

# WEBの仕組み

g's academy

ふくしま はやと

# サーバーとクライアント

クライアント

① <https://google.com>にアクセス  
= **リクエスト**

サーバー

ページ  
を見せてください

了解です。  
用意してあるやつ  
そっちに送ります

③表示される

- HTML
- CSS
- JavaScript

② **レスポンス**

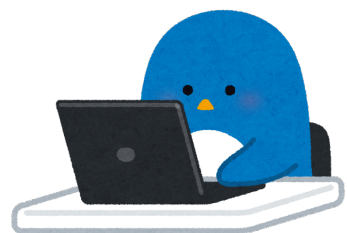
- HTML
- CSS
- JavaScript

用意してある

- HTML
- CSS
- JavaScript

# サーバーとクライアント

サーバー



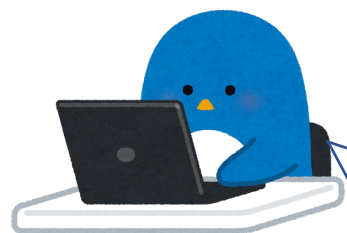
<https://gsacademy.jp/reading/2019/11/000359/>

サーバー:gsacademy.jp  
└reading  
└2019  
└11  
└000359  
└(index.html)



# サーバーとクライアント

*gsacademy.jp*サーバー



httpsという  
通信方法で  
通信しましょう！

<https://gsacademy.jp>



httpsという  
通信方法ね！  
ok!!!

The screenshot shows the Chrome DevTools Network tab. A list of requests is on the left, with '000359/' selected. The right pane shows the 'General' tab for this request. The request is a GET to 'https://gsacademy.jp/reading/2019/11/000359/' with a status of 200. The response headers are visible, showing a 200 OK status and various headers like 'cache-control: no-store, no-cache, must-revalidate' and 'content-type: text/html; charset=UTF-8'.

# サーバーとクライアント

クライアント

① <https://google.com>にアクセス  
= **リクエスト**

サーバー

ページ  
を見せてください

了解です。  
用意してあるやつ  
そっちに送ります

既存ファイル

=中身が決まっている

③表示される

- HTML
- CSS
- JavaScript

② **レスポンス**

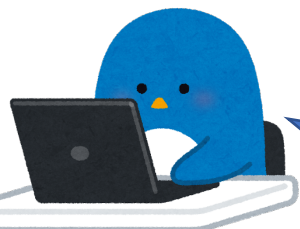
- HTML
- CSS
- JavaScript

用意してある

- HTML
- CSS
- JavaScript

# サーバーとクライアント

クライアント



こんにちは。  
ふくしまです。

③表示される

こんにちは。  
ゲストさん。

② レスポンス

- HTML
- CSS
- JavaScript

中身が決まっている  
場合

こんにちは。  
ゲストさん。



# サーバーとクライアント

クライアント

お名前: ふくしま  
出身: 埼玉県  
送信



① リクエスト

サーバー

お！この人は、  
埼玉県出身の  
ふくしまさんだ。



② レスポンス

- HTML
- CSS
- JavaScript

レスポンスする  
ファイルの「名前」の部分  
と「出身地」の部分を書き  
換えてレスポンスしよう



# サーバーとクライアント

クライアント

お名前: ふくしま  
出身: 埼玉県  
送信



① リクエスト

この処理をするのが、サーバー言語 = PHP  
(他にもPython, Ruby等。)  
PHPの授業はこの処理を書いていきます！！

お！この人は、  
埼玉県出身の  
ふくしまさんだ。



② レスポンス

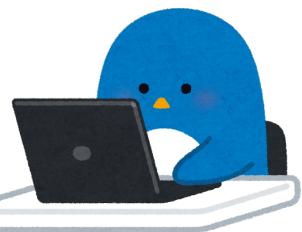
- HTML
- CSS
- JavaScript

レスポンスする  
ファイルの「名前」の部分  
と「出身地」の部分を書き  
換えてレスポンスしよう

# サーバーとクライアント

クライアント

サーバー



誰が来ても変わらない内容をレスポンス  
= 静的(*static*)なWEBページ

レスポンスする内容を変えるサイト  
= 動的(*dynamic*)なWEBページ

# サーバーとクライアント

クライアント

サーバー

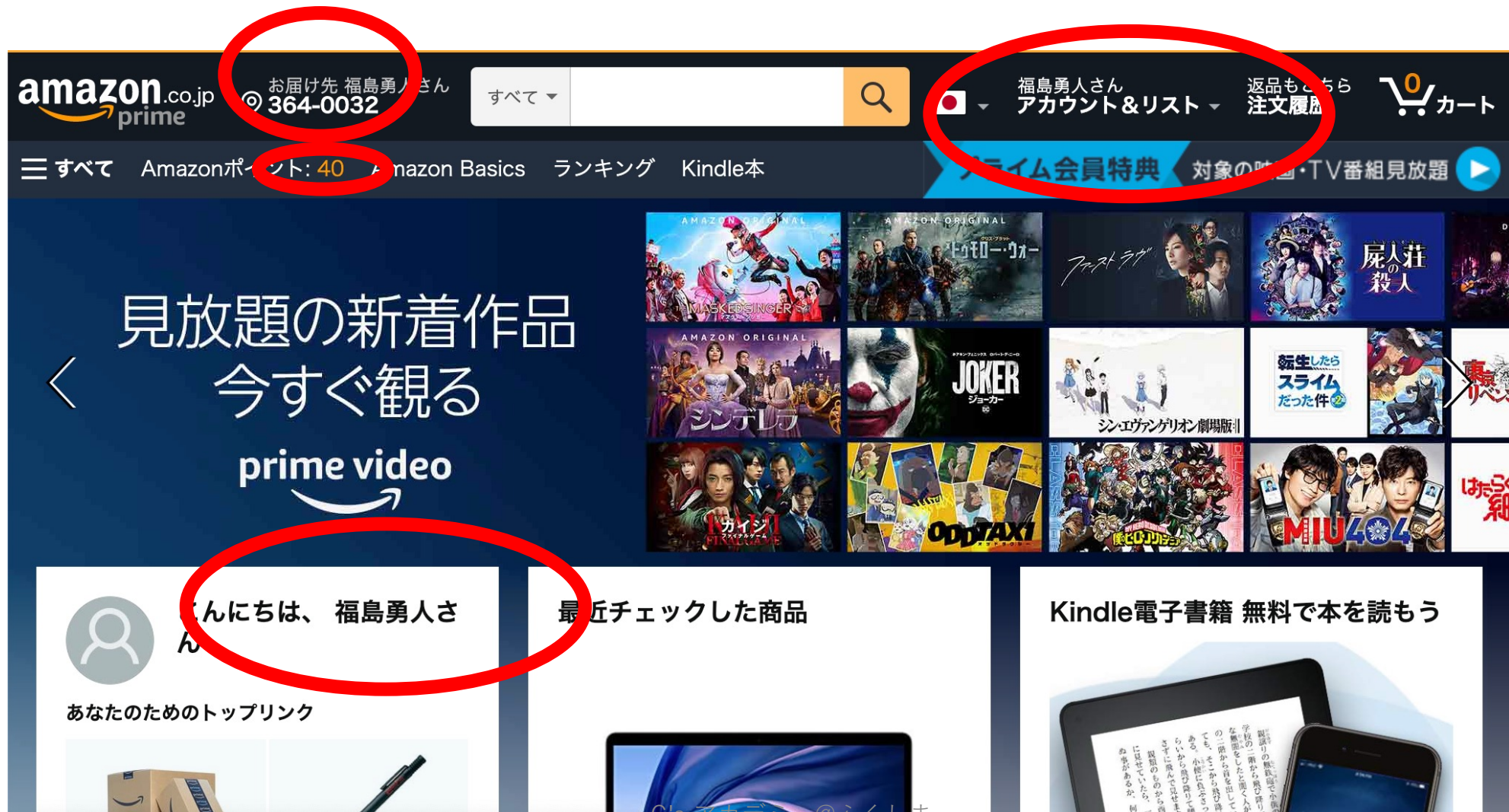


誰が来ても変わらない内容をレスポンス  
= 静的(*static*)なWEBページ

レスポンスする内容を変えるサイト  
= 動的(*dynamic*)なWEBページ

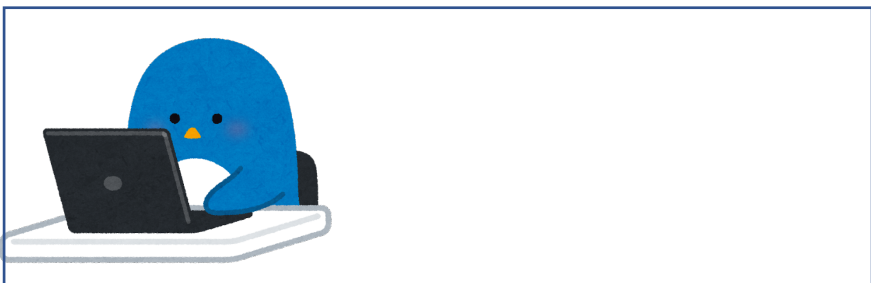
PHPは動的なWEBページを  
作成することができる。

# 動的WEBサイトの例



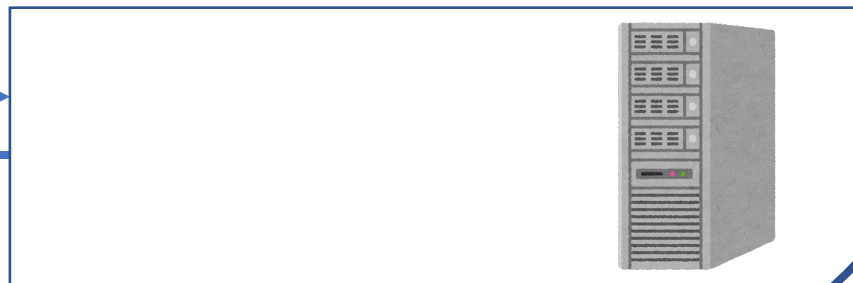
# サーバーとクライアント

クライアント

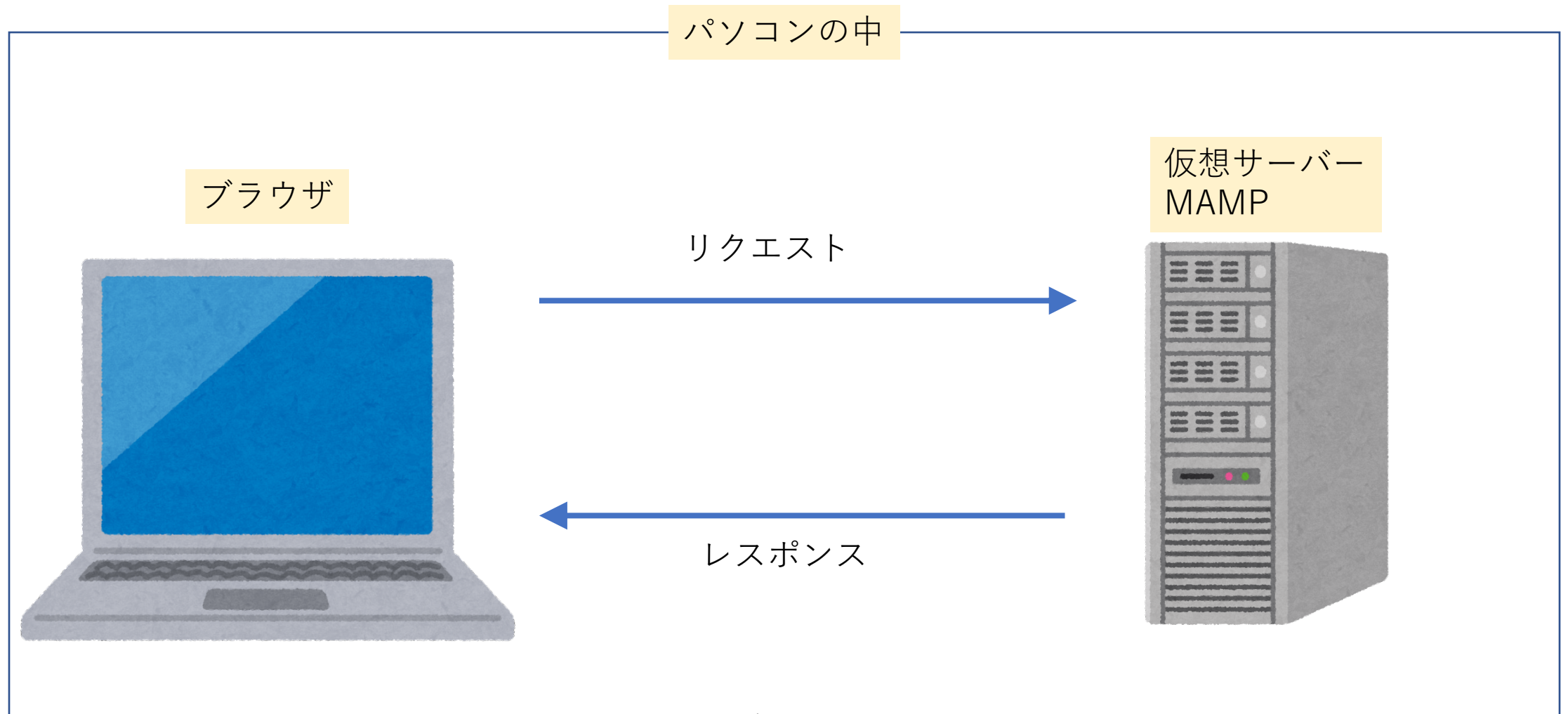


サーバーを用意する必要があります。

サーバー



# サーバーとクライアント



# サーバーとクライアント

サーバー



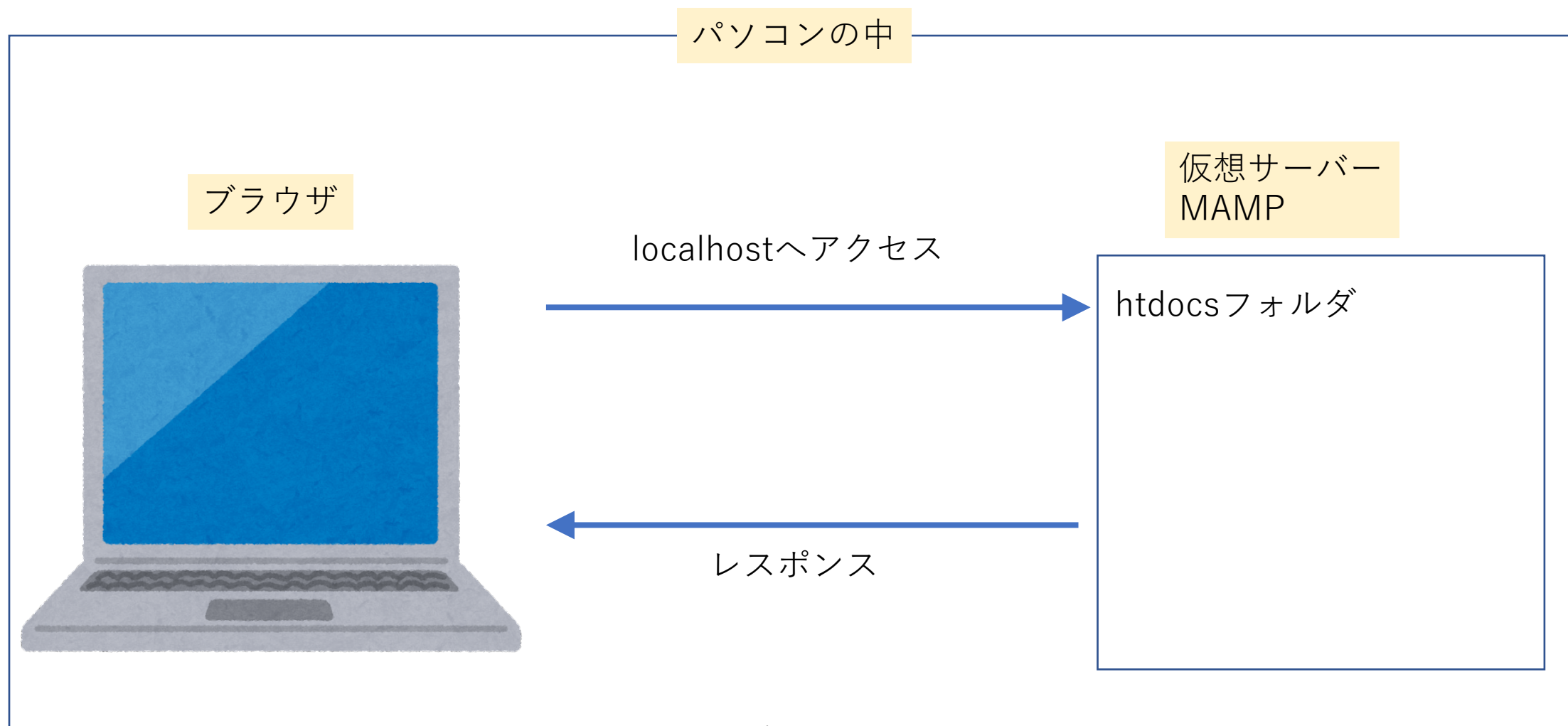
<https://gsacademy.jp/reading/2019/11/000359/>

サーバー:gsacademy.jp  
└reading  
└2019  
└11  
└000359  
└(index.html)





# サーバーとクライアント





# (参考) APIのイメージ

