

リクエスト

- XMLHTTP
- FETCH
- AXIOS



AJAX

- ASYNCHRONOUS
- JAVASCRIPT
- AND
- XML

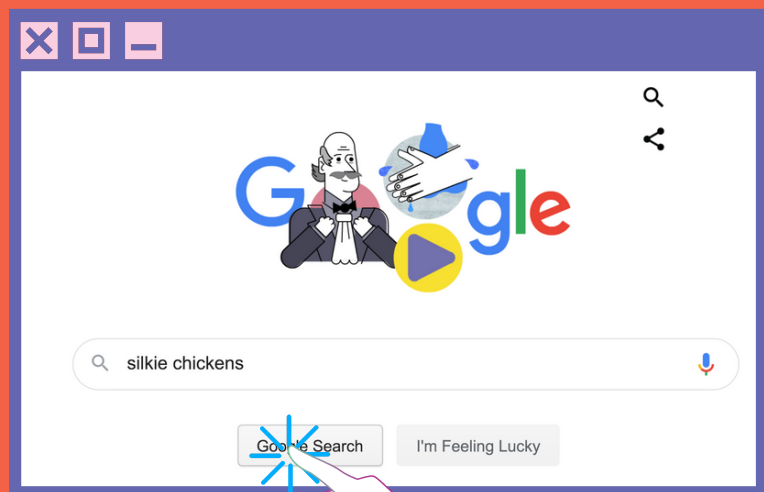


HTTP リクエスト

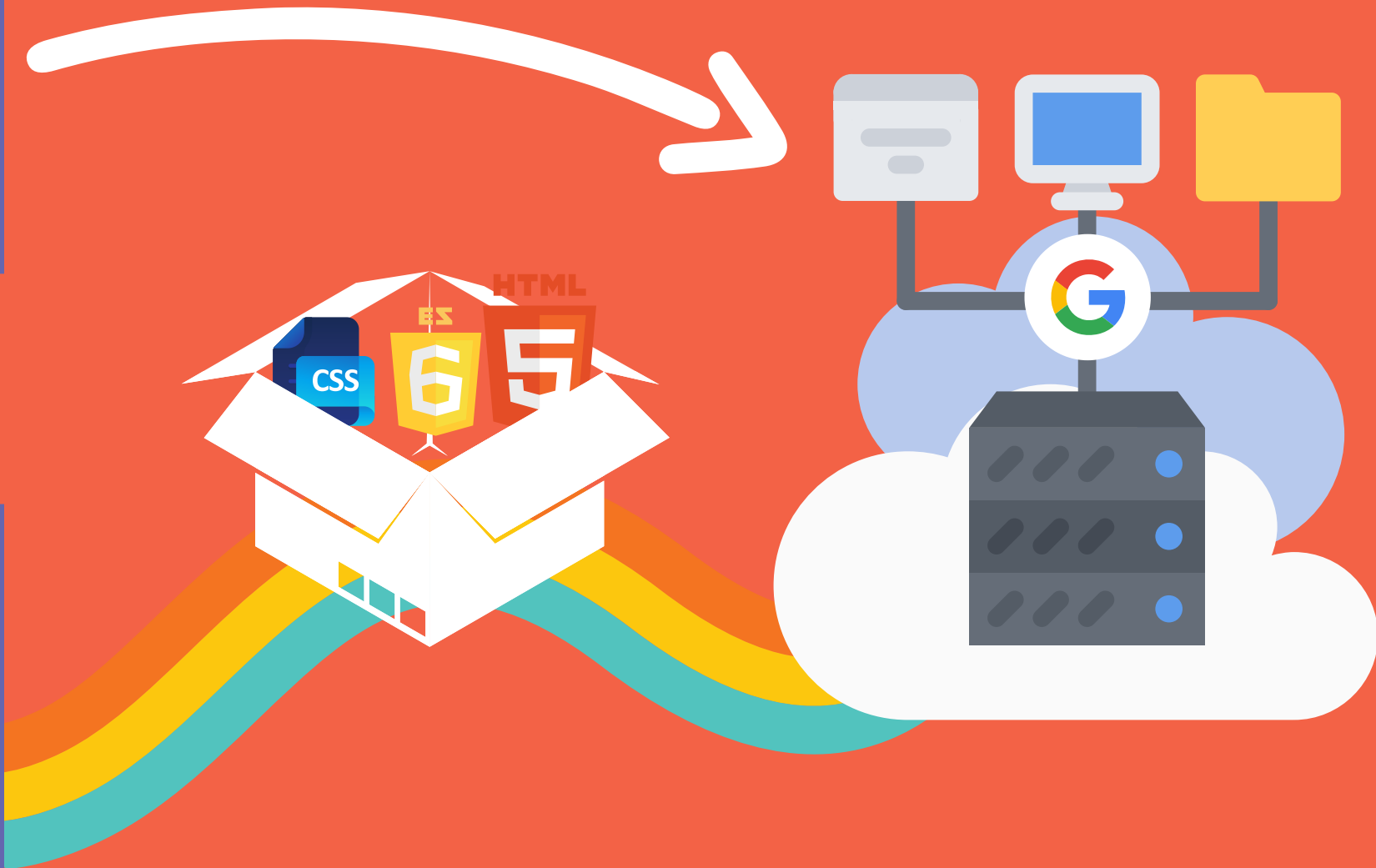
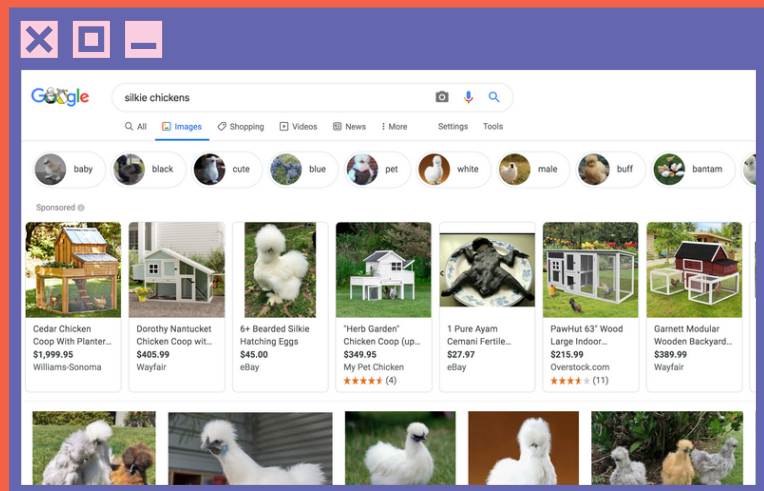


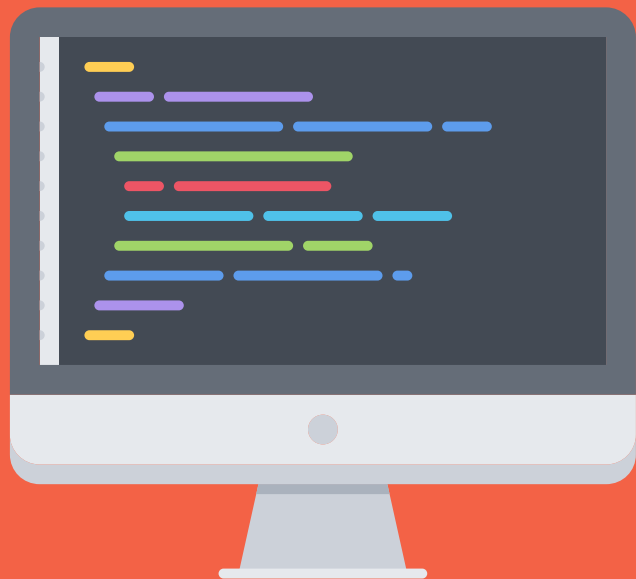
- ウェブでのコミュニケーションの基盤
- "Hyper Text Transfer Protocol"
- リクエスト → ○○の情報がほしいです
- レスポンス → はいよ！





↓これください！
GOOGLE.COM/SEARCH?Q=CHICKENS





↓これください！

[API.CRYPTONATOR.COM/API/TICKER/BTC-JPY](https://api.cryptonator.com/api/ticker/btc-jpy)



価格は 3582613.01

AJAX

- ASYNCHRONOUS
- JAVASCRIPT
- AND
- XML



AJAJ

- ASYNCHRONOUS
- JAVASCRIPT
- AND
- JSON



JSON

- Java
- Script
- Object
- Notation

```
{  
  "squadName": "Super hero squad",  
  "homeTown": "Metro City",  
  "formed": 2016,  
  "secretBase": "Super tower",  
  "active": true,  
  "members": [  
    "Molecule Man",  
    "Madame Uppercut",  
    "Eternal Flame"  
  ]  
}
```


POST MAN



HOPPP

SCOTCH



HTTPS://WWW.AMAZON.COM?
K=BUCATINI&CRID=3IDTUFDWS0DA5&
SPREFIX=BUCAT%2CAPS%2C317&REF=N
B_SB_NOSS_2


?sort=desc&color=blue

XMLHttpRequest

- 昔はajaxと言えばXMLHttpRequest
- Promiseはサポートされていないのでコールバックが多くなりがち
- 使い方がちょっと難しい



XMLHttpRequest



```
const myReq = new XMLHttpRequest();

myReq.onload = function() {
  const data = JSON.parse(this.responseText);
  console.log(data);
};
myReq.onerror = function(err) {
  console.log('ERROR!', err);
};
myReq.open('get', 'https://icanhazdadjoke.com/', true);
myReq.setRequestHeader('Accept', ' application/json');
myReq.send();
```

FETCH



Fetch API

- JavaScriptでリクエストを投げる
モダンなやり方
- Promiseをサポートしている
- Internet Explorerでは使えない☹

FETCH

```
fetch('https://icanhazdadjoke.com/23/2', {  
  headers : { Accept: 'application/json' }  
})  
  .then((res) => {  
    if (res.status !== 200) {  
      console.log('Problem!', res.status);  
      return;  
    }  
    res.json().then((data) => {  
      console.log(data);  
    });  
  })  
  .catch(function(err) {  
    console.log('Fetch Error', err);  
  });
```

AXIOS

HTTPリクエスト用の
外部ライブラリ

