

**Universidade do Estado do Amazonas**

**Codificação - Boas práticas e estrutura das pastas**

**Manaus - AM  
2019**

# SUMÁRIO

<b>Indentação</b>	<b>3</b>
<b>Comentários e Documentação</b>	<b>3</b>
<b>Convenções de nomes</b>	<b>3</b>
<b>Tratamento de erros (try/catch)</b>	<b>3</b>
<b>Padrões de Projetos (Design Patterns)</b>	<b>4</b>
<b>Testar e Depurar</b>	<b>4</b>
<b>Tamanho</b>	<b>4</b>

## **Introdução**

Ao escrever os códigos para um sistema, o desenvolvedor deve ter em mente que seus scripts devem estar legíveis tanto para si quanto para outros que possam ler e contribuir no desenvolvimento do projeto.

Além disso, ao fazer uso das boas práticas de programação, seus códigos possuem maiores chances de produzir menos erros. Caso ocorra algum erro de execução, será mais fácil de identificá-lo.

## **Indentação**

Código não indentado é aquele que está desalinhado e desorganizado. Em python, por exemplo, a indentação é necessária para seu funcionamento, mas em outras linguagens como PHP ou javascript, isso não influencia na hora de compilar o código.

Para facilitar a manutenção e até mesmo de aprendizagem de um código ela é recomendada. Vista a complexidade de alguns códigos, até mesmo profissionais experientes precisam analisar com calma e paciência um algoritmo para entendê-lo. Além de facilitar a leitura, deixa o código mais limpo e bonito.

## **Comentários e Documentação**

Comentários podem ser usados de duas formas:

1. Explicar o algoritmo ou a lógica usada;
2. Documentar o projeto, descrevendo a especificação do código.

Ao criar um método, procure sempre informar qual é a finalidade do mesmo e se ele irá retornar ou não alguma informação. Mas o excesso de comentários pode deixar o código poluído, difícil de ler e de atualizar, para isso recomenda-se colocar textos curtos e que sejam relevantes para a leitura e compreensão do código.

## **Convenções de nomes**

Colocando nomes adequados e padronizados, a leitura do código se torna muito mais intuitiva para o programador, indicando, por exemplo, o que uma variável irá armazenar ou o que um determinado método irá fazer. Sabendo utilizar dos nomes para variáveis, classes e métodos o ciclo de desenvolvimento fica muito mais fácil.

## **Tratamento de erros (try/catch)**

Todas as linguagens de programação têm suporte ao tratamento de erros, mas nem sempre seu uso é obrigatório. É altamente recomendável o

tratamento de erros, oferecendo assim maior segurança e confiabilidade ao sistema. O básico do tratamento de erros é o sistema mostrar uma mensagem quando uma exceção ocorrer, o que deve ser feito tanto na fase de desenvolvimento, para o programador, quanto para o usuário final, quando o sistema estiver em produção.

## **Padrões de Projetos (Design Patterns)**

O uso da programação em camadas, facilita manter várias pessoas em um mesmo projeto de software, permitindo dividir melhor as funcionalidades da aplicação. Dessa forma, enquanto algumas pessoas trabalham na camada de visão, outras poderão manipular o código relacionado às regras de negócio ou que controlam o fluxo das informações.

Para isso utiliza-se do MVC (Model-View-Controller):

1. Model: será responsável pela manipulação dos dados e pelas operações que o sistema irá executar;
2. View: é a camada de apresentação;
3. Controller: também conhecido como Controlador, é responsável por gerenciar todas as informações no sistema, definindo o comportamento do programa.

## **Testar e Depurar**

Procure executar muitos testes no seu sistema e também aprenda a usar um depurador para analisar o código em tempo de execução, facilitando a localização de defeitos.

## **Tamanho**

Evite classes muito longas para evitar complicações na leitura e depuração.

## Referências

- Boas práticas de programação. Disponível em <<https://programandotudopython.blogspot.com>> Acesso em 24 de março de 2019;
- Boas práticas de Programação. Disponível em <<https://www.devmedia.com.br/boas-praticas-de-programacao>> Acesso em 24 de março de 2019;
- Get started with Docker Compose. Disponível em <<https://docs.docker.com>> Acesso em 24 de maio de 2019.