

# Atividade - Python (TED)

[Expressões regulares](#) são ferramentas extremamente poderosas quando o assunto é **manipulação e extração de padrões em textos**. Para esta atividade, utilize a estrutura de código apresentada em sala, aplicando boas práticas de **Orientação a Objetos em Python**, com o objetivo de ler um arquivo `.txt` contendo um texto de sua escolha e realizar as seguintes operações:

## Requisitos da Classe Principal

A classe principal do projeto deve conter os seguintes métodos:

### 1. Leitura do arquivo de texto

- Ler um arquivo `.txt` contendo múltiplas linhas (pode ser uma redação, carta ou qualquer outro texto da sua escolha).

### 2. Filtrar palavras que começam com determinada letra

- Retornar o conteúdo do arquivo contendo apenas as palavras que se iniciam com uma letra específica.

### 3. Filtrar palavras que contenham determinada letra em qualquer parte

- Retornar o conteúdo do arquivo contendo palavras que possuam determinada letra, independentemente da posição.

### 4. Substituição de vírgulas por pontos

- Substituir todas as vírgulas por pontos no texto lido.

### 5. Extração de datas

- Retornar apenas as datas encontradas no texto, utilizando expressões regulares.

### 6. Implementação de pelo menos um Dunder Method

- Utilize ao menos um [Dunder method](#) para adicionar um comportamento especial à sua classe.

### 7. Sinta-se livre para implementar outras funcionalidades com regex ou vários Dunders

### 8. Tarefa Bônus ★

1. Implemente um método que identifique e oculte informações sensíveis (como e-mails, CPFs ou telefones) no texto

## DICAS IMPORTANTES

- Utilize [tipagem estática](#) sempre que possível (especificando tipos de variáveis e retornos de funções).
- Utilize [Docstrings](#) para documentar cada classe/função criada.

- Use [ambientes virtuais](#) para gerenciar as dependências do seu projeto.
- **Estruture seu projeto em módulos**, como demonstrado em sala:
  - O arquivo `main.py` deve ser responsável por executar o programa.
  - As classes devem ser definidas dentro de uma **pasta principal** do projeto.
- Se estiver utilizando o **VSCode**, considere instalar as seguintes extensões:
  - [Python \(ms-python.python\)](#)
  - [Flake8 \(ms-python.flake8\)](#)
  - [Mypy Type Checker \(ms-python.mypy-type-checker\)](#)
- Se estiver utilizando o **PyCharm**, ele já vem com a maioria dessas ferramentas integradas.
- Caso esteja usando o **Google Collab**, embora ele não ofereça suporte a essas extensões, **as boas práticas de codificação ainda devem ser seguidas**.

## Avaliação

- A atividade poderá ser realizada em grupos de até **três integrantes**.
- Será exigida uma apresentação oral do código desenvolvido, seguida de um momento para arguição conduzido pelo professor e pelos colegas de turma

### Critérios Positivos:

A equipe poderá **receber pontuação adicional** se demonstrar:

1. Domínio técnico e clareza na explicação do código e das decisões tomadas.
2. Aplicação adequada e criativa de expressões regulares.
3. Utilização de **Dunder Methods** de forma funcional e relevante para a proposta.
4. Demonstração de **trabalho em equipe** e distribuição equilibrada das responsabilidades.

### Critérios de Penalização:

A equipe poderá **perder pontos** nos seguintes casos:

1. Não seguir os padrões de codificação e organização solicitados na atividade.
2. Não demonstrar domínio do código, sugerindo que se trata de uma cópia de uma LLM, da internet ou de outro colega.
3. Não respeitar os limites de colaboração (como grupos com mais de três membros).
4. Apresentação incompleta ou ausência de partes funcionais obrigatórias do projeto.