

# Web Scraping

## Autenticação



# Exemplo 03: SPTrans

## Documentação da API OlhoVivo

```
u_sptrans <- "http://api.olhovivo.sptrans.com.br/v2.1"  
endpoint <- "/Posicao"  
u_sptrans_busca <- paste0(u_sptrans, endpoint)  
(r <- httr::GET(u_sptrans_busca))
```

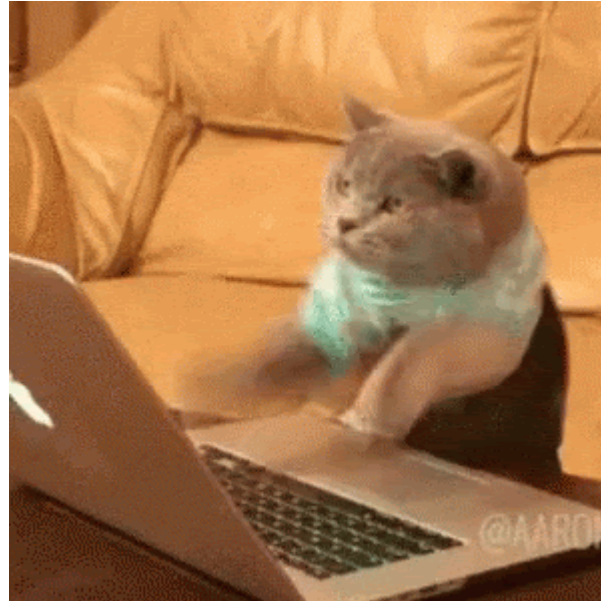
```
Response [http://api.olhovivo.sptrans.com.br/v2.1/Posicao]  
Date: 2020-11-11 17:39  
Status: 401  
Content-Type: application/json; charset=utf-8  
Size: 61 B
```

```
httr::content(r)
```

```
#> $Message
```

```
#> [1] "Authorization has been denied for this request."
```

# Vamos ao R!



# GET e POST

- **GET** utiliza parâmetros na URL
  - Utilizar parâmetro query=
- **POST** utiliza parâmetros no corpo da requisição
  - Utilizar o parâmetro body=
  - Pode utilizar parâmetros na URL, mas não é comum
- A principal diferença entre os dois tipos é a **segurança**

# Exemplo 04: Twitter

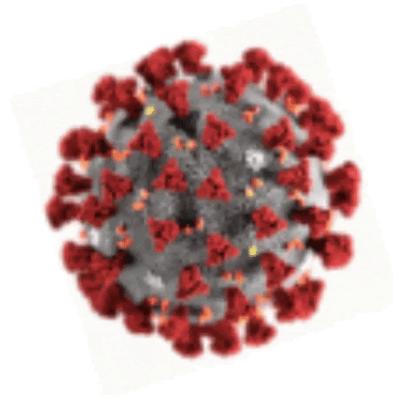
- Utiliza OAuth2.0
- É possível autenticar sem fazer uma conta de desenvolvedor
- Usamos o pacote {rtweet}
- Outros pacotes podem ter uma autenticação mais complicada
- Exemplos de pacotes que utilizam OAuth2.0
  - {googlesheets4}
  - {rdrop2}
- O {httr} possui funcionalidades para conectar com serviços OAuth2.0
- Um exemplo de implementação é o pacote {auth0}

# Vamos ao R!



# Exemplo 05: Covid-19

- Nem sempre é necessário aplicar web scraping...
- No caso do [site do Ministério da Saúde](#), só é necessário se nosso interesse for atualizar a base periodicamente.



# Vamos ao R!

