

PHP a MySQL

HTML formuláře a jejich zpracování v PHP

Projekt

CZ.1.07/1.5.00/34.1009

DUM

VY_32_INOVACE_286

Ing. Karel Johanovský

Střední průmyslová škola Jihlava

2013



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Identifikační údaje

Projekt	<i>Inovace výuky prostřednictvím ICT</i>
<i>Číslo projektu</i>	<i>CZ.1.07/1.5.00/34.1009</i>
<i>Číslo DUM</i>	<i>VY_32_INOVACE_286</i>
<i>Autor</i>	<i>Ing. Karel Johanovský</i>
<i>Datum vytvoření</i>	<i>16. března 2014</i>
<i>Tematický celek</i>	<i>Programování a vývoj aplikací - PHP a MySQL</i>
Téma	<i>HTML formuláře a jejich zpracování v PHP</i>
<i>Anotace</i>	<i>Podpora výuky programování dynamických webů</i>
<i>Metodický pokyn</i>	<i>Prezentace s výkladem, časová náročnost 20 minut</i>
<i>Inovace</i>	<i>Podpora vjemu informací u žáka ve fázi expozice . a zejména ve fázi fixace získaných poznatků (dostupný materiál – možnost libovolného počtu opakování)</i>

1 Úvod

- Úvod

2 HTML

- Formulář
- Textový vstup
- Vstup pro heslo
- Checkbox
- Radio - button
- Select
- Submit

3 PHP

- Superglobální proměnné
- Získání dat z formulářů

- Nejčastějším úkolem dynamických webových stránek je zpracovávat nějaká uživatelská data.

- Nejčastějším úkolem dynamických webových stránek je zpracovávat nějaká uživatelská data.
- Tato data je potřeba nějak do aplikace dostat. Základním způsobem jsou HTML formuláře.

- Nejčastějším úkolem dynamických webových stránek je zpracovávat nějaká uživatelská data.
- Tato data je potřeba nějak do aplikace dostat. Základním způsobem jsou HTML formuláře.
- Cílem této prezentace je zopakovat základní prvky HTML formulářů a ukázat jak data z nich zpracovat v PHP.

- Nejčastějším úkolem dynamických webových stránek je zpracovávat nějaká uživatelská data.
- Tato data je potřeba nějak do aplikace dostat. Základním způsobem jsou HTML formuláře.
- Cílem této prezentace je zopakovat základní prvky HTML formulářů a ukázat jak data z nich zpracovat v PHP.
 - textový vstup

- Nejčastějším úkolem dynamických webových stránek je zpracovávat nějaká uživatelská data.
- Tato data je potřeba nějak do aplikace dostat. Základním způsobem jsou HTML formuláře.
- Cílem této prezentace je zopakovat základní prvky HTML formulářů a ukázat jak data z nich zpracovat v PHP.
 - textový vstup
 - vstup pro hesla

- Nejčastějším úkolem dynamických webových stránek je zpracovávat nějaká uživatelská data.
- Tato data je potřeba nějak do aplikace dostat. Základním způsobem jsou HTML formuláře.
- Cílem této prezentace je zopakovat základní prvky HTML formulářů a ukázat jak data z nich zpracovat v PHP.
 - textový vstup
 - vstup pro hesla
 - checkbox

- Nejčastějším úkolem dynamických webových stránek je zpracovávat nějaká uživatelská data.
- Tato data je potřeba nějak do aplikace dostat. Základním způsobem jsou HTML formuláře.
- Cílem této prezentace je zopakovat základní prvky HTML formulářů a ukázat jak data z nich zpracovat v PHP.
 - textový vstup
 - vstup pro hesla
 - checkbox
 - radio

- Nejčastějším úkolem dynamických webových stránek je zpracovávat nějaká uživatelská data.
- Tato data je potřeba nějak do aplikace dostat. Základním způsobem jsou HTML formuláře.
- Cílem této prezentace je zopakovat základní prvky HTML formulářů a ukázat jak data z nich zpracovat v PHP.
 - textový vstup
 - vstup pro hesla
 - checkbox
 - radio
 - select

- Formuláře se používají k odesílání dat na server, který je zpracuje.

- Formuláře se používají k odesílání dat na server, který je zpracuje.
- Na jedné stránce může být více různých formulářů.

Formulář

- Formuláře se používají k odesílání dat na server, který je zpracuje.
- Na jedné stránce může být více různých formulářů.
- Data z formulářů mohou být odeslaná kamkoli.

Formulář

- Formuláře se používají k odesílání dat na server, který je zpracuje.
- Na jedné stránce může být více různých formulářů.
- Data z formulářů mohou být odeslaná kamkoli.
- Každý HTML formulář je uzavřen mezi tagy `<form>`

Formulář

- Formuláře se používají k odesílání dat na server, který je zpracuje.
- Na jedné stránce může být více různých formulářů.
- Data z formulářů mohou být odeslaná kamkoli.
- Každý HTML formulář je uzavřen mezi tagy `<form>`
- Každý formulář má dva důležité atributy:

- Formuláře se používají k odesílání dat na server, který je zpracuje.
- Na jedné stránce může být více různých formulářů.
- Data z formulářů mohou být odeslaná kamkoli.
- Každý HTML formulář je uzavřen mezi tagy `<form>`
- Každý formulář má dva důležité atributy:
 - action: Zde se jako hodnota uvádí URL, kam se pošlou data, obvykle nějaký PHP, nebo podobný script.

- Formuláře se používají k odesílání dat na server, který je zpracuje.
- Na jedné stránce může být více různých formulářů.
- Data z formulářů mohou být odeslaná kamkoli.
- Každý HTML formulář je uzavřen mezi tagy `<form>`
- Každý formulář má dva důležité atributy:
 - action: Zde se jako hodnota uvádí URL, kam se pošlou data, obvykle nějaký PHP, nebo podobný script.
 - method: Zde se uvede metoda, kterou mají být data odeslána.

- Formuláře se používají k odesílání dat na server, který je zpracuje.
- Na jedné stránce může být více různých formulářů.
- Data z formulářů mohou být odeslaná kamkoli.
- Každý HTML formulář je uzavřen mezi tagy `<form>`
- Každý formulář má dva důležité atributy:
 - action: Zde se jako hodnota uvádí URL, kam se pošlou data, obvykle nějaký PHP, nebo podobný script.
 - method: Zde se uvede metoda, kterou mají být data odeslána.

HTML - formulář

```
<form action="" method="">  
    formulářové prvky  
</form>
```

Metody pro odeslání dat

Metody pro odeslání dat

- Metody pro odeslání dat.

Metody pro odeslání dat

- Metody pro odeslání dat.
 - GET: Používanější, ale data jsou součástí URL.
 - `http://adresa/prihlasit.php?user=abc&pass=abc&login=login`

Metody pro odeslání dat

- Metody pro odeslání dat.
 - GET: Používanější, ale data jsou součástí URL.
 - `http://adresa/prihlasit.php?user=abc&pass=abc&login=login`
 - POST: Data součástí dotazu, pro hesla a obsáhlá data.
 - `http://adresa/prihlasit.php`

Textový vstup

Textový vstup

- Pro zadávání textových vstupů uživatele.

Textový vstup

- Pro zadávání textových vstupů uživatele.
- Používá se nepárový tag `<input>`, který je typu: `text`.

Textový vstup

- Pro zadávání textových vstupů uživatelé.
- Používá se nepárový tag `<input>`, který je typu: text.
- Atribut name slouží k identifikaci vstupu, atribut value je hodnota, kterou je pole předvyplněno.

Textový vstup

- Pro zadávání textových vstupů uživatelé.
- Používá se nepárový tag `<input>`, který je typu: text.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou je pole předvyplněno.

Textový vstup

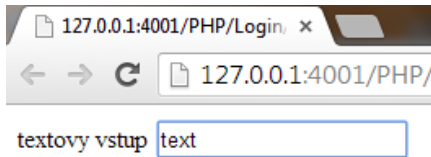
```
<input type='text' name='muj_text' value='text' />
```

Textový vstup

- Pro zadávání textových vstupů uživatele.
- Používá se nepárový tag `<input>`, který je typu: text.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou je pole předvyplněno.

Textový vstup

```
<input type='text' name='muj_text' value='text' />
```



Vstup pro heslo

Vstup pro heslo

- Pro zadávání textových vstupů uživatele, které nebudou čitelné.

Vstup pro heslo

- Pro zadávání textových vstupů uživatele, které nebudou čitelné.
- Používá se nepárový tag `<input>`, který je typu: `password`.

Vstup pro heslo

- Pro zadávání textových vstupů uživatele, které nebudou čitelné.
- Používá se nepárový tag `<input>`, který je typu: `password`.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou je pole předvyplněno.

Vstup pro heslo

- Pro zadávání textových vstupů uživatele, které nebudou čitelné.
- Používá se nepárový tag `<input>`, který je typu: `password`.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou je pole předvyplněno.

Vstup pro heslo

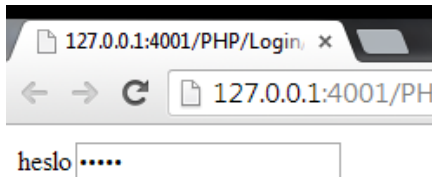
```
<input type='password' name='moje_heslo' value='heslo' />
```

Vstup pro heslo

- Pro zadávání textových vstupů uživatele, které nebudou čitelné.
- Používá se nepárový tag `<input>`, který je typu: `password`.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou je pole předvyplněno.

Vstup pro heslo

```
<input type='password' name='moje_heslo' value='heslo' />
```



Checkbox

Checkbox

- Slouží pro „zatrhnutí“ nějaké možnosti. Výběr N hodnot z M .

Checkbox

- Slouží pro „zatrhnutí“ nějaké možnosti. Výběr N hodnot z M.
- Používá se nepárový tag `<input>`, který je typu: checkbox.

Checkbox

- Slouží pro „zatrhnutí“ nějaké možnosti. Výběr N hodnot z M.
- Používá se nepárový tag `<input>`, který je typu: checkbox.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou vstup bude mít, pokud je zatrhnut (zbytečné).

Checkbox

- Slouží pro „zatrhnutí“ nějaké možnosti. Výběr N hodnot z M.
- Používá se nepárový tag `<input>`, který je typu: checkbox.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou vstup bude mít, pokud je zatrhnut (zbytečné).
- Pokud není checkbox zatrhnut, neodesílá se.

Checkbox

- Slouží pro „zatrhnutí“ nějaké možnosti. Výběr N hodnot z M.
- Používá se nepárový tag `<input>`, který je typu: checkbox.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou vstup bude mít, pokud je zatrhnut (zbytečné).
- Pokud není checkbox zatrhnut, neodesílá se.

Checkbox

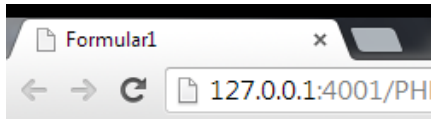
```
<input type='checkbox' name='muj_checkbox' value='hodnota' />
```

Checkbox

- Slouží pro „zatrnutí“ nějaké možnosti. Výběr N hodnot z M.
- Používá se nepárový tag `<input>`, který je typu: checkbox.
- Atribut `name` slouží k identifikaci vstupu, atribut `value` je hodnota, kterou vstup bude mít, pokud je zatrhnut (zbytečné).
- Pokud není checkbox zatrhnut, neodesílá se.

Checkbox

```
<input type='checkbox' name='muj_checkbox' value='hodnota' />
```



checkbox ☒

Radio - button

Radio - button

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.

Radio - button

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.
- Používá se nepárový tag `<input>`, který je typu: radio.

Radio - button

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.
- Používá se nepárový tag `<input>`, který je typu: radio.
- Atribut `name` slouží k identifikaci skupiny vstupů, atribut `value` je hodnota, kterou vstup bude mít, pokud je zatrhnut (nutné pro určení, který byl zatrhnut).

Radio - button

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.
- Používá se nepárový tag `<input>`, který je typu: radio.
- Atribut `name` slouží k identifikaci skupiny vstupů, atribut `value` je hodnota, kterou vstup bude mít, pokud je zatrhnut (nutné pro určení, který byl zatrhnut).

Radio - button

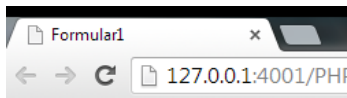
```
radio1<input type='radio' name='radio' value ='radio1' checked />  
radio2<input type='radio' name='radio' value ='radio2' />  
radio3<input type='radio' name='radio' value ='radio3' />
```

Radio - button

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.
- Používá se nepárový tag `<input>`, který je typu: radio.
- Atribut `name` slouží k identifikaci skupiny vstupů, atribut `value` je hodnota, kterou vstup bude mít, pokud je zatrhnut (nutné pro určení, který byl zatrhnut).

Radio - button

```
radio1<input type='radio' name='radio' value ='radio1' checked />  
radio2<input type='radio' name='radio' value ='radio2' />  
radio3<input type='radio' name='radio' value ='radio3' />
```



radio1 ☒
radio2 ☐
radio3 ☐

Select

Select

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.

Select

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.
- Používá se párový tag `<select>`.

Select

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.
- Používá se párový tag `<select>`.
- Atribut name slouží k identifikaci vstupu. Dále obsahuje tagy option, a ty mají atribut value. (nutné pro určení, jaká hodnota byla vybrána).

Select

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.
- Používá se párový tag <select>.
- Atribut name slouží k identifikaci vstupu. Dále obsahuje tagy option, a ty mají atribut value. (nutné pro určení, jaká hodnota byla vybrána).

Select

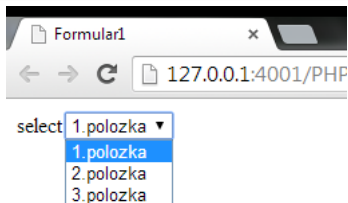
```
select<select name='select'>
  <option value='1'>1.polozka</option>
  <option value='2'>2.polozka</option>
  <option value='3'>3.polozka</option>
</select>
```

Select

- Slouží pro „vybrání“ nějaké možnosti. Výběr 1 hodnoty z N.
- Používá se párový tag <select>.
- Atribut name slouží k identifikaci vstupu. Dále obsahuje tagy option, a ty mají atribut value. (nutné pro určení, jaká hodnota byla vybrána).

Select

```
select<select name='select'>
  <option value='1'>1.polozka</option>
  <option value='2'>2.polozka</option>
  <option value='3'>3.polozka</option>
</select>
```



Submit

- Slouží k odeslání dat vyplněných v formuláři.

Submit

- Slouží k odeslání dat vyplněných v formuláři.
- Používá se nepárový tag `<input>`, který je typu: submit.

Submit

- Slouží k odeslání dat vyplněných v formuláři.
- Používá se nepárový tag `<input>`, který je typu: submit.
- Atribut name slouží k identifikaci vstupu. Atribut value je nápis, který bude na tlačítku (Zbytečné).

Submit

- Slouží k odeslání dat vyplněných v formuláři.
- Používá se nepárový tag `<input>`, který je typu: submit.
- Atribut name slouží k identifikaci vstupu. Atribut value je nápis, který bude na tlačítku (Zbytečné).
- Vždy se odesílá.

Submit

- Slouží k odeslání dat vyplněných v formuláři.
- Používá se nepárový tag `<input>`, který je typu: submit.
- Atribut name slouží k identifikaci vstupu. Atribut value je nápis, který bude na tlačítku (Zbytečné).
- Vždy se odesílá.

Submit

```
<input type='submit' name='submit' value='odeslat' />
```



Superglobální proměnné

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém scriptu. (Ne vždy, všechna)

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém scriptu. (Ne vždy, všechna)
- Tato pole jsou:

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém scriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém scriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).
 - `$_SERVER` - různé systémové proměnné např. údaje o serveru i uživateli, apod.

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém skriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).
 - `$_SERVER` - různé systémové proměnné např. údaje o serveru i uživateli, apod.
 - `$_ENV` - proměnné poskytované skriptu z prostředí.

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém skriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).
 - `$_SERVER` - různé systémové proměnné např. údaje o serveru i uživateli, apod.
 - `$_ENV` - proměnné poskytované skriptu z prostředí.
 - `$_POST` - proměnné poskytované skriptu přes HTTP POST.

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém skriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).
 - `$_SERVER` - různé systémové proměnné např. údaje o serveru i uživateli, apod.
 - `$_ENV` - proměnné poskytované skriptu z prostředí.
 - `$_POST` - proměnné poskytované skriptu přes HTTP POST.
 - `$_GET` - proměnné poskytované skriptu přes HTTP GET.

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém skriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).
 - `$_SERVER` - různé systémové proměnné např. údaje o serveru i uživateli, apod.
 - `$_ENV` - proměnné poskytované skriptu z prostředí.
 - `$_POST` - proměnné poskytované skriptu přes HTTP POST.
 - `$_GET` - proměnné poskytované skriptu přes HTTP GET.
 - `$_COOKIE` - proměnné poskytované skriptu přes HTTP cookies.

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém skriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).
 - `$_SERVER` - různé systémové proměnné např. údaje o serveru i uživateli, apod.
 - `$_ENV` - proměnné poskytované skriptu z prostředí.
 - `$_POST` - proměnné poskytované skriptu přes HTTP POST.
 - `$_GET` - proměnné poskytované skriptu přes HTTP GET.
 - `$_COOKIE` - proměnné poskytované skriptu přes HTTP cookies.
 - `$_FILES` - proměnné poskytované skriptu přes HTTP POST upload souborů.

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém skriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).
 - `$_SERVER` - různé systémové proměnné např. údaje o serveru i uživateli, apod.
 - `$_ENV` - proměnné poskytované skriptu z prostředí.
 - `$_POST` - proměnné poskytované skriptu přes HTTP POST.
 - `$_GET` - proměnné poskytované skriptu přes HTTP GET.
 - `$_COOKIE` - proměnné poskytované skriptu přes HTTP cookies.
 - `$_FILES` - proměnné poskytované skriptu přes HTTP POST upload souborů.
 - `$_REQUEST` - proměnné poskytované skriptu přes libovolný vstupní mechanismus (POST / GET / COOKIE...).

Superglobální proměnné

- V PHP používáme několik tzv. „superglobálních“ polí.
- To znamená, že je můžete použít v každém skriptu. (Ne vždy, všechna)
- Tato pole jsou:
 - `$GLOBALS` - všechny běžně používané proměnné v PHP (i všechny superglobální proměnné, vč. `$GLOBALS`).
 - `$_SERVER` - různé systémové proměnné např. údaje o serveru i uživateli, apod.
 - `$_ENV` - proměnné poskytované skriptu z prostředí.
 - `$_POST` - proměnné poskytované skriptu přes HTTP POST.
 - `$_GET` - proměnné poskytované skriptu přes HTTP GET.
 - `$_COOKIE` - proměnné poskytované skriptu přes HTTP cookies.
 - `$_FILES` - proměnné poskytované skriptu přes HTTP POST upload souborů.
 - `$_REQUEST` - proměnné poskytované skriptu přes libovolný vstupní mechanismus (POST / GET / COOKIE...).
 - `$_SESSION` - proměnné registrované pro aktuální session.

Získání dat z formulářů

- V PHP tedy data z formulářů získáme:

Získání dat z formulářů

- V PHP tedy data z formulářů získáme:
 - Z pole `$_GET`, pokud byl formulář odeslán metodou GET.

- V PHP tedy data z formulářů získáme:
 - Z pole `$_GET`, pokud byl formulář odeslán metodou GET.
 - Z pole `$_POST`, pokud byl formulář odeslán metodou POST.

- V PHP tedy data z formulářů získáme:
 - Z pole `$_GET`, pokud byl formulář odeslán metodou GET.
 - Z pole `$_POST`, pokud byl formulář odeslán metodou POST.
 - Z pole `$_REQUEST`, ať byl odeslán GET, nebo POST.

Získání dat z formulářů

- V PHP tedy data z formulářů získáme:
 - Z pole `$_GET`, pokud byl formulář odeslán metodou GET.
 - Z pole `$_POST`, pokud byl formulář odeslán metodou POST.
 - Z pole `$_REQUEST`, ať byl odeslán GET, nebo POST.
- Důležitý je atribut `name` u formulářového prvku.

Získání dat z formulářů

- V PHP tedy data z formulářů získáme:
 - Z pole `$_GET`, pokud byl formulář odeslán metodou GET.
 - Z pole `$_POST`, pokud byl formulář odeslán metodou POST.
 - Z pole `$_REQUEST`, ať byl odeslán GET, nebo POST.
- Důležitý je atribut `name` u formulářového prvku.
- Podle tohoto jména vznikne v superglobálním poli položka a její hodnota bude mít hodnotou, která byla vypněna ve formuláři.

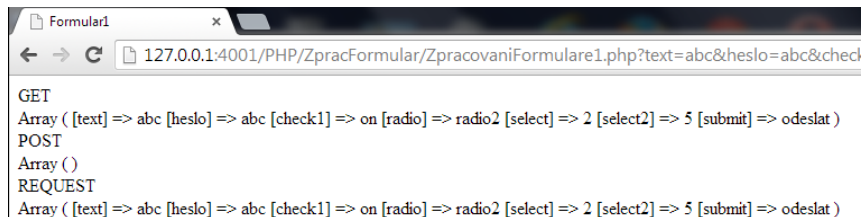
Výpis polí

Ukázka výpisu superglob. polí

```
print_r($_GET);  
print_r($_POST);  
print_r($_REQUEST);
```

Ukázka výpisu superglob. polí

```
print_r($_GET);  
print_r($_POST);  
print_r($_REQUEST);
```



```
GET  
Array ( [text] => abc [heslo] => abc [check1] => on [radio] => radio2 [select] => 2 [select2] => 5 [submit] => odeslat )  
POST  
Array ()  
REQUEST  
Array ( [text] => abc [heslo] => abc [check1] => on [radio] => radio2 [select] => 2 [select2] => 5 [submit] => odeslat )
```


- Zopakovali jsme si základní formulářové prvky jazyka HTML.

- Zopakovali jsme si základní formulářové prvky jazyka HTML.
- Ukázali jsme si jak dostat jejich obsah do PHP.



T. Converse, J. Park, C. Morgan.

PHP5 and MySQL Bible

Wiley Publishing, Inc., 2004.

ISBN 0-7645-5746-7

- Tento materiál je určen pro bezplatné používání pro potřeby výuky a vzdělávání na všech typech škol a školských zařízení. Jakékoliv další využití podléhá autorskému zákonu.
- Všechna neocitovaná autorská díla jsou dílem autora.