

Systemutveckling PHP

Föreläsning 02 - Grunderna i PHP

Närvaro

- Om närvaro
- Streaming

Dagens ämnen

- PHP i webbmiljö
- Arrayer
- Formulär
- Sträng-funktioner

Schema v2

	Mån	Tis	Ons	Tors	Fre
08:30 - 09:00		Egna studier		Handledning	Egna studier
09:00 - 09:30					
09:30 - 10:00	Föreläsning				
10:00 - 10:30	Övningar		Föreläsning	Föreläsning	
10:30 - 11:00			Övningar	Övningar	
11:00 - 11:30		Egna studier			Egna studier
11:30 - 12:00	Lunch				
12:00 - 12:30			Lunch	Öppet forum	
12:30 - 13:00	Föreläsning			Lunch	
13:00 - 13:30			Föreläsning		
13:30 - 14:00		Egna studier			Egna studier
14:00 - 14:30	Projekt			Repetition	
14:30 - 15:00			Projekt	Quiz	
15:00 - 15:30					
15:30 - 16:00					

PHP i webbmiljö

- PHP-filer kan innehålla HTML och/eller PHP-kod.

```
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="css/main.css">
<script src="main.js"></script>
</head>
<body>
<?php
    echo date("Y-m-d H:i:s");
?>
</body>
</html>
```

PHP i webbmiljö

- PHP-filer kan innehålla hur många PHP-block som helst.

```
<body>
<h1>Hello world!</h1>
Dagens datum: <?php echo date("Y-m-d H:i:s"); ?><br>
<em>Hello world</em> baklänges blir: <?php echo strrev("Hello world"); ?>
</body>
```

Vad gör denna funktion?

PHP i webbmiljö

- Allt vi gjort hittills kan vi göra i webbmiljö.

```
<h1>Hello world:</h1>
<?php
    for ($i = 1; $i <= 5; $i++) {
        echo "$i<br>";
    }
?>
</body>
```

HELLO world!

1
2
3
4
5

PHP i webbmiljö

- Vad man kan göra i PHP beror på vilka moduler man har slagit på.
- Man slår på/av moduler i php.ini.
- Det finns en speciell funktion i PHP som visar information om installationen.

```
extension=imap.so
extension=gettext.so
extension=pgsql.so
extension=pdo_pgsql.so
```

PHP Version 7.2.1

[illegible]

Övning

- Skapa en sida som skriver ut er PHP-konfiguration.
- Se till att tillägget imagick finns.
 - <https://ourcodeworld.com/articles/read/349/how-to-install-and-enable-the-imagick-extension-in-xampp-for-windows>
 - <https://github.com/jdlx/install-Imagick-on-MAMP/wiki/Imagick-for-MAMP-Pro---installation-guide>
- Använd imagick i valfritt exempel. Ni behöver inte förstå koden.
 - <https://phpimagick.com/Tutorial>

Värden som hör ihop

- Hur ska man göra om man har flera studenter?
- Begränsande!

```
$student = "Kalle";
```

```
$student = "Lisa";
```

```
$student2 = "Jessica";
```

```
$student3 = "Lisa";
```

?

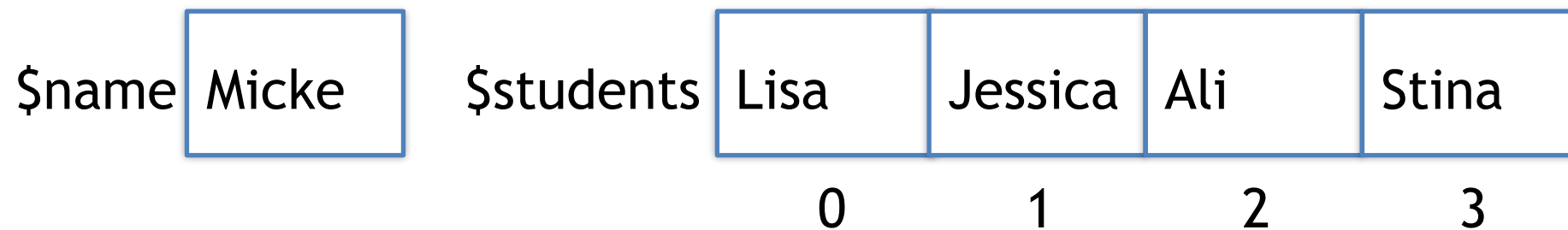
Array

En array är en speciell datatyp som kan innehålla flera värden.

```
$studenter = ["Lisa", "Jessica"];

// Kan stå på flera rader
$studenter = [
    "Lisa",
    "Jessica"
];
```

Array



- En array är en speciell datatyp som kan innehålla flera värden.

```
$name = "Micke";  
$students = [  
    "Lisa",  
    "Jessica",  
    "Ali",  
    "Stina"  
];  
  
echo $name . PHP_EOL;  
echo $students[0];
```

Array

- Det finns en speciell funktion för att skriva ut arrayer.

```
$students = [  
    "Lisa",  
    "Jessica"  
];  
  
print_r($students);
```

```
Array  
(  
    [0] => Lisa  
    [1] => Jessica  
)
```

Array

Man kan även skapa en array med funktionen array.

```
$students = array();  
print_r($students);
```

```
$students = array("kalle");  
print_r($students);
```

```
Array  
(  
)  
Array  
(  
    [0] => kalle  
)
```

Array

- Man kan även få den att returnera värdet.

```
$students = [  
    "Lisa",  
    "Jessica"  
];  
  
$arr = print_r($students, true);  
echo $arr;
```

```
Array  
(  
    [0] => Lisa  
    [1] => Jessica  
)
```

Array - lägga till värde

- Det är enkelt att lägga till nya värden.

```
$students = [  
    "Lisa",  
    "Jessica"  
];  
$students[] = "Kalle";  
  
print_r($students);
```

```
Array  
(  
    [0] => Lisa  
    [1] => Jessica  
    [2] => Kalle  
)
```


Associativ array

- En array kan även använda strängar som nycklar.

```
$students["name"] = "Kalle";  
$students["age"] = 25;  
$students["shoesize"] = 44;  
  
print_r($students);  
  
echo $students["name"];
```

```
Array  
(  
    [name] => Kalle  
    [age] => 25  
    [shoesize] => 44  
)  
Kalle
```

Associativ array

- Man kan även sätta värden direkt som nyckel => värde.

```
$students = [  
    "name" => "Kalle",  
    "age" => 25,  
    "shoesize" => 44  
];  
  
print_r($students);
```

```
Array  
(  
    [name] => Kalle  
    [age] => 25  
    [shoesize] => 44  
)
```

Multidimensionell array

- En array kan ha en annan array som värde.

```
$course = [  
  "teacher" => "Micke",  
  "students" => [  
    0 => [  
      "name" => "Kalle",  
      "age" => 25  
    ]  
  ]  
];  
  
$course["students"][1] = [  
  "name" => "Stina",  
  "age" => 28  
];  
  
print_r($course);
```

```
Array  
(  
    [teacher] => Micke  
    [students] => Array  
        (  
            [0] => Array  
                (  
                    [name] => Kalle  
                    [age] => 25  
                )  
            [1] => Array  
                (  
                    [name] => Stina  
                    [age] => 28  
                )  
        )  
    )
```

Array - ändra värde

- Ange nyckeln / indexet för den nyckel du vill ändra.

```
$studerter = [  
    "Lisa",  
    "Jessica"  
];  
$studerter[0] = "Kalle";  
  
print_r($studerter);  
  
$course = [  
    "teacher" => "Micke",  
    "students" => []  
];  
  
$course["teacher"] = "Stina";  
  
print_r($course);
```

```
Array  
(  
    [teacher] => Stina  
    [students] => Array  
        (  
        )  
    )  
)
```

compact

- compact() looks for a variable with that name in the current symbol table and adds it to the output array such that the variable name becomes the key and the contents of the variable become the value for that key

```
$city = "San Francisco";  
$state = "CA";  
$event = "SIGGRAPH";
```

```
$location_vars = array("city", "state");
```

```
$result = compact("event", "nothing_here", $location_vars);  
print_r($result);
```

```
Array  
(  
    [event] => SIGGRAPH  
    [city] => San Francisco  
    [state] => CA  
)
```

implode

- Returns a string containing a string representation of all the array elements in the same order, with the glue string between each element.

```
$array = array('lastname', 'email', 'phone');  
$comma_separated = implode(",", $array);
```

```
echo $comma_separated; // lastname,email,phone
```

```
// Empty string when using an empty array:
```

```
var_dump(implode('hello', array())); // string(0) ""
```

array_values

- Return all the values of an array.

```
$array = array("size" => "XL", "color" => "gold");  
print_r(array_values($array));
```

```
Array  
(  
    [0] => XL  
    [1] => gold  
)
```

Övningar

- <https://www.w3schools.com/php/>
 - Arrays
 - Sorting arrays
 - Arrays multi (under PHP Advanced)
- Om du hinner:
 - <https://www.w3resource.com/php-exercises/php-array-exercises.php>

Objekt

- Ett objekt är ganska likt en array i PHP, åtminstone så långt vi vet om dem än.
- Ett standardobjekt skapas med `stdClass()`. Lägg också märke till nyckelordet *new*.

```
$course = new stdClass();  
  
$course->name = "System development in PHP";  
$course->points = 50;  
$course->approved = true;  
  
print_r($course);
```

```
stdClass Object  
(  
    [name] => System development  
    [points] => 50  
    [approved] => 1  
)
```

Objekt

Ett objekt är ganska likt en array i PHP, åtminstone så långt vi vet om dem än.

```
$course = new stdClass();  
  
$course->name = "System developme  
$course->points = 50;  
$course->approved = true;  
  
print_r($course);
```

Superglobala variabler

- Det finns ett par fördefinierad superglobala variabler som alltid finns tillgängliga.

- \$GLOBALS
- \$_SERVER
- \$_GET
- \$_POST
- \$_FILES
- \$_COOKIE
- \$_SESSION
- \$_REQUEST
- \$_ENV

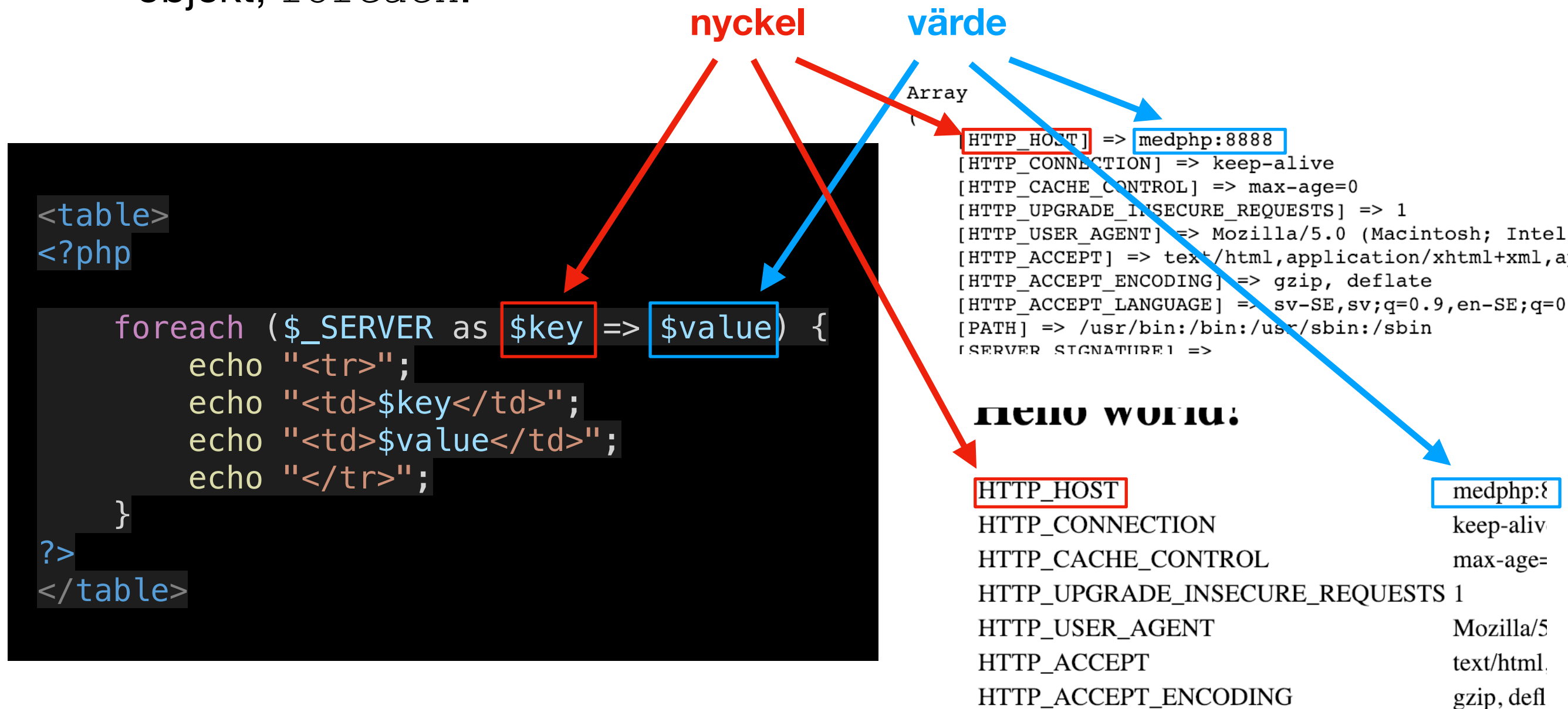
```
<body>
<pre>
<?php print_r ($_SERVER); ?>
</pre>
</body>
```

Array
(

```
[HTTP_HOST] => medphp:8888
[HTTP_CONNECTION] => keep-alive
[HTTP_CACHE_CONTROL] => max-age=0
[HTTP_UPGRADE_INSECURE_REQUESTS] => 1
[HTTP_USER_AGENT] => Mozilla/5.0 (Macintosh; Intel Mac
[HTTP_ACCEPT] => text/html,application/xhtml+xml,appli
[HTTP_ACCEPT_ENCODING] => gzip, deflate
[HTTP_ACCEPT_LANGUAGE] => sv-SE,sv;q=0.9,en-SE;q=0.8,e
[PATH] => /usr/bin:/bin:/usr/sbin:/sbin
[SERVER_SIGNATURE] =>
```

foreach

- Sist pratade vi om `for`-loopar.
- Det finns en speciell `for`-loop man kan använda för arrayer och objekt, `foreach`.



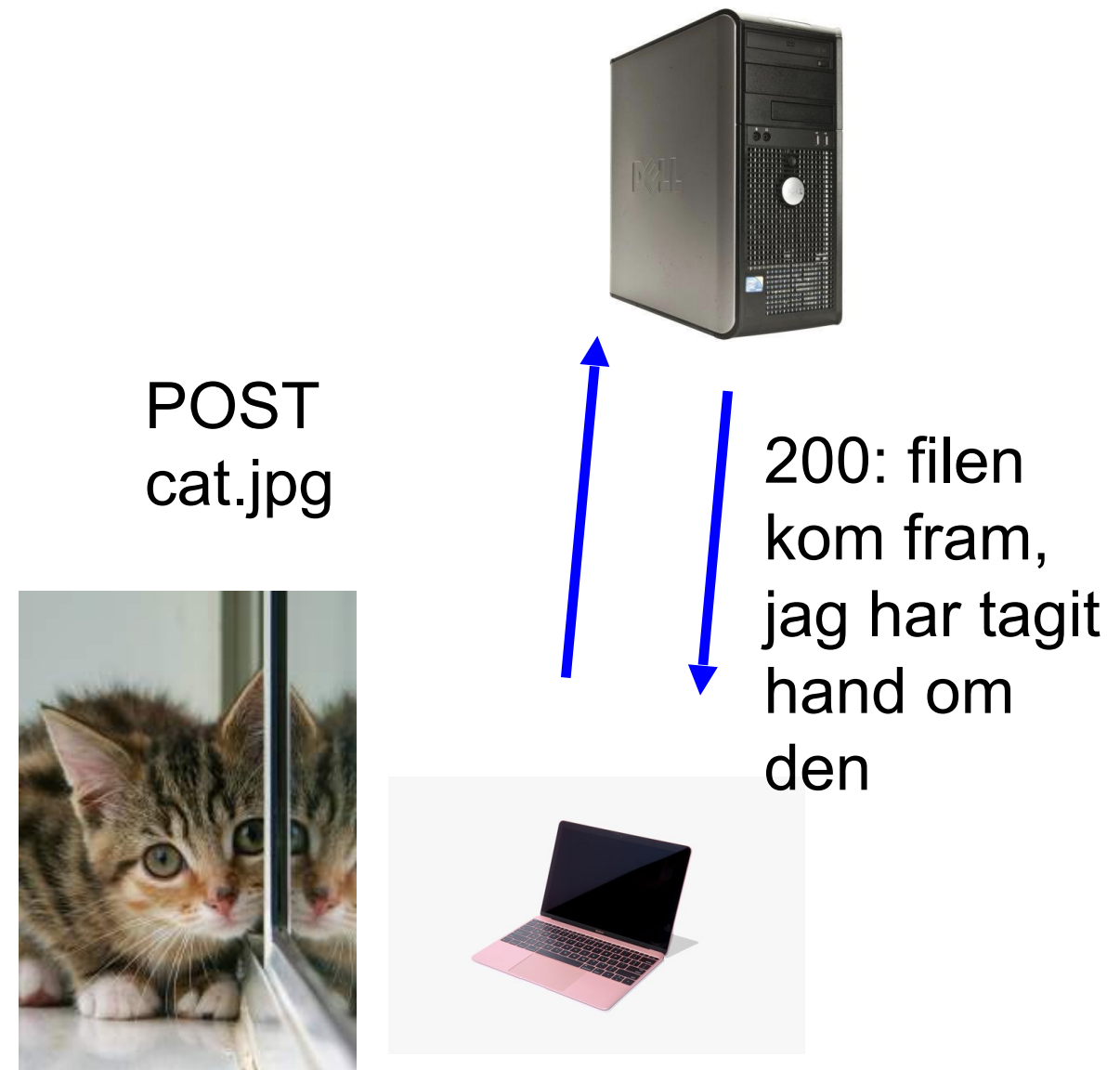
HTML - formulär

- Servern kan skicka information till besökaren (klienten) i form av html, css osv.



HTML - formulär

- Hur kan besökaren (klienten) skicka information till servern?
- Varför skulle vi vilja att besökaren kan skicka information till oss?
- Hur kan vi hantera informationen som besökaren skickar till oss?



HTML - formulär

- Saker vi kan göra med information besökaren skickar:
 - Posta ett inlägg a la facebook
 - Visa en kategori av böcker som besökaren valt
 - Autentisera användare
 - Mer?

HTML - formulär

```
<form method="post" action="some_page.php">
```

```
Namn:<br>
```

```
<input type="text" name="name" placeholder="Skriv  
ditt namn här"><br>
```

```
Ålder:
```

```
<select name="age">  
  <option value="age_group1">0-30</option>  
  <option value="age_group2">30-60</option>  
  <option value="age_group3">60-</option>  
</select>
```

```
</form>
```


HTML - formulär

```
<form method="post" action="some_page.php">
```

```
  Intressen:<br>
```

```
    <input type="checkbox" name="hobbies"
value="fotboll">Fotboll<br>
```

```
    <input type="checkbox" name="hobbies"
value="hockey">Hockey<br>
```

```
  Favorit-färg:
```

```
    <input type="radio" name="color"
value="yellow">Gul<br>
```

```
    <input type="radio" name="color" value="red">Röd<br>
```

```
</form>
```

HTML - formulär

```
<form method="post" action="some_page.php">  
  <input type="number" min="0" max="100" step="2"><br>  
  <input type="email"><br>  
  <input type="color"><br>  
</form>
```

Övning 1

- Skapa en array som innehåller alla superglobaler.
- För varje superglobal, skriv ut alla nycklar och dess värden.

Övning 2

- Skapa en sida med ett formulär med följande komponenter: textruta, textarea, select, radioknapp, checkbox, colorpicker
- Posta sidan till en ny sida och skriv där ut alla nycklar och värden som användaren har skickat.

Övning 3

- Strängfunktionerna i PHP:
<http://php.net/manual/en/ref.strings.php>
- Skapa ett formulär som låter användare använda följande funktioner:
 - htmlentities, explode, crypt, str_word_count, strip_tags, strlen

Övning 4

- Skapa spelet Nim. En hög har från början 21 pinnar. Två spelare turas om att ta 1, 2 eller 3 pinnar från högen. Den som tar den sista pinnen har förlorat.
- Håll reda på hur många pinnar som finns kvar i högen, vad spelarna heter och vems tur det är.

Lunch

- 12:00 - 13:00

MySQL

- Se till att MySQL funkar i era miljöer
- Det finns en ersättare som heter MariaDB, båda funkar.
- Installera en DB-editor typ MySQL Workbench.
<https://www.mysql.com/products/workbench/>
- Koppla upp er med er editor mot databasen och installera databasen *classic models*.
<http://www.mysqltutorial.org/mysql-sample-database.aspx>

MySQL övningar

1. Lista alla kunder i USA.
2. Lista alla kunder i USA som har kontakt på företaget och namnet på den kontakten (tips: salesRepEmployeeNumber).
3. Lista alla kunder i USA oavsett om de har en kontakt på företaget eller inte. (Är det någon skillnad i antalet rader?)
4. Lista alla kunder i USA, deras kontaktpersoner och kontaktpersonernas chefer.
5. Lista alla ordrar.
6. Lista alla ordrar under 2003.
7. Lista alla beställda produkter.
8. Lista alla produkter som beställdes under 2003.
9. Lista hur många av varje produkt som såldes under 2003. Sortera på antalet produkter fallande, därefter på produktnamnet.

Sammanfattning

- PHP i webbmiljö
- Arrayer
- Formulär
- Sträng-funktioner

Utvärdering

- Prata i grupper om 2-3 personer i två minuter.
- Vad har varit bra idag?
- Vad skulle kunna förbättras?

Tack för idag!

Vi ses i morgon!

Mikael Olsson
mikael.olsson@emmio.se
076-174 90 43

