[MATHEUS PEIXOTO RIBEIRO VIEIRA](mailto:matheus.peixoto@aluno.ufop.edu.br) - 22.1.4104

Push do docker:

docker push matheuspeixotoribeirovieira/tp\_05:bcc264.5

Executar docker:

docker run -it -p 5000:5000 matheuspeixotoribeirovieira/tp\_05:bcc264.5

Para executar a aplicação, digite no docker:

python3 application.py

**Link do vídeo:** <https://youtu.be/X1m1ykY2O5g>

**Link da apresentação powerpoint:** [Apresentação de slides](https://docs.google.com/presentation/d/1gzEre9f2gvB7f7bKKIpgbKZaikpfCgjwHOc5RZNfL1g/edit?usp=sharing)

**REST (Representational State Transfer)**

Uma API são regras e protocolos que facilitam a troca de mensagens entre múltiplas aplicações, de forma que a integração entre diferentes linguagens de programação seja facilitada.

Dessa forma, o que acontece é uma aplicação receber uma entrada, processá-la e enviar uma saída, como se fosse uma caixa preta.

Todavia, é necessário que ocorra uma padronização na forma em que entradas e saídas sejam semelhantes, assim até outros usuários podem utilizar essa aplicação.

O REST é uma API muito utilizada para aplicações que necessitam de conversas entre cliente e servidores, dessa forma, o cliente envia um tipo de dado, o servidor processa e retorna uma resposta, sendo que esta conversa é feita por protocolo HTTP.

Nas requisições que o cliente envia teremos:

* Endpoint: O endereço desejado que deseja que ocorra uma requisição, como se fosse uma chamada de função.
* Parâmetros: São enviados pela URL da mensagem para identificar o que quer ser passado, como se fosse uma variável.
* Body: São parâmetros enviados de forma que não seja visível pela URL, de forma que não fique visível para outros usuários que consigam interceptar a mensagem
* Cabeçalhos: Podem possuir chaves para autenticação na API, entre outros.

Juntamente com as respostas da API, também temos as respostas do HTTP, sendo elas Respostas informativas (100 a 199), sucesso (200 a 299), redirecionamento (300 a 399), erros do cliente (400 a 499) e erro interno do servidor (500 a 599).

Ademais, os envios e recebimento de dados, podem ser feitos por arquivos de diferentes tipos, entre eles o JSON, que é o mais famoso e mais utilizado na indústria, o XML, um modelo mais antigo, o HTML, para exibição de telas em navegadores, arquivos de texto, etc.

O funcionamento de um servidor que funciona com rest é muito semelhante com um crud, possuindo equivalências:

* Create (POST): envia informações para serem adicionadas a um banco de dados. Em geral, os dados estão no corpo da requisição.
* Read (GET): responsável pela consulta de informações, como a solicitação de uma página web, ou informações de um ou mais dados. Em geral, os dados de solicitação podem ser vistos pela URL.
* Update (PUT): atualiza informações existentes.
* Delete (DELETE): remove dados.

Referências:

AMAZON. O que é a API RESTful? – Explicação sobre a API RESTful – AWS. Amazon Web Services, Inc. Disponível em: <https://aws.amazon.com/pt/what-is/restful-api/>. Acesso em: 2 fev. 2024.

MOZILLA. ‌Códigos de status de respostas HTTP - HTTP | MDN. MDN Web Docs. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Status>. Acesso em: 2 fev. 2024.

‌IBM TECHNOLOGY. What is a REST API? Disponível em: <https://www.youtube.com/watch?v=lsMQRaeKNDk&ab\_channel=IBMTechnology>. Acesso em: 2 fev. 2024.

REDHAT. O que é uma API REST (Representational State Transfer)? Redhat.com. Disponível em: <https://www.redhat.com/pt-br/topics/api/what-is-a-rest-api>. Acesso em: 2 fev. 2024.

REDHAT. ‌What is an API? Redhat.com. Disponível em: <https://www.redhat.com/en/topics/api/what-are-application-programming-interfaces>. Acesso em: 2 fev. 2024.

CURRY, Caleb. REST API Crash Course - Introduction + Full Python API Tutorial. Disponível em: <https://www.youtube.com/watch?v=qbLc5a9jdXo&t=547s&ab\_channel=CalebCurry>. Acesso em: 3 fev. 2024.