

[Hora do Discovery]

[Desafios de Programação]

[Atividade avaliativa em times, vale 1.0 ponto]



[DESAFIO]

Descoberta de
conhecimento na
prática



Temas:

- Orientação a objetos
- Interface de usuário
- Automação de tarefas
- Banco de Dados
- Análise de dados
- Web scraping
- PyScript
- Armazenamento de dados no navegador

Objetivo:

Criar uma **oficina** de compartilhamento de conhecimentos e práticas relacionadas a Python.

Times:



Criar **nomes** para os times que tenham relação com o **tema** **escolhido**.

Desafio em times.

Usar a prática do Pair Programming (adaptada).

[Explicação]

Prática
Pair Programming

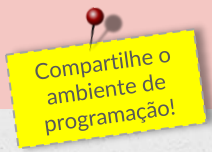
do
eXtreme
Programming (XP)

Programação em pares (*Pair Programming*)

Duas pessoas implementam juntas o código diante do mesmo computador, revezando-se no teclado.

Uma pessoa é o "controlador", escreve o código, enquanto o outro, chamado de "observador" ("navegador"), analisa cada linha do código. Os papéis são revezados e, geralmente, tem-se um iniciante e um desenvolvedor mais experiente.

<https://youtu.be/a8WaP3Fwqa0>



Compartilhe o
ambiente de
programação!

Método:

- (1) **Escolha** do time, conforme o **tema** indicado na sala virtual.
- (2) O time deve fazer uma **busca exploratória** para compreensão do tema de forma conceitual, com **registros** de pesquisa (local de pesquisa, data da pesquisa, síntese dos estudos).
- (3) O time deve **compreender** o tema e **levantar/desenvolver exemplos** de aplicação (se possível, fazer exercício com a turma).
- (4) O time de registrar a **importância** do tema para projetos Python.
- (5) O time de registrar as **limitações e desafios** sobre o tema.

O que temos que entregar na sala virtual?

Apresentação e registros em slides + **vídeo** de apresentação + código-fonte (se necessário).

Postar materiais acima **na sala virtual às 19hs, conforme data da atividade.**



O que temos que entregar na aula?

Apresentação no formato de **oficina dos time** (10 minutos):

- **Registros, importância, exemplos, limitações e desafios.**
- Quanto ao exemplo, se possível, fazer exercício com a turma.

Critérios de avaliação:

- (0,2) **Pensamento computacional** (atende o que pede). **Requisito: atender todas às regras.**
- (0,2) **Originalidade, clareza e confiabilidade** (pautado em referências, capacidade de pensar em problemas e soluções).
- (0,2) **Registros** (referências, data, síntese, importância, exemplos, limitações e desafios).
- (0,2) **Originalidade** (capacidade de pensar em problemas e soluções que envolvam o tema).
- (0,2) **Oficina em aula + vídeo de apresentação** (clareza e profundidade – detalhes - na apresentação, cumprimento dos prazos).



Bom desafio!

[Não execute, pense antes!]



Principais referências para apoiar o desafio:

Foram incluídos links iniciais em cada tema (ver no segundo slide).

O time deve complementar com outros estudos e materiais, mantendo o registros das referências utilizadas.