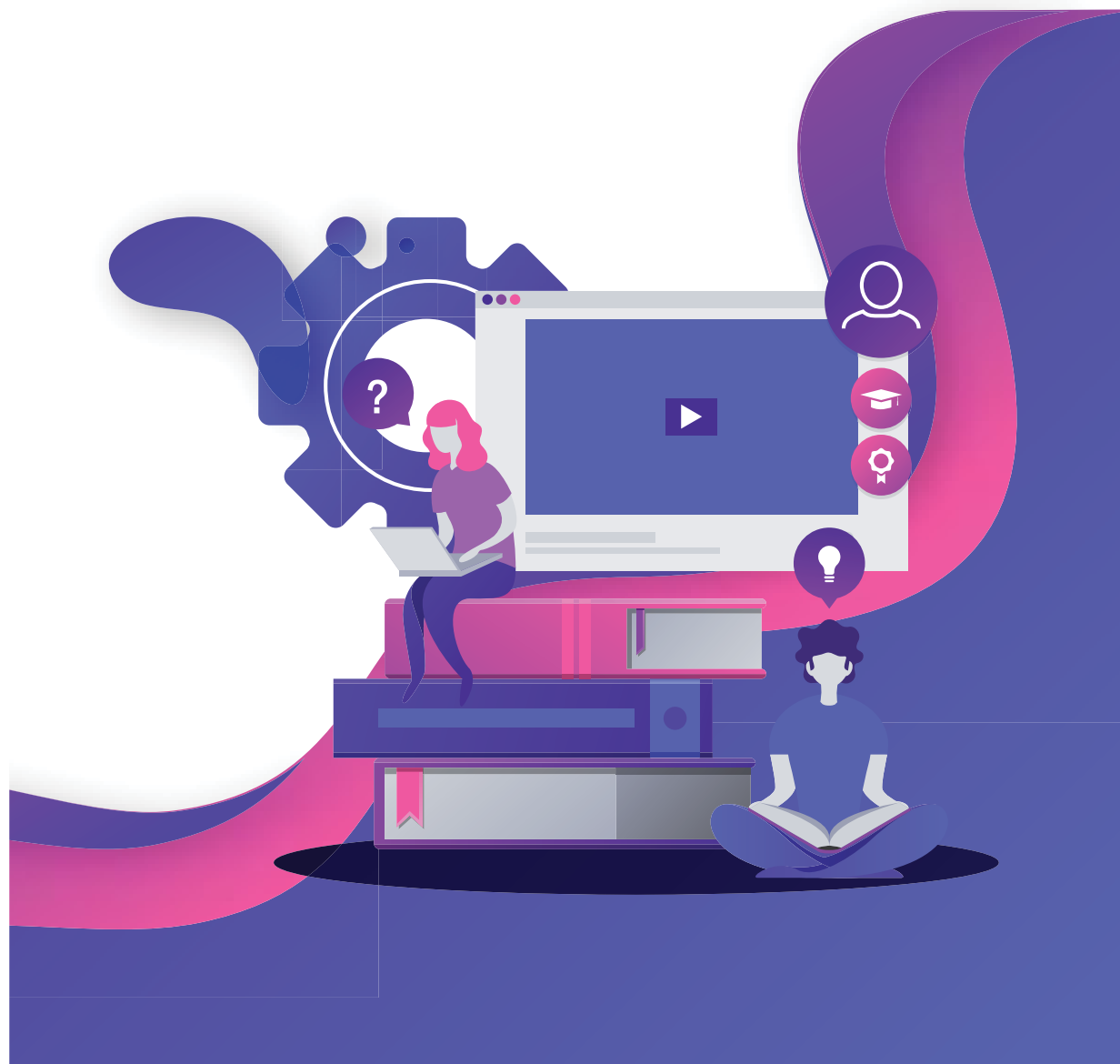
Hoje já passaram pelo meu treinamento mais de 2 mil pessoas que subiram na carreira e conseguiram entrar no mercado de trabalho por causa dos conhecimentos que adquiriram com o curso que criei.

Abaixo você pode conferir toda a grade curricular e também já pode se inscrever e começar a estudar agora mesmo no link → **CLIQUE AQUI PARA COMEÇAR O CURSO.**

**sim! Iniciar o curso agora mesmo**



**clique aqui!**



## Depoimentos dos alunos felizes



Wagner Felipe Martins



"Estou impressionado com a qualidade do curso e com o conteúdo ensinado. :)"



Tiago Silva Pereira



"Ótima explicação e ótima didática e suporte ótimo. Graças a esse curso eu consegui entrar no mercado de trabalho como programador web."



Adevaldo Penha Costa  
Atualizada há 3 dias



Muito bem ministrado o curso uma boa didática de ensino fácil compreensão fator essencial para aprendizagem

- + Informações importantes
- + Explicações claras
- + Apresentação cativante
- + Atividades práticas úteis
- + Descrição precisa do curso
- + Instrutor experiente

Replied to [Iago Holanda's comment](#) on [Alex Fernando Egidio's public publicação](#)



**Jair Henrique Dresch** Cara não to recebendo nada e nem quero... o curso é foda d mais.... o Alex explica bem ... não é pra encarecer o curso mas tá barato pelo que ele se propõe a fazer e a ensinar... já fiz cursos de outros porfessores de c# e Javascript... mas não tem comparação a esse...

Curtir · há cerca de um mês

Comentou na [publicação pública](#) de [Alex Fernando Egidio](#)

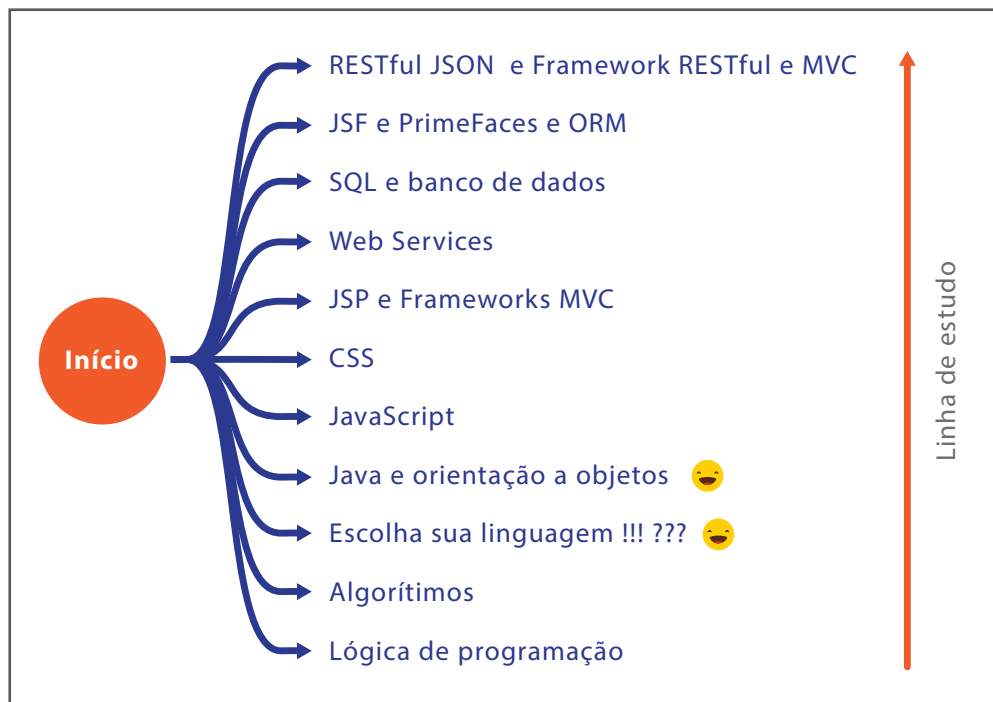


**Rodrigo Apolinário** Conheci seu trabalho ano passado me identifiquei muito com seu método de ensino e de compartilhar conhecimento ainda mais nessa linguagem fascinante que é o Java, acabei adquirindo dois curso e não me arrependo tive dúvidas e vc se propôs em até acessar a máquina pelo Teamviwer parabéns e sucesso Alex Fernando Egidio por nos ajudar tanto javaavançado, agora só depende da gente.

Curtir · 2 respostas · 1 · há cerca de um mês



## Mapa mental → Linha de estudos Java Web



## JavaScript não é Java

Qual é a diferença entre o JavaScript e o Java?

A linguagem de programação JavaScript, desenvolvida pela Netscape, Inc., não faz parte da plataforma Java.

O JavaScript não cria applets ou aplicações independentes. Na sua forma mais comum, o JavaScript fica embutido nos documentos HTML e pode fornecer níveis de interatividade para páginas Web que não são acessíveis com um HTML simples.

Diferenças-chave entre o Java e o JavaScript:

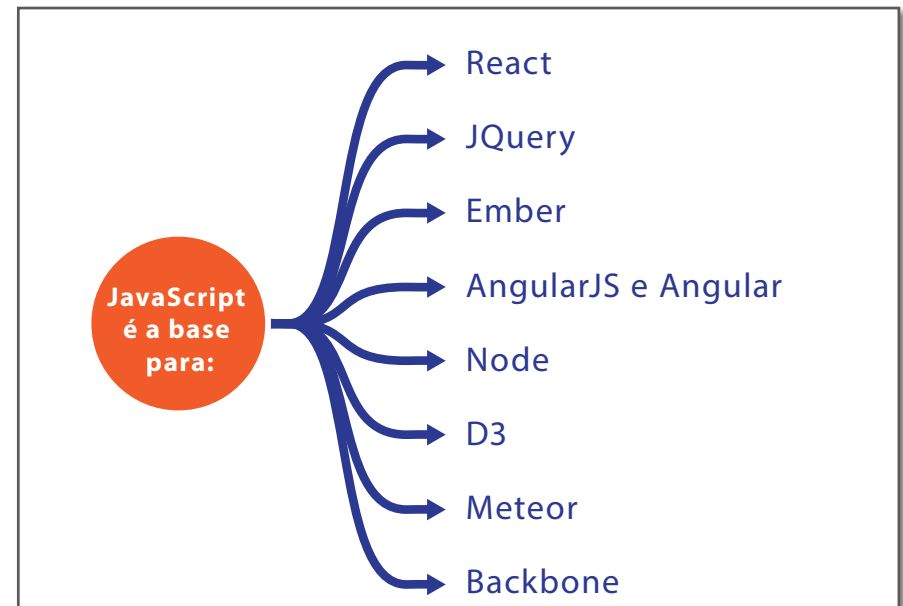
Java é uma linguagem de Programação POO, ao passo que Java Script é uma linguagem de Scripts POO.

Java cria aplicações executadas em uma máquina virtual ou em um browser, ao passo que o código JavaScript é executado apenas em um browser.

O código Java precisa ser compilado, ao passo que os códigos JavaScript estão totalmente em texto atuando juntos com o navegador e páginas bem do lado do cliente.

Com JavaScript sua maior importância é manipular elementos html e acessar recursos por POST, GET, JSON, iterações na página com o cliente, acessar RESTful e outros.

JavaScript é a base para que você domine qualquer outro frameworks que são baseados em JavaScript como na imagem abaixo:



Os mais famosos no mercado são **React**, JQuery e AngularJs.

Existem muitos outros frameworks baseados em JavaScript acima estou citando apenas alguns mais famosos, mas não se assuste sabendo apenas um deles você já se dá muito bem na carreira de programador e se quiser ser um programador Full-Stack → **CLIQUE AQUI**.

O JavaScript é a língua franca da web. Cresceu rapidamente não apenas por causa do que oferece, mas também por causa da comunidade de código aberto que a rodeia.

Os frameworks e bibliotecas acima mencionadas são verificações obrigatórias para qualquer desenvolvedor JavaScript.

Todos eles fornecem alguma forma de explorar JavaScript e desenvolvimento **front-end**.

Essa parte também entra os web design voltados para criação de sites web e portais.

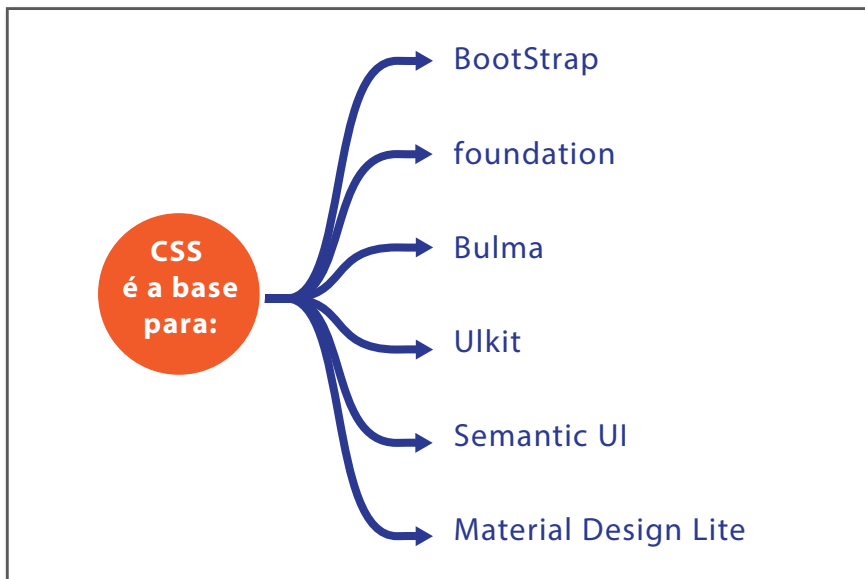
Caso queira entrar no mercado de sites e portais o Curso de Web Design Completo | Aprenda Web Design → **CLIQUE AQUI**.

## Cascading Style Sheets (CSS)

É um simples mecanismo para adicionar estilo (cores, fontes, espaçamento, etc.) a um documento web. Em vez de colocar a formatação dentro do documento, o CSS cria um link para uma página que contém os estilos.

Basicamente o que o CSS faz é deixar o seu sistema bonito, lindão mesmo, quanto mais conhecimento em CSS mais bonita você conseguirá deixar a sua aplicação.

Veja abaixo alguns dos mais famosos frameworks CSS:



Mais uma vez não fique desesperado com esse mundo para se aprender, os mais famosos e mais usados são o **Bootstrap** e o **Material Design Lite**

Hoje é muito comum criarmos uma aplicação inteira sem usar CSS puro, basicamente o que fazemos é escolher o layout que precisamos para o front-end, escolher nosso frameworks JavaScript e criar a lógica no back-end com Java.

Assim nossa aplicação é desenvolvida com uma aparência bonita e bem mais rapidamente.

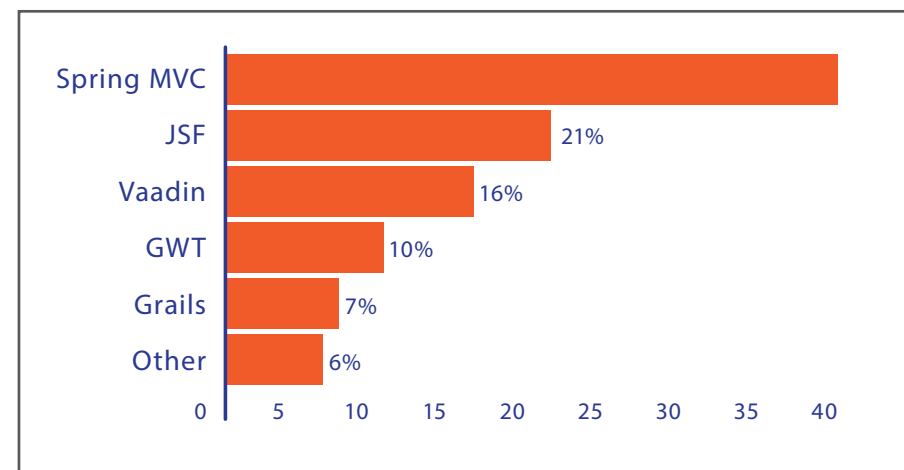


## Frameworks MVC em Java Web

Model-view-controller (MVC), em português modelo-visão-controlador, é um padrão de arquitetura de software (não confundir com um design pattern) que separa a representação da informação da interação do usuário com ele. O modelo (model) consiste nos dados da aplicação, regras de negócios, lógica e funções.

O mundo Java Web é constituída por inúmeros frameworks que nos ajudam e nos dão muitos recursos completos para desenvolvermos nossas aplicações da melhor forma possível.

Abaixo o gráfico mostra os que são mais populares do mercado a nível mundial:



Nesse gráfico temos na liderança e isso não é de agora, Spring MVC e JSF já lideram a mais de anos esse mercado de desenvolvimentos em aplicações corporativas para grandes empresas.

Em nosso curso iremos focar no Spring MVC e JSF e vamos aprender do básico ao avançado o que garantirá que as portas do mercado de trabalho se abram pra você e isso eu garanto 100%.

Então não perca tempo e muito menos o valor promocional do curso. Clique no botão abaixo:

**sim! Iniciar o curso agora mesmo**



**clique aqui!**

## Frameworks ORM Hibernate e JPA

Mapeamento objeto-relacional (ou ORM, do inglês: Object-relational mapping) é uma técnica de desenvolvimento utilizada para reduzir a impedância da programação orientada aos objetos utilizando bancos de dados relacionais.

As tabelas do banco de dados são representadas através de classes e os registros de cada tabela são representados como instâncias das classes correspondentes.

Com esta técnica, o programador não precisa se preocupar com os comandos em linguagem SQL; ele usará uma interface de programação simples que faz todo o trabalho de persistência.

Não é necessária uma correspondência direta entre as tabelas de dados e as classes do programa.

A relação entre as tabelas onde originam os dados e o objecto que os disponibiliza é configurada pelo programador, isolando o código do programa das alterações à organização dos dados nas tabelas do banco de dados.

A forma como este mapeamento é configurado depende da ferramenta que estamos a usar. Como exemplo, o programador que use Hibernate na linguagem Java pode usar arquivos XML ou o sistema de anotações que a linguagem providencia.



## Hibernate

O Hibernate é um framework para o mapeamento objeto-relacional escrito na linguagem Java, mas também é disponível em .Net com o nome NHibernate.

### JPA → Java Persistence API

Java Persistence API (ou simplesmente JPA) é uma API padrão da linguagem Java que descreve uma interface comum para frameworks de persistência de dados.

A JPA define um meio de mapeamento objeto-relacional para objetos Java simples e comuns (POJOs), denominados beans de entidade.

## Web Services em Java Web

Web Service é uma solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes. Com esta tecnologia é possível que novas aplicações possam interagir com aquelas que já existem e que sistemas desenvolvidos em plataformas diferentes sejam compatíveis.

Os Web Services são componentes que permitem às aplicações enviar e receber dados.

Cada aplicação pode ter a sua própria “linguagem”, que é traduzida para uma linguagem universal, um formato intermediário como XML, Json, CSV, etc.

Para as empresas, os Web Services podem trazer agilidade para os processos e eficiência na comunicação entre cadeias de produção ou de logística. Toda e qualquer comunicação entre sistemas passa a ser dinâmica e principalmente segura, pois não há intervenção humana.

Essencialmente, o Web Service faz com que os recursos da aplicação do software estejam disponíveis sobre a rede de forma normalizada. Outras tecnologias fazem a mesma coisa; por exemplo, os browsers da Internet acessam as páginas Web disponíveis usando por norma as tecnologias da Internet, HTTP e HTML.

No entanto, estas tecnologias não são bem-sucedidas na comunicação e integração de aplicações. Existe uma grande motivação sobre a tecnologia Web Service pois possibilita que

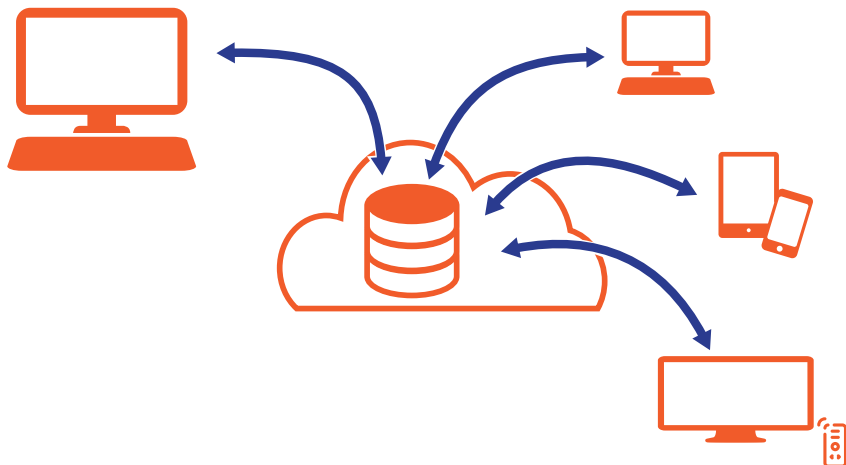
diferentes aplicações comuniquem-se entre si e utilizem recursos diferentes.

Utilizando a tecnologia Web Service, uma aplicação pode invocar outra para efetuar tarefas simples ou complexas mesmo que as duas aplicações estejam em diferentes sistemas e escritas em linguagens diferentes. Por outras palavras, os Web Services fazem com que os seus recursos estejam disponíveis para que qualquer aplicação cliente possa operar e extrair os recursos fornecidos pelo Web Service.

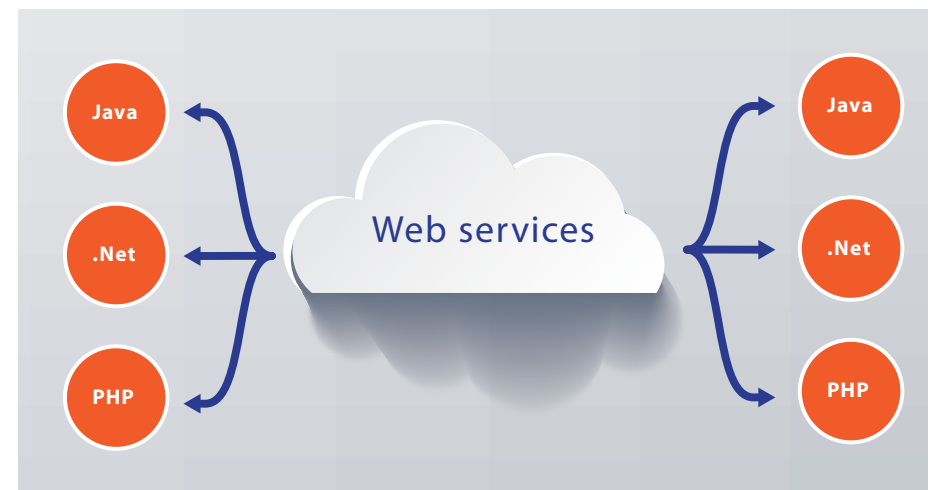
Os Web Services são identificados por um URI (Uniform Resource Identifier), descritos e definidos usando XML (Extensible Markup Language). Um dos motivos que tornam os Web Services atractivos é o facto deste modelo ser baseado em tecnologias standards, em particular XML e HTTP (Hypertext Transfer Protocol).

Os Web Services são utilizados para disponibilizar serviços interativos na Web, podendo ser acessados por outras aplicações usando, por exemplo, o protocolo SOAP (Simple Object Access Protocol) ou hoje em dia JSON mais RESTful.

Abaixo segue uma boa imagem que podemos representar os web services ilustrando que podem ser acessados por vários dispositivos diferentes.



Abaixo segue uma boa imagem que podemos representar os web services ilustrando que podem ser acessados por vários sistemas criados com diferentes linguagens e dispositivos diferentes.



## Banco de dados e SQL

Structured Query Language, ou Linguagem de Consulta Estruturada ou SQL, é a linguagem de pesquisa declarativa padrão para banco de dados relacional (base de dados relacional).

Todos os bancos de dados relacionais suportam consulta em SQL padrão.

Um banco de dados “é uma coleção de dados inter-relacionados, representando informações sobre um domínio específico”, ou seja, sempre que for possível agrupar informações que se relacionam e tratam de um mesmo assunto, posso dizer que tenho um banco de dados.

Podemos exemplificar situações clássicas como uma lista telefônica, um catálogo de CDs ou um sistema de controle de RH de uma empresa.

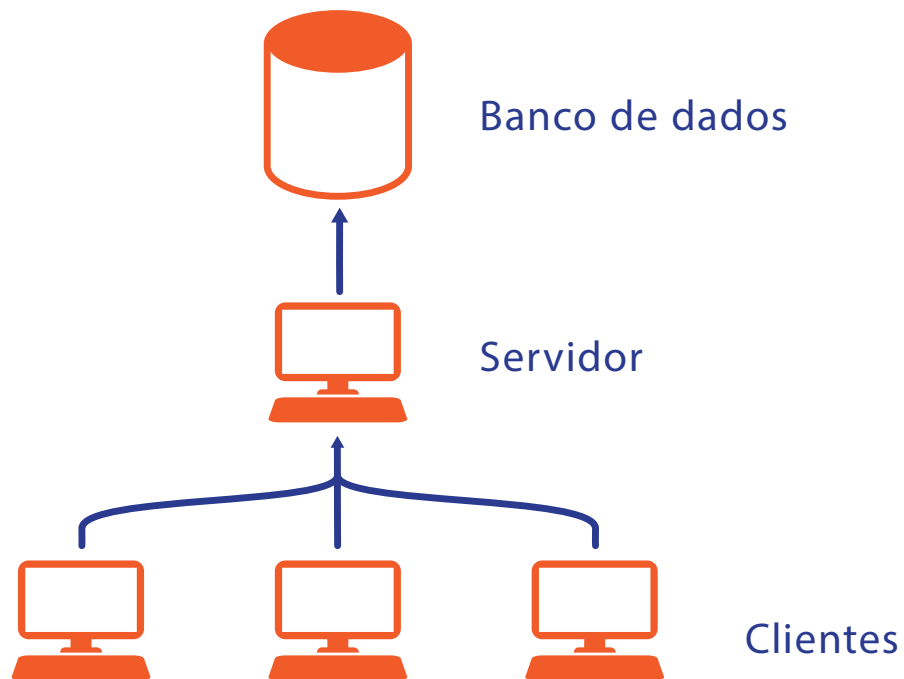
Já um sistema de gerenciamento de banco de dados (SGBD) é um software que possui recursos capazes de manipular as informações do banco de dados e interagir com o usuário. Exemplos de SGBDs são: Oracle, SQL Server, DB2, PostgreSQL, MySQL, o próprio Access ou Paradox, entre outros.

O objetivo final de um banco de dados a armazenar todos os dados e que podem ser acessados por uma aplicação.

## Segurança em aplicações Java Web

Para lançar aplicativos móveis, aplicações web, softwares e sistemas de TI confiáveis, as empresas devem ter um processo de desenvolvimento seguro para resistir a ataques mal-intencionados. Este processo envolve práticas seguras em cada fase do desenvolvimento.

Em nosso curso aprenderemos Spring Security que é o maior e mais usado no mundo inteiro.



## Frameworks de Componentes e MVC

Talvez você ainda não conheça nada sobre isso, ou talvez já ouviu falar, mas ainda está perdido e não sabe como começar.

Hoje é muito comum JSF e PrimeFaces estar na lista de pré-requisitos de boas vagas de emprego para programadores Java.

Então, deixe eu te ajudar de uma vez por todas para você começar hoje mesmo e se tornar um programador melhor e mais requisitado! ;)

## O que é JSF?

Antes de entender o que é PrimeFaces, você precisa saber o que é JSF (JavaServer Faces).

JSF é uma tecnologia para desenvolvimento web que utiliza um modelo de interfaces gráficas baseado em eventos.

A tecnologia foi definida pelo JCP (Java Community Process), que é o mecanismo que cria especificações de tecnologias Java padronizadas.

Isso torna JSF um padrão de desenvolvimento e facilita o trabalho dos fornecedores de ferramentas, ao criarem produtos que valorizam a produtividade no desenvolvimento de interfaces visuais.

## O que é PrimeFaces?

JSF fornece alguns componentes para os desenvolvedores, como campos de entrada de dados, botões, tabelas de dados, etc.

A grande questão é que esses componentes são “puros”, simples, sem estilos visuais.

Isso não é um problema, se você quer criar o layout do zero da sua aplicação e de todos os componentes (ou ainda usar algum framework CSS para isso, como Bootstrap).

Mas para quem busca alta produtividade e qualidade visual praticamente sem trabalho extra, precisa conhecer o PrimeFaces.

PrimeFaces é uma bibliotecas de componentes ricos para aplicações criadas com JavaServer Faces.

É a biblioteca de componentes para JSF mais popular no mundo!

**Agora vou mostrar a você as ferramentas e frameworks que ensinarei no meu treinamento de Formação em Java Web Completa.**



Abaixo todas as tecnologias mais usadas no mercado para desenvolvimento de sistemas em Java para web, todas trabalhando juntas em só projeto. Economize tempo tendo um projeto e curso completo passo a passo sem perder tempo.

### 1. Spring Framework (MVC e RESTful)

Framework mais utilizado em todo mundo em aplicações web, injeção de dependência, controle de transações, arquitetura em MVC, exemplos de arquitetura de serviços com JQuery, Spring JDBC e trabalhando com JSF, Ajax e arquitetura MVC.

### 2. Spring Security

Segurança em aplicações web, controle de acesso a urls, acesso por papéis de cada usuários.

### 3. JSF 2.2

Produtividade alta com componentes prontos, construa páginas altamente funcionais com pouco código, sendo produtivo com a facilidade do JSF, realize ajax sem javascript de forma simplificada.

### 4. PrimeFaces

Componentes customizados e integrados com JSF, adicione componentes funcionais ao projeto, este projeto web conta com um ótimo exemplo de carregamento por demanda em tabelas de dados nas páginas da aplicação.

### 5. IReport

Relatórios com a mais famosa ferramenta de relatórios em Java.

## 6. JQuery

Facilitando as operações nas páginas e Ajax super produtivo.

## 7. Facelets

Templates e componentes genéricos em JSF.

## 8. Hibernate

Alta produtividade com o mais famoso frameworks de persistencia de dados.

## 9. JPA

Alta produtividade junto com o Hibernate.

## 10. Hibernate Envers

Gerencia o log de alterações de todos os registro da base de dados.

## 11. Google Charts

Gráficos perfeitos com este frameworks da Google.

## 12. FlyWay

E também da Google FlyWay gerencie versões da sua base de dados e executa SQL que você necessita para a sua aplicação durante a inicialização da aplicação.

## 13. BootStrap

Bootstrap é um framework web com código-fonte aberto para desenvolvimento de componentes de interface e front-end para sites e aplicações web usando HTML, CSS e JavaScript

#### 14. AngularJS

AngularJS é um framework JavaScript código aberto, mantido pelo Google, que auxilia na execução de single-page applications.

#### 15. PgAdmin e SQL

O pgAdmin é um software gráfico para administração do SGBD PostgreSQL disponível para Windows e UNIX. É uma ferramenta gráfica desenvolvida pela equipe de desenvolvimento do PostgreSQL.

#### 16. Eclipse e Apache Tomcat

Eclipse é a IDE mais utilizada juntos com o servidor mais usado no mundo.

Então, o que está esperando pra aprender todo esse conteúdo mais completo do Brasil? **CLIQUE AQUI**

**sim! Iniciar o curso agora mesmo**

 **clique aqui!**

# Agora você irá conhecer cada aula do curso

Vamos iniciar a lista das aulas do curso Java Web.

Legenda: ● Módulo ▶ Vídeo

- Dicas para uma ótima experiência com o curso
  - ▶ Vídeo de dicas
  - ▶ Tenha uma internet com boa conexão
- Apresentação do curso
  - ▶ Vídeo de apresentação
  - ▶ Obtendo suporte ao curso
- Configuração do ambiente Java
  - ▶ A versão do Java deve ser seguida e obedecida
  - ▶ Conhecendo a JRE e a JDK
- ▶ Download das ferramentas de desenvolvimento e E-BOOK
- ▶ Download da JDK e do Eclipse IDE - Kit de desenvolvimento em Java
- ▶ Instalando a JDK
- ▶ Download do Apache Tomcat e PostgreSQL
- ▶ Colocando o Apache Tomcat em seu lugar
- ▶ Variáveis do Eclipse IDE
- ▶ Testando Apache Tomcat e Configurando variáveis de ambiente Java
- ▶ Executando e testando o Apache Tomcat
- ▶ PostgreSQL ou MySQL
- ▶ Instalando o banco de dados PostgreSQL
- ▶ Configurando a IDE eclipse
- ▶ Configurando Apache Tomcat na IDE Eclipse
- ▶ Criando um projeto de exemplo no Eclipse
- ▶ Importando um projeto de exemplo no Eclipse
- ▶ Download do Eclipse Oxygen - versão mais atual
- ▶ Configurando Apache Tomcat 9 no Eclipse IDE
- ▶ Importando um NOVO projeto de exemplo no Eclipse - Resolvendo configurações

- ▶ Instalando o banco de dados MySQL
- ▶ Debug no Eclipse IDE
- ▶ Agradecimento do módulo de ambiente
- ▶ Deixe uma sugestão de melhoria para o curso
- ▶ Reporte algum erro que tenha acontecido
- Instalação do Ambiente e Projeto
  - ▶ Instalação do projeto e ambiente Java
- DICAS e opiniões pessoais sobre a carreira #1
  - ▶ Minha história como desenvolvedor é igual à sua?
  - ▶ Programador iniciante e a documentação do Java
  - ▶ JUnit salvou minha equipe
  - ▶ Java SE desktop vale a pena investir no Swing?
- Servlets e JSP antes do JSF
  - ▶ Site Oficial do banco do Brasil é em JSP (Java) - JavaServer Pages
  - ▶ Introdução ao JSP
  - ▶ Instalando Tomcat
  - ▶ Criando projeto em JSP
  - ▶ Passando parâmetros
  - ▶ Tag Expressão
  - ▶ Tag declarativa
  - ▶ Objetos implícitos
  - ▶ Session
  - ▶ Directivas
  - ▶ Include e Tag lib em JSP
  - ▶ Forwards
  - ▶ Include
  - ▶ JavaBeans em JSP
  - ▶ Setters e Getters
  - ▶ Expression language jsp
  - ▶ Criando e entendendo a primeira Servlets
- ▶ Conhecendo o JSTL
- ▶ Filter e conexão em JDBC - Parte 1
- ▶ Filter e conexão em JDBC - Parte 2
- ▶ Login com JDBC e Servlets
- ▶ Inserindo usuário em JDBC e JSP - Parte 1
- ▶ Inserindo usuário em JDBC e JSP - Parte 2
- ▶ Aprenda quando será criado classes Java
- ▶ Excluindo usuário em JDBC e JSP
- ▶ Editando usuário com JDBC e JSP - Parte 1
- ▶ Editando usuário com JDBC e JSP - Parte 2
- ▶ DICA - Como lidar com erros sem se apavorar
- ▶ Capturando exceções em Ajax com jQuery e Servlets JSP - Parte 1
- ▶ Capturando exceções em Ajax com jQuery e Servlets JSP - Parte 2
- ▶ Capturando exceções em Ajax com jQuery e Servlets JSP - Parte 3

- ▶ Diferença em usar o C:OUT do JSTL
- ▶ Filter de autenticação de usuário - Parte 1
- ▶ Filter de autenticação de usuário - Parte 2
- ▶ Filter de autenticação de usuário - Parte 3
- ▶ Filter de autenticação de usuário - Parte 4
- ▶ Encerrando a Sessão e o acesso do usuário ao sistema
- ▶ Load page com jQuery
- ▶ Progress bar javascript jQuery - Parte 1
- ▶ Progress bar javascript jQuery - Parte 2
- ▶ Upload de Imagens - Parte 1
- ▶ Upload de Imagens - Parte 2
- ▶ Datatable jQuery server side - Parte 1
- ▶ Datatable jQuery server side - Parte 2
- ▶ Datatable jQuery server side - Parte 3
- ▶ Datatable jQuery server side - Parte 4
- ▶ Exportando PDF e Excel com JSP e Servlets - Parte 1
- ▶ Exportando PDF e Excel com JSP e Servlets - Parte 2
- ▶ Exportando PDF e Excel com JSP e Servlets - Parte 3
- ▶ Exportando PDF e Excel com JSP e Servlets - Parte 4
- ▶ Calendar jQuery - Parte 1
- ▶ Calendar jQuery - Parte 2
- ▶ Múltiplos bancos de dados em JDBC
- ▶ Trabalhando com datas
- ▶ Criando um Gantt chart - Parte 1
- ▶ Criando um Gantt chart - Parte 2
- ▶ Criando um Gantt chart - Parte 3
- ▶ Criando um Gantt chart - Parte 4
- ▶ Criando um Gantt chart - Parte 5
- ▶ Deixe uma sugestão de melhoria para o curso
- ▶ Reporte algum erro que tenha acontecido
- DICAS e opiniões pessoais sobre a carreira #2
  - ▶ Por que eu uso o eclipse a 10 anos
  - ▶ Versões de frameworks e a relação com o mercado
- Introdução ao JSF
  - ▶ Arquitetura da evolução web
  - ▶ O que são especificações
  - ▶ Protocolo HTTP - GET e o RESPONSE
  - ▶ Protocolo HTTP - POST e o PUT
  - ▶ Protocolo HTTP – DELETE
  - ▶ O que é o JSF - JavaServer Faces
  - ▶ Principais componentes JSF
  - ▶ Quem criou o JSF?
  - ▶ Bibliotecas que completam o JSF
  - ▶ Criando primeiro projeto JSF
  - ▶ Configurando o Path do Maven no Projeto

- ▶ Entendendo o ManagedBean JSF
- ▶ Criando primeiro ManagedBean JSF
- ▶ Criando primeira página XHTML – JSF
- ▶ Corrigindo erros de biblioteca no projeto
- ▶ Entendendo o código da página JSF
- ▶ HTML junto com JSF
- ▶ Expression Language JSF
- ▶ Ligando atributos com expression language JSF
- ▶ O que são escopos dos ManagedBean?
- ▶ Entendendo cada escopo ManagedBean
- ▶ Request Scoped
- ▶ View Scoped
- ▶ Session Scoped
- ▶ Application Scoped
- ▶ Backing bean componentes html JSF
- ▶ Ciclo de vida do JSF
- ▶ Arquivo FacesConfig.xml do JSF
- ▶ Arquivo Web.xml do JSF
- ▶ Navegação implícita em JSF
- ▶ Navegação dinâmica em JSF
- ▶ Redirecionamento de URL em JSF
- ▶ Navegação explícita em JSF
- ▶ Propriedade ID dos componentes JSF
- ▶ Eventos DHTML com JSF
- ▶ Inputtext, Inputsecret e Inputtextarea - Componentes de entrada em JSF
- ▶ DICA - Fechando as TAGS JSF mais rapidamente
- ▶ Outputtext e outputLabel - Diferença entre eles em JSF
- ▶ Deixe uma sugestão de melhoria para o curso
- ▶ Reporte algum erro que tenha acontecido
- Introdução ao JPA e Hibernate - Persistência de dados
  - ▶ O que é persistência de dados?
- ▶ Mapeamento objeto relacional
- ▶ Porque usar ORM?
- ▶ JPA e o Hibernate
- ▶ JPA e o Hibernate com Maven
- ▶ Mapeamento básico com JPA e Hibernate
- ▶ Persistence.xml do JPA
- ▶ Testando o JPA e a criação de tabelas
- ▶ Classe JPA Util
- ▶ Criando um DAO genérico
- ▶ Salvando os dados no banco de dados
- ▶ Corrigindo vários problemas em projetos
- ▶ Merge e Update com JPA e JSF
- ▶ Delete com JPA e JSF
- ▶ Resolvendo problemas ao remover objeto
- ▶ DataTable - Carregando dados com JSF e JPA



- ▶ Editar com `setPropertyActionListener` JSF e JPA
- ▶ Remover com `setPropertyActionListener` JSF e JPA
- ▶ Deixe uma sugestão de melhoria para o curso
- ▶ Reporte algum erro que tenha acontecido
- Aprofundando em JSF e JPA
  - ▶ Aprofundando em JSF e JPA
  - ▶ Melhorando o `DataTable` JSF
  - ▶ Graphic image JSF - Colocando imagens na página
  - ▶ `SelectOneMenu` em JSF - Selecione uma opção em uma lista
  - ▶ `SelectManyMenu` em JSF - Selecione várias opções em uma lista
  - ▶ `SelectBooleanCheckBox` em JSF - Selecione verdadeiro ou falso
  - ▶ Limpando formulários com `type reset` do JSF
  - ▶ Imagens em botões e links
  - ▶ Transformando o cadastro de pessoa em usuário do sistema
  - ▶ Filter para autenticação de usuário logado
  - ▶ Código fonte para download
  - ▶ Criando a rotina de login e autenticando como banco de dados
  - ▶ Mostrando e ocultando de acordo com o perfil do usuário
  - ▶ Menu básico e modelo de lançamento
  - ▶ Criando o `managedBean` de lançamentos
  - ▶ Finalizando o CRUD de lançamentos
  - ▶ Carregando os lançamentos em tela
  - ▶ Editando e removendo lançamentos
  - ▶ Finalizando o controle de acessos
  - ▶ Código fonte para download
  - ▶ Deixe uma sugestão de melhoria para o curso
  - ▶ Reporte algum erro que tenha acontecido
  - ▶ Aprendendo o `SelectOneRadio`
  - ▶ Aprendendo o `Select ManyCheckBox`
  - ▶ Criando mensagens para o usuário em JSF
  - ▶ Criando Templates e evitando repetição de código
  - ▶ Trabalhando com datas e converter de datas
  - ▶ Validações e `TimeZone` em datas no JSF
  - ▶ Validações em campos de entrada de texto
  - ▶ Código fonte para download
- Introdução ao Hibernate Bean Validator
  - ▶ Descobrimos o que é e como será implementado o Bean Validator
  - ▶ Adicionando Bean Validator com Maven
  - ▶ Validando campos simples com Bean Validator
  - ▶ Conhecendo mais anotações e validações do Bean Validator

- ▶ Anotações extras e documentação oficial
- Curso JSF 2.2 e PrimeFaces - Criando toda a estrutura do projeto profissional
  - ▶ Criando o projeto
  - ▶ Hibernate Util
  - ▶ Adicione seu projeto ao SVN ou qualquer outro repositório de versão de códigos fontes
  - ▶ Interface de CRUD do Hibernate
  - ▶ Implementação do CRUD do Hibernate Parte 1
  - ▶ Implementação do CRUD do Hibernate Parte 2
  - ▶ Códigos das aulas para download
  - ▶ Xml de configuração do hibernate Parte 1
  - ▶ Xml de configuração do hibernate Parte 2
  - ▶ Criando o arquivo Context.xml
  - ▶ Criando o Util Framework
  - ▶ Enum de permissão de acessos
  - ▶ Criando a anotação @IdentificaCampoPesquisa
  - ▶ Criando a classe ObjetoCampoConsulta
  - ▶ Criando o ViewScope do Spring Parte 1
  - ▶ Criando o ViewScope do Spring Parte 2
  - ▶ Códigos das aulas para download
  - ▶ Classes de Mensagens em JSF e PrimeFaces
  - ▶ Criando a classe de Datas úteis com junit
  - ▶ Criando a classe ReportUtil para geração e impressão dos relatórios
  - ▶ Criando a interface de CRUD ActionViewPadrao
  - ▶ Criando a abstração para os ManagedBeans JSF
  - ▶ Códigos das aulas para download
  - ▶ Criando o BeanReportView responsável por injetar os dados nos relatórios
  - ▶ Criando as classes responsáveis por manipular as exceções em JSF
  - ▶ Criando o ContextLoaderListener do Spring Frameworks
  - ▶ Configurando Hibernate Envers
  - ▶ Códigos das aulas para download
  - ▶ Criando o Filter FilterOpenSessionInView
  - ▶ Criando a estrutura de pastas de páginas web JSF
  - ▶ Deixe uma sugestão de melhoria para o curso
  - ▶ Reporte algum erro que tenha acontecido
- Curso JSF 2.2 e PrimeFaces - Declarando as configurações de cada Framework
  - ▶ Configurando JSF e PrimeFaces Parte 1
  - ▶ Configurando JSF e PrimeFaces Parte 2
  - ▶ Configurando JSF e PrimeFaces Parte 3

- ▶ Criando a tela de acesso negado
  - ▶ Criando a tela de página não encontrada
  - ▶ Criando a tela de erro genérico
  - ▶ Códigos das aulas para download
  - ▶ Páginas iniciais configuração básica
  - ▶ Criando o Faces Config XML de configuração do JSF
  - ▶ Configurando Servlet.xml do Spring
  - ▶ Criando applicationContext.xml do Spring Frameworks - Parte 1
  - ▶ Criando applicationContext.xml do Spring Frameworks - Parte 2
  - ▶ Códigos das aulas para download
  - ▶ Criando o applicationContext-security.xml do Spring Security - Parte 1
  - ▶ Criando o applicationContext-security.xml do Spring Security - Parte 2
  - ▶ Finalizando o Web.xml
  - ▶ Iniciando o sistema pela primeira vez
  - ▶ Código as aulas para download
  - ▶ Deixe uma sugestão de melhoria para o curso
  - ▶ Reporte algum erro que tenha acontecido
  - ▶ Criando o Ajax Status
  - ▶ Testando o Ajax Status
  - ▶ Criando Template inicial - Parte 1
  - ▶ Criando Template inicial - Parte 2
  - ▶ Criando Template inicial - Parte 3
  - ▶ Código as aulas para download
  - ▶ Criando Template inicial - Parte 4
  - ▶ Criando Template inicial - Parte 5
  - ▶ Criando Template inicial - Parte 6
  - ▶ Criando Template inicial - Parte 7
  - ▶ Criando Template inicial - Parte 8
  - ▶ Código as aulas para download
  - ▶ Criando Template inicial - Parte 9
  - ▶ Deixe uma sugestão de melhoria para o curso
  - ▶ Reporte algum erro que tenha acontecido
- Curso JSF 2.2 e PrimeFaces - Construindo o projeto
    - ▶ Avisos
    - ▶ CSS e JavaScript
    - ▶ Criando login - Parte 1
    - ▶ Criando login - Parte 2
    - ▶ Criando o encerramento da sessão - Parte 1
    - ▶ Revisão
    - ▶ Estrutura de Persistência
    - ▶ Controller Session de controle de usuário
    - ▶ DAO login de controle de acesso
    - ▶ Serviços de Login
    - ▶ Código as aulas para download
    - ▶ Criando o encerramento da sessão - Parte 2

- Curso de JSF 2.2 com PrimeFaces

- Construindo os cadastros do projeto

- ▶ Criando as classes de modelos do cadastro de cidades
    - ▶ Criando as classes de serviços cidade
    - ▶ Criando tela de cidades Parte 1
    - ▶ Criando tela de cidades Parte 2
    - ▶ Código as aulas para download
    - ▶ Criando tela de cidades Parte 3
    - ▶ Criando tela de cidades Parte 4
    - ▶ Criando tela de cidades Parte 5
    - ▶ DataTable PrimeFaces Parte 1
    - ▶ DataTable PrimeFaces Parte 2
    - ▶ DataTable PrimeFaces Parte 3
    - ▶ Criando Componentes com Composite Parte 1
    - ▶ Código as aulas para download
    - ▶ Criando Componentes com Composite Parte 2

- ▶ Criando Componentes com Composite Parte 3
    - ▶ Criando Componentes com Composite Parte 4
    - ▶ Criando Componentes com Composite Parte 5
    - ▶ Finalizando cadastro de cidade Parte 6
    - ▶ Corrigindo novos métodos implementados
    - ▶ Código as aulas para download
    - ▶ Deixe uma sugestão de melhoria para o curso
    - ▶ Reporte algum erro que tenha acontecido


- Curso de JSF 2.2 com PrimeFaces - Relatórios e Cadastros

- ▶ Criando report default
    - ▶ Corrigindo o report default
    - ▶ Criando o relatório de cidades
    - ▶ Exportando o relatório em vários formatos

- ▶ Código as aulas para download
    - ▶ Tornando o módulo de relatório genérico
    - ▶ Tela de pesquisa Parte 1
    - ▶ Tela de pesquisa Parte 2
    - ▶ Tela de pesquisa Parte 3
    - ▶ Tela de pesquisa Parte 4
    - ▶ Tela de pesquisa Parte 5
    - ▶ Tela de pesquisa Parte 6
    - ▶ Tela de pesquisa Parte 7
    - ▶ Carregamento por demanda Parte 1
    - ▶ Carregamento por demanda Parte 2
    - ▶ Carregamento por demanda Parte 3
    - ▶ Carregamento por demanda Parte 4
    - ▶ Alterar Senha Parte 1
    - ▶ Alterar Senha Parte 2
    - ▶ Código as aulas para download
    - ▶ Alterar Senha Parte 3
    - ▶ Alterar Senha Parte 4
    - ▶ Corrigindo erro Hibernate

- ▶ Deixe uma sugestão de melhoria para o curso
- ▶ Reporte algum erro que tenha acontecido
- Criando tela de mensagem
  - ▶ Criando tela de mensagem - Parte 1
  - ▶ Criando tela de mensagem - Parte 2
  - ▶ Criando tela de mensagem - Parte 3
  - ▶ Criando tela de mensagem - Parte 4
  - ▶ Criando tela de mensagem - Parte 5
  - ▶ Criando tela de mensagem - Parte 6
  - ▶ Criando tela de mensagem - Parte 7
  - ▶ Criando tela de mensagem - Parte 8
  - ▶ Criando tela de mensagem - Parte 9
  - ▶ Criando tela de mensagem - Parte 10
  - ▶ Código as aulas para download
  - ▶ Deixe uma sugestão de melhoria para o curso
  - ▶ Reporte algum erro que tenha acontecido
- Criando o cadastro de Funcionário
  - ▶ Criando condição de pesquisa
  - ▶ Corrigindo tela de pesquisa
  - ▶ Criando a tela de pesquisa de funcionário
  - ▶ Cadastro de funcionário - Parte 1
  - ▶ Cadastro de funcionário - Parte 2
  - ▶ Cadastro de funcionário - Parte 3
  - ▶ Cadastro de funcionário - Parte 4
  - ▶ Relatório de funcionário
  - ▶ Criando relatório de acessos com sub-reports
  - ▶ Restringindo o CPF no cadastro de funcionário
  - ▶ Deixe uma sugestão de melhoria para o curso
  - ▶ Reporte algum erro que tenha acontecido
- Criando o cadastro de título
  - ▶ Criando a entidade do título – Parte 1
  - ▶ Códigos fontes aulas para download
  - ▶ Criando a entidade do título - Parte 2
  - ▶ Criando a entidade do título - Parte 3
  - ▶ Criando a entidade do título - Parte 4
  - ▶ Criando a entidade do título - Parte 5
  - ▶ Códigos fontes aulas para download
  - ▶ Deixe uma sugestão de melhoria para o curso
  - ▶ Reporte algum erro que tenha acontecido
- Criando o gráfico de título
  - ▶ Criando a entidade do título - Parte 1
  - ▶ Criando a entidade do título - Parte 2

- ▶ Criando a entidade do título - Parte 3
- ▶ Criando a entidade do título - Parte 4
- ▶ Criando a entidade do título - Parte 5
- ▶ Deixe uma sugestão de melhoria para o curso
- ▶ Reporte algum erro que tenha acontecido
- Entendendo o Flyway
  - ▶ Como o FlyWay trabalha no projeto
  - ▶ Controlando o banco com Flyway
- Implantando o projeto no Tomcat
  - ▶ Gerando War e implantando no Tomcat
- Aulas extras em JavaServer Faces (JSF)
  - ▶ Upload de arquivos com JSF 2.2
  - ▶ Lendo e gravando csv em JSF 2.2
- ▶ Download do csv em JSF 2.2
- ▶ Restringindo o CPF no cadastro de funcionário
- ▶ Parâmetros e SQL dentro o iReport
- Entendendo o MVC + JPA + JSON + Relacionamentos em Hibernate e JPA
  - ▶ Entendendo relacionamentos uma para muitos e muitos para um
  - ▶ Entendendo relacionamentos muitos para muitos
  - ▶ Estrutura do projeto
  - ▶ Um cadastro de pessoa em MVC + JSON e AJAX
  - ▶ Estrutura JSON
  - ▶ Cadastro de pessoas com MVC + JSON
  - ▶ Download do projeto
  - ▶ Deixe uma sugestão de melhoria para o curso
  - ▶ Reporte algum erro que tenha acontecido
- AngularJS 1.6 + Spring Frameworks RESTful
  - ▶ Download e apresentação do projeto completo que será desenvolvido
  - ▶ Introdução ao AngularJS 1.6
  - ▶ Criando um projeto AngularJS
  - ▶ Propriedades do AngularJS
  - ▶ Correção para as próximas aulas
  - ▶ Criando um formulário simples em AngularJS
  - ▶ Finalizando um formulário simples em AngularJS
  - ▶ Expressões em AngularJS
  - ▶ Diretivas no AngularJS
  - ▶ Filters em AngularJS
  - ▶ O que são services em AngularJS

**sim! Iniciar o curso agora mesmo**  
 **clique aqui!**

Mais de 2.000 alunos felizes

- ▶ Usando o Http e conhecendo o Ajax com AngularJS
- ▶ Montando tabelas de dados
- ▶ Criando o componente Select com AngularJS e Html
- ▶ Entendendo sobre o DOM
- ▶ Conhecendo o Resources Ajax com AngularJS
- ▶ Conhecendo um pouco de Events no AngularJS
- ▶ Validando formulários
- ▶ Animações nas páginas
- ▶ Formatando um campo de moeda com AngularJS
- ▶ Conhecendo a injeção de dependência com AngularJS
- ▶ Criando filter em tabelas de dados
- ▶ Instalando o Layout BootStrap em nosso projeto
- ▶ Testando as Rotas do AngularJS
- ▶ Criando e conhecendo DataSource no Tomcat Apache
- ▶ Configurando o Hibernate
- ▶ Configurando o Spring Frameworks
- ▶ Conhecendo o padrão Model - DAO e Controller
- ▶ Iniciando a criação da lista de cliente
- ▶ Iniciando a criação da lista de cliente - Parte 2
- ▶ Iniciando o cadastro de clientes
- ▶ Editando o cliente e entendendo o RESTful
- ▶ Adicionando notificação ao nosso sistema
- ▶ Criando componentes
- ▶ Criando o select de cidades e estados
- ▶ Resolvendo problema com Google Chrome e o Select (Combo)
- ▶ Conhecendo o jQuery Mask
- ▶ Criando o Upload de Imagens para nosso projeto
- ▶ Paginação e desempenho com front-end e back-end
- ▶ Finando o cadastro de clientes
- ▶ Iniciando o cadastro de fornecedores
- ▶ Criando o cadastro de livros da nossa loja
- ▶ Melhorando os cadastros criados
- ▶ Iniciando a criação da loja de livros
- ▶ Criando a loja de livros - Parte 2
- ▶ Criando a loja de livros - Parte 3
- ▶ Criando a loja de livros - Parte 4
- ▶ Adicionando cliente a nossa loja de livros
- ▶ Finalizar o pedido de nossa loja de livros
- ▶ Imprimindo nosso pedido – Parte 1
- ▶ Imprimindo nosso pedido - Parte 2
- ▶ Resolvendo problemas em relatórios
- ▶ Criando gráfico de pedidos com Google Chart
- ▶ Implementando segurança nosso projeto
- ▶ Implementando segurança nosso projeto - Spring Security
- ▶ Implantando nossa aplicação no servidor



- ▶ Escondendo parte com Ng-Hide
- ▶ Separando os controllers no AngularJS
- ▶ UTF-8 e codificação de caracteres
- ▶ Revisando as configurações do nosso projeto
- ▶ Redirecionando de acordo com o perfil do usuário
- ▶ Deixe uma sugestão de melhoria para o curso
- ▶ Reporte algum erro que tenha acontecido

- **Spring Boot** criando API RESTful - Curso Extra

- ▶ Conhecendo o curso de APIs com Spring Boot → **CLIQUE AQUI**
- ▶ DEMONSTRAÇÃO - Entendendo a arquitetura RESTful
- ▶ DEMONSTRAÇÃO - Entendendo o Spring Boot
- ▶ DEMONSTRAÇÃO - Criando um projeto com o Spring Boot
- ▶ DEMONSTRAÇÃO - Criando Controller com Spring Boot

- ▶ **Grade curricular completa** do curso de Spring Boot API RESTful

- Imprimindo o seu certificado de conclusão do curso
  - ▶ Imprimir certificado digital

Nossa, é muito conteúdo mesmo, viu como eu disse que é o maior curso do Brasil.

Você só precisa deste curso para se tornar um programador Java Web de sucesso.

**sim! Iniciar o curso agora mesmo**

 **clique aqui!**

Inicie agora o curso clicando no botão acima.

Estou esperando você pra estudar comigo na minha sala de aula on-line. Abraços!



**Baixe o conteúdo gratuito do curso**

**Conheça outros curso na área de programação  
que são TOP 1 no Brasil.**



## Full-Stack Web PHP

Domine as principais tecnologias do mercado e se torne Desenvolvedor Full-Stack, com salários na faixa dos R\$5.000,00.

1.600 alunos ⌚ 75h 🏆 certificado

[CONHECER O CURSO](#)



## E-Book Full-Stack PHP

Este livro digital irá lhe ensinar tudo sobre as principais tecnologias web para você dar o ponta-pé inicial no Universo Full-Stack.

478 alunos ⌚ e-book

**Gratuito**

[RECEBER AGORA](#)



## PHP Básico ao Expert

Este é o curso mais completo do Brasil. Você aprenderá do básico, ao nível profissional no PHP com acesso ilimitado.

15.425 alunos ⌚ 150h 🏆 certificado

[CONHECER O CURSO](#)



## Profissional Cloud

Porque o Profissional Cloud é tão valioso? E porque precisa se preocupar muito menos com tarefas manuais?

498 alunos ⌚ e-book

**Gratuito**

[CONHECER O CURSO](#)



## Full-Stack Web Java EE

Mais de 600 aulas em PrimeFaces, JSF, Spring, Hibernate, JPA, Jreport, CDI e muitos mais. Iremos do básico ao avançado.

2.985 alunos ⌚ 100h 🏆 certificado

[CONHECER O CURSO](#)



## Aprender a Programar

Descubra os 8 passos exatos para aprender a programar do absoluto zero! mesmo que ache que programar é difícil!

759 alunos ⌚ 2h

**Gratuito**

[CONHECER O CURSO](#)



## Games com JAVA

São 7 jogos desenvolvidos no curso e com uma própria game engine, desenvolvida também no curso completo.

648 alunos ⌚ 40h 🏆 certificado

[CONHECER O CURSO](#)



## Crie App Kotlin básico

Neste eBook, você irá criar o seu primeiro projeto Android em Kotlin em pouco mais de 10 minutos.

5.842 alunos ⌚ e-book

**Gratuito**

[CONHECER O CURSO](#)

# Abraços de quem quer seu sucesso!

## Alex Fernando Egidio

*Desenvolvedor Sênior em Java*

Me siga nas redes sociais



*Compartilha esse e-book com seus amigos, OK!*

*Tchau :)*

**sim! Iniciar o curso agora mesmo**



**clique aqui!**

**ATENÇÃO! AS VAGAS ESTÃO SE  
ESGOTANDO RAPIDAMENTE!**

Entre em contato comigo



javaavancado@javaavancado.com



alex.fernando.egidio@gmail.com



(45) 9 9979-5800



(44) 9 8821-2355



<https://www.javaavancado.com>

Cnpj: 26.934.453/0001-89

Rua: Pioneiro Antônio de Ganello

Bairro: Jardim Dias I

Cidade: Maringá/PR

Cep: 87025-758

Meu objetivo é fazer você entrar  
no mercado de trabalho com todo  
o conhecimento que tenho pra  
passar a você.