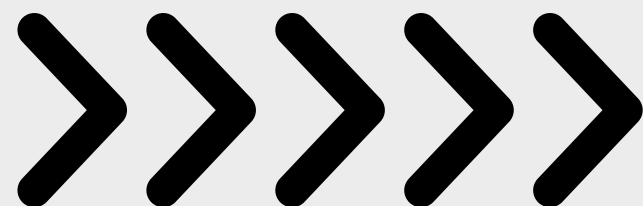




DEBUGEICOMR

Testes com API JSONPlaceholder
Usando a Linguagem R



INTRODUÇÃO AO R



- **R** é uma linguagem de programação amplamente usada para análise de dados, estatísticas e visualização.
- **Foco no R:** Permite uma análise estatística poderosa, com pacotes e bibliotecas voltados para manipulação de dados e modelagem matemática.
- **Uso no mercado:** Amplamente utilizado em ciência de dados, estatística, bioinformática e em diversas áreas de pesquisa.



MINHA APRENDIZAGEM



Aprendizado sobre R: Aprofundei-me na sintaxe do R e em seu ecossistema, buscando melhorar minha capacidade de escrever código mais limpo e eficiente.

Bibliotecas e Ferramentas:

- **httr:** Para fazer requisições HTTP de forma prática e segura.
- **R6:** Para estruturar o código em formato de orientação a objetos.
- **testthat:** Para escrever testes automatizados e garantir que o código esteja funcionando corretamente.





OBJETIVO DO PROJETO



- O objetivo principal deste projeto foi criar uma solução de testes automatizados para interagir com a API JSONPlaceholder.
- Além de testar a API, busquei também aprender a organizar o código de maneira modular, utilizando práticas de qualidade de software.



TECNOLOGIAS E BIBLIOTECAS



TECNOLOGIAS

RStudio: Ambiente de Desenvolvimento Integrado (IDE) para a linguagem R, usado para escrever, executar e debugar código.

R: Linguagem usada para desenvolver a aplicação.

BIBLIOTECAS

httr: Bibliotecas para realizar as requisições HTTP e obter dados da API.

R6: Facilita a criação de classes e objetos, permitindo uma abordagem de programação orientada a objetos em R.

testthat: Biblioteca que ajuda a escrever e organizar testes automatizados, garantindo a qualidade do código.





FUNCIONALIDADES DA CLASSE



- 01** **check_status_jsonplaceholder()** – Verifica o status da API.
- 02** **get_todos()** – Obtém todos os "to-dos" da API.
- 03** **search_todo(id)** – Busca um item específico por ID.
- 04** **search_user(id_user, status)** – Retorna os to-dos de um usuário, com a possibilidade de filtrar por status (completo ou pendente).
- 05** **status_todo(status)** – Filtra os to-dos com base no status de conclusão (completo ou pendente).



TESTES AUTOMATIZADOS



Testes Criados:

- **Verificação de status HTTP:** Garantindo que a API esteja respondendo corretamente.
- **Validação de Retorno:** Verificando se os dados retornados são coerentes e conforme esperado.
- **Erros de Requisição:** Testando como a aplicação lida com falhas de comunicação com a API.
- **Filtros:** Garantindo que os filtros por status (completo/pendente) funcionem corretamente.

Objetivo dos testes: Validar a confiabilidade da aplicação e garantir que a classe funcione conforme o esperado.





CONCLUSÃO

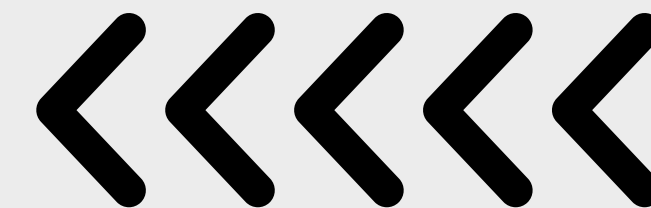


O projeto permitiu uma compreensão mais profunda sobre testes automatizados em R e a construção de uma aplicação organizada e de fácil manutenção.

Próximos Passos: Expandir o projeto com novos endpoints da API e melhorar os testes, incluindo casos de uso mais complexos.

O aprendizado sobre o uso de R6 e testthat foi uma grande conquista e ampliou minha visão sobre como estruturar testes e código em R.





OBRIGADO

Apresentado por Pedro Rufino

