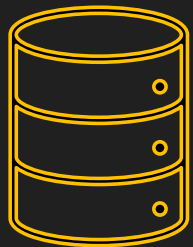


# API

Application  
Programming  
Interface



# API

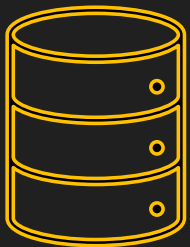


Pede Informações

Código do Lira

Devolve uma resposta

# API



**API do  
Instagram**

**Pede Informações**



**Devolve uma resposta**

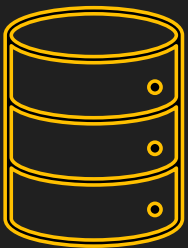


**Código do Lira**

# FERRAMENTAS COM API



# REQUISIÇÕES



**API do  
Instagram**

Requisição



Pede Informações

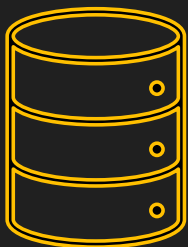


Devolve uma resposta



**Código do Lira**

# REQUISIÇÕES



**API do  
Instagram**

Requisição



Pede Informações



**JSON**

Devolve uma resposta



**Código do Lira**

# JSON

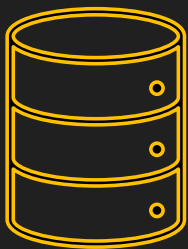
Um simples formato de texto, um padrão

No Python é essencialmente um dicionário Python

Ex: Um usuário do Facebook em formato JSON

```
{"ID254": {"nome": "Lira", "idade": 27, "Amigos": ["ID255",  
"ID256", "ID302"], "foto_perfil": "lirabrabo.png"}}
```

# MAS O QUE É REST?



**API do  
Instagram**

Requisição



Pede Informações



**JSON**

Devolve uma resposta



**Código do Lira**



MAS O QUE É REST?

E REST API?

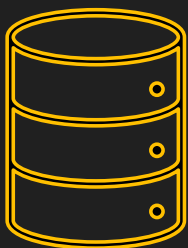
E RESTful API?

# MAS O QUE É REST?

É um padrão que explica como é feita a requisição

## E REST API?

## E RESTful API?



**API do  
Instagram**

**Pede Informações**

**JSON**

**Devolve uma resposta**

**Código do Lira**

**Requisição**

A informação é enviada no formato JSON,  
mas como eu peço essa informação?

# O REST DIZ:

Por meio dos protocolos web:

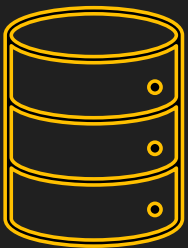
GET -> Para pegar uma informação

POST -> Para criar uma informação

PUT (PATCH) -> Para atualizar uma informação

DELETE -> Para deletar uma informação

# Quero pegar informações de um usuário



**API do  
Instagram**

Pede Informações

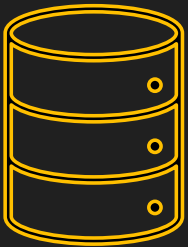
**GET**

Devolve uma resposta

**JSON**

**Código do Lira**

# Quero postar uma foto



**API do  
Instagram**

Envia Informações



**POST JSON**

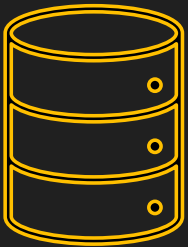
Devolve uma resposta



**JSON**

**Código do Lira**

# Quero alterar o título de um post



**API do  
Instagram**

Envia Informações



**PUT JSON**

Devolve uma resposta



**JSON**

**Código do Lira**

# Você também vai ouvir CRUD

CRUD são as ações que você pode fazer em um Banco de Dados:

Create -> POST -> Para criar uma informação

Read -> GET -> Para pegar uma informação

Update -> PUT (PATCH) -> Para atualizar uma informação

Delete -> DELETE -> Para deletar uma informação





# Como você faz essas requisições no seu código?

```
import requests
```

```
requests.get(link)
```

```
requests.post(link)
```

```
...
```



# Ex: Instagram - Usuários

```
import requests  
requests.get("api.instagram.com/usuarios")
```

Resposta: um JSON com todos os usuários do instagram

Tipo:

```
{"ID254": "jpdelira", "ID357": "alonpinheiro", "ID222":  
"sjuliacampos", ...}
```



## Ex: Instagram – 1 usuário

```
import requests  
requests.get("api.instagram.com/usuarios/jpdelira")
```

Resposta: um JSON com as informações do usuário

jpdelira. Tipo:

```
{"ID254": {"nome": "Lira", "idade": 27, "Amigos": ["ID255",  
"ID256", "ID302"], "foto_perfil": "lirabrabo.png"}}
```

## Ex: Instagram - Postar

```
import requests  
requests.post("api.instagram.com/usuarios/jpdelira/post",  
data={"foto": "lirasurfista.png", "titulo": "Surfo mt",  
"descrição": "para aulas, favor inbox"})
```

Resposta: um JSON dizendo se deu tudo certo com o post ou não

## Ex: Instagram – Pegar info de um post

```
import requests  
requests.get("api.instagram.com/usuarios/jpdelira/post/5")
```

Resposta: um JSON com as informações do post. Tipo:

```
{"foto": "lirasurfista.png", "titulo": "Surfo mt", "descrição":  
"para aulas, favor inbox"}
```

## Ex: Instagram – Atualizar um post

```
import requests  
requests.patch("api.instagram.com/usuarios/jpdelira/post  
/5", data={"titulo": "Treinando para as Olimpíadas"})
```

Resposta: um JSON dizendo se deu certo ou não

# Mas por que uma API?

Segurança

Diz o que pode e não pode ser feito

Facilita para outras aplicações – Aplicativo, Site e Código do Lira

Você pode fazer o que quiser no Instagram, desde que a API continue funcionando



# Bora programar

Consumindo uma API (jogo rápido)

- Biblioteca requests

Criar uma API

- Flask

