

1. Vorbereitung

1. Ein homogenes Array kann im Gegensatz zum heterogenen Array nur einen Datentyp speichern
2. Die Parameter aus der URL werden in das Array query gespeichert
3. Für jeden key im Interface AssocStringString wird in der Konsole der Key und das dazugehörige Query ausgegeben
4. Dass jeder Origin (= Domain + Port) auf die Inhalte der Seite zugreifen darf

2. Lokale Implementation

2.

```
for (key in query) {  
    _response.write(key + ":" + query[key])  
}
```

4. Ajax

2.

setupColorDivs() -> färbt die divs und fügt ihnen jeweils einen EventListener hinzu

handleClickOnDiv() -> wird bei Klick auf eines der divs aufgerufen, gibt die Farbe des geklickten divs in der Konsole aus und gibt die Farbe des divs an die Funktion sendRequest weiter

sendRequest -> Sendet den Request an den Server und fügt dem Request einen EventListener hinzu, welcher in der Zeile danach getriggert wird und die Funktion handleStateChange aufruft

handleStateChange -> Sobald der Request verarbeitet wurde, wird in der Konsole ausgegeben, dass der Vorgang abgeschlossen ist, der Datentyp String ist und die Anfrage erfolgreich war
3.

Die Klasse XMLHttpRequest dient dazu, um Daten von einer URL zu erhalten

readyState = Status der Anfrage (0-4)

responseType = Kann gesetzt werden, um den Datentyp der Antwort zu ändern. Default ist String

status = Status der Antwort auf die Anfrage (200 = erfolgreiche Anfrage)

statusText = Wie status, jedoch als ganze Antwortnachricht (z.B. OK)
4.

Asynchrone Datenübertragung zwischen Browser und Server. Ermöglicht zB HTTP-Anfragen durchzuführen, während eine HTML-Seite angezeigt wird.