

# FULL STACK WEB DEVELOPER

FERNANDO LIRA

***FLAG***

# JAVASCRIPT

# APRESENTAÇÃO – FERNANDO LIRA

3



it.fernandolira@gmail.com



<https://www.linkedin.com/in/fernandolira74/>



+351 93 317 99 21



@fernandolira74



**Começa agora e... termina mais tarde!**

```
console.log(1);  
console.log(2);  
  
//O setTimeout é async!  
setTimeout(() => {  
    console.log(3);  
}, 2000);  
  
console.log(4);  
console.log(5);
```



## <https://restcountries.com/> - API

<https://restcountries.com/v3.1/all> - API Endpoint

► GET <https://restcountries.com/v3.1/name/portugal>

Estado **200 OK** ⓘ  
Versão HTTP/1.1  
Transferido 1,43 KB (tamanho 2,81 KB)

### ▼ Cabeçalhos de resposta (222 B)

ⓘ Connection: Keep-Alive  
 ⓘ content-encoding: gzip  
 ⓘ content-length: 1243  
 ⓘ Content-Type: application/json  
 ⓘ Date: Sun, 09 Jan 2022 12:20:43 GMT  
 ⓘ Keep-Alive: timeout=5, max=100  
 ⓘ Server: Apache/2.4.38 (Debian)

### ▼ Cabeçalhos do pedido (512 B)

ⓘ Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,\*/\*;q=0.8  
 ⓘ Accept-Encoding: gzip, deflate, br  
 ⓘ Accept-Language: pt-PT,pt;q=0.8,en;q=0.5,en-US;q=0.3  
 ⓘ Cache-Control: max-age=0  
 ⓘ Connection: keep-alive  
 ⓘ Host: restcountries.com  
 ⓘ Sec-Fetch-Dest: document  
 ⓘ Sec-Fetch-Mode: navigate  
 ⓘ Sec-Fetch-Site: none  
 ⓘ Sec-Fetch-User: ?1  
 ⓘ Upgrade-Insecure-Requests: 1  
 ⓘ User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:95.0) Gecko/20100101 Firefox/95.0

<https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Status> - Status Codes

<https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Methods> - HTTP Request

## HTTP Response - JSON

```
{
  "name": {
    "common": "Portugal",
    "official": "Portuguese Republic",
    "nativeName": {
      "por": {
        "official": "República portuguesa",
        "common": "Portugal"
      }
    },
    "tld": ".pt",
    "cca2": "PT",
    "ccn3": "620",
    "cca3": "PRT",
    "cioc": "POR",
    "independent": true,
    "status": "officially-assigned",
    "unMember": true,
    "currencies": {
      "EUR": {
        "name": "Euro",
        "symbol": "€"
      }
    },
    "idd": {
      "root": "+3",
      "suffixes": ["51"]
    },
    "capital": "Lisbon",
    "altSpellings": ["PT", "Portuguesa", "Portuguese Republic", "República Portuguesa"],
    "region": "Europe",
    "subregion": "Southern Europe",
    "languages": {
      "por": "Portuguese"
    },
    "translations": {
      "ara": {
        "official": "الجمهورية البرتغالية",
        "common": "البرتغال"
      },
      "ces": {
        "official": "Portugalská republika",
        "common": "Portugalsko"
      },
      "cym": {
        "official": "Portuguese Republic",
        "common": "Portugal"
      },
      "deu": {
        "official": "Portugiesische Republik",
        "common": "Portugal"
      },
      "est": {
        "official": "Portugali Vabariik",
        "common": "Portugal"
      },
      "fin": {
        "official": "Portugalin tasavalta",
        "common": "Portugali"
      },
      "fra": {
        "official": "République portugaise",
        "common": "Portugal"
      },
      "hrv": {
        "official": "Portugalska Republika",
        "common": "Portugal"
      },
      "hun": {
        "official": "Portugál Köztársaság",
        "common": "Portugália"
      },
      "ita": {
        "official": "Repubblica portoghese",
        "common": "Portogallo"
      },
      "jpn": {
        "official": "ポルトガル共和国",
        "common": "ポルトガル"
      },
      "kor": {
        "official": "포르투갈 공화국",
        "common": "포르투갈"
      },
      "nld": {
        "official": "Portugese Republiek",
        "common": "Portugal"
      },
      "per": {
        "official": "جمهوری پرتغال",
        "common": "پرتغال"
      },
      "pol": {
        "official": "Republika Portugalska",
        "common": "Portugalia"
      },
      "por": {
        "official": "República portuguesa",
        "common": "Portugal"
      },
      "rus": {
        "official": "Португальская Республика",
        "common": "Португалия"
      },
      "slk": {
        "official": "Portugalská republika",
        "common": "Portugalsko"
      },
      "spa": {
        "official": "República Portuguesa",
        "common": "Portugal"
      },
      "swe": {
        "official": "Republiken Portugal",
        "common": "Portugal"
      },
      "urd": {
        "official": "جمہوریہ پرتگال",
        "common": "پرتگال"
      },
      "zho": {
        "official": "葡萄牙共和国",
        "common": "葡萄牙"
      }
    },
    "latlng": [39.5, -8.0],
    "landlocked": false,
    "borders": ["ESP"],
    "area": 92090.0,
    "demonyms": {
      "eng": {
        "f": "Portuguese",
        "m": "Portuguese"
      },
      "fra": {
        "f": "Portugaise",
        "m": "Portugais"
      }
    },
    "flag": "\uD83C\uDDF5\uD83C\uDDF9",
    "maps": {
      "googleMaps": "https://goo.gl/maps/BaTBSyc4GWMmbAKB8",
      "openStreetMaps": "https://www.openstreetmap.org/relation/295480"
    },
    "population": 10305564,
    "gini": {
      "2018": 33.5
    },
    "fifa": "POR",
    "car": {
      "signs": ["P"],
      "side": "right"
    },
    "timezones": ["UTC-01:00", "UTC"],
    "continents": ["Europe"],
    "flags": {
      "png": "https://flagcdn.com/w320/pt.png",
      "svg": "https://flagcdn.com/pt.svg",
      "coatOfArms": {
        "png": "https://mainfacts.com/media/images/coats_of_arms/pt.png",
        "svg": "https://mainfacts.com/media/images/coats_of_arms/pt.svg"
      },
      "tartOfWeek": "monday",
      "capitalInfo": {
        "latlng": [38.72, -9.13]
      },
      "postalCode": {
        "format": "####-####",
        "regex": "^(\\d{7})$"
      }
    }
  }
}
```

## Promise

```
const paises = async () => { // basta a palavra async para
  tornar a função assíncrona!
};

const listaPaises = paises();

console.log(listaPaises); // Um função assíncrona devolve sempre
uma Promise!
```



## Promise

```
const paises = async () => { // basta a palavra async para tornar a função
  assincrona!

  const response = await fetch('https://restcountries.com/v3.1/all');
  //O fetch devolve uma Promise, então usamos a palavra await para que o javascript
  atribua
  // o valor a response apenas quando a Promise estive concluída!
  // Não esquecer que estamos dentro da função assíncrona!!!
  // Também podemos usar o fetch com ficheiros JSON locais
  //Por exemplo fetch('autores.json')
  console.log(response);
};

const listaPaises = paises();

console.log(listaPaises); // Um função assincrona devolve sempre uma Promise!
```

## Promise

```
const paises = async () => {  
  const response = await  
  fetch('https://restcountries.com/v3.1/all');  
  const data = await response.json(); //retorna uma promise!!  
  console.log(data);  
};  
  
paises();
```

## Promise

```
const paises = async () => {  
    const response = await  
    fetch('https://restcountries.com/v3.1/all');  
    const data = await response.json(); //retorna uma promise!!  
  
    return data;  
};  
  
paises()  
    .then( data => { console.log('Promise concluída',data)});
```

## Promise

```
console.log(1);  
console.log(2);  
  
países()  
  .then( data => { console.log('Promise concluída',data)});  
  
console.log(3);  
console.log(4);
```

## Catch errors

```
const paises = async () => {  
    const response = await  
    fetch('https://restcountries.com/v3.1/all');  
    const data = await response.json();  
    // Quando surge algum erro nas Promise, a função é rejeitada  
    e devolve um erro  
    // Por exemplo, com o json no formato inválido  
  
    return data;  
};  
  
paises()  
    .then( data => { console.log('Promise concluída',data)})  
    .catch( err => { console.log('Promise  
rejeitada',err.message)}); //Promise rejeitada
```

## Catch errors

```
const paises = async () => {  
  
    const response = await fetch('https://restcountries.com/v3.1/allxxx');  
    //A Promise anterior não vem rejeitada mas pode estar incorreta, por exemplo  
    não existir!  
    if (response.status !== 200)  
        throw new Error('Não é possível ler os dados');  
  
    const data = await response.json();  
  
    return data;  
};  
  
paises()  
    .then( data => { console.log('Promise concluída',data)})  
    .catch( err => { console.log('Promise rejeitada',err.message)});
```

GET https://restcountries.com/v3.1/allxxx

Estado	404 Not Found ?
Versão	HTTP/1.1
Transferido	291 B (tamanho 88 B)

▼ Cabeçalhos de resposta (203 B)

?	Connection: Keep-Alive
?	content-length: 88
?	Content-Type: application/json
?	Date: Sun, 09 Jan 2022 13:11:54 GMT
?	Keep-Alive: timeout=5, max=100
?	Server: Apache/2.4.38 (Debian)

# EXERCÍCIOS