

PHP + AJAX == ???

TRY / CATCH

För enklare error-hantering, bara visa meddelandet

```
$pdo->setAttribute( PDO::ATTR_ERRMODE, PDO::ERRMODE_
```

Samma som vi gör med **options**-argumentet när vi skapar vår
PDO

För mer errorhantering gällande **PDO**

```
try{  
    $pdo = new PDO( /**/ );  
}catch(PDOException $error){  
    echo $error->getMessage();  
}
```

AJAX

Dynamisk uppdatering av innehåll

Det vi egentligen gör:

Hämtar en annan sida och appendar dess innehåll

fetch

```
fetch( 'https://localhost/myApi' )  
  .then( data => data.json() )  
  .then( json => console.log( json ) );
```

Sidan är kanske egentligen `localhost:80/myApi.json`

Servern har sagt till att innehållet ska levereras som JSON

`data.json()` konverterar innehållet till läslig JSON

I PHP har vi arrayer som måste konverteras

```
json_encode($phpArray); //JSON.stringify() kinda  
json_decode($jsonObject); //JSON.parse() kinda
```

JSON är bara en sträng och kan därför echo ut.

echo är bara för strängar

Eftersom vi `echo` ut en sträng är det sidans `body`

HEADERS - BODY

```
//hello.php
```

```
<?php
```

```
echo 'Hello!'; //body of 'hello.php'
```

Vi kan hämta sidan med `ajax`

`.text()` om det är något annat än JSON

```
fetch( 'hello.php' )  
  .then(data => data.text())  
  .then(text => console.log(text));
```

POST

Att skicka formulär fungerar typ på samma sätt
Men oftast echoar vi inte ut något när vi postar.

```
fetch( 'post.php' ,  
{  
  method: 'POST' ,  
  body:  ?????  
} )
```

Vad skickar vi med och i vilket format?

FormData

```
//jQuery  
let formData = $( '#myForm' ).serialize();
```

Plockar alla input och gör till key/value -> "name"="value"

The `.serialize()` method creates a text string in standard URL-encoded notation.

```
application/x-www-form-urlencoded
```

==

```
post.php?key1=value1&key2=value2
```

Man kan sätta `'Content-type'` själv

```
header( 'Content-type: application/json' );  
header( 'Content-type: application/x-www-form-urlencoded'
```

Jag använde tidigare en annan header

```
header( 'Location: /' )
```

Koden ovan redirectar till root

Om du vill göra ett eget API kanske?

Enable Cross Origin Resource Sharing

```
<?php
```

```
header( "Access-Control-Allow-Origin: *" );
```

Tillåter att alla domäner kan hämta data från vår server

Kom ihåg

Strängar och semikolon

Adress, port och databasnamn

```
form.addEventListener('submit', (event) => {  
    event.preventDefault();  
});
```