

Processorji

Izdelava spletne aplikacije s PHP in MySQL podatkovno bazo


Izdelajte spletno aplikacijo s skriptnim jezikom PHP, ki bo omogočala upravljanje podatkovne baze MySQL podatkovnega strežnika. Najprej je potrebno ustvariti podatkovno bazo in tabelo, v katero bomo shranjevali podatke o procesorjih.

Podatki so shranjeni v podatkovni bazi lokalnega podatkovnega strežnika MySQL. Podatki za dostop do strežnika so:

- ime strežnika: **localhost**,
- uporabniško ime: **root**,
- geslo: **admin** in
- ime podatkovne baze: **procesorji**.

Najprej ustvarite podatkovno bazo **procesorji** in tabelo **procesor**, ki vsebuje pet podatkov:

- **id** (celo število, ki se samodejno povečuje),
- **naziv** (besedilo dolgo 30 znakov),
- **znamka** (besedilo dolgo 30 znakov),
- **cena** (float) in
- **stevilo_jeder** (celo število).

#	Ime	Vrsta	Pravilo za razvrščanje znakov	Atributi	Null	Privzeto	Pripombe	Dodatno
<input type="checkbox"/>	1 id 	int			Ne	Brez		AUTO_INCREMENT
<input type="checkbox"/>	2 naziv	varchar(30)	utf8_slovenian_ci		Ne	Brez		
<input type="checkbox"/>	3 znamka	varchar(30)	utf8_slovenian_ci		Ne	Brez		
<input type="checkbox"/>	4 cena	float			Ne	Brez		
<input type="checkbox"/>	5 stevilo_jeder	int			Ne	Brez		

Slika 1: Tabela procesor

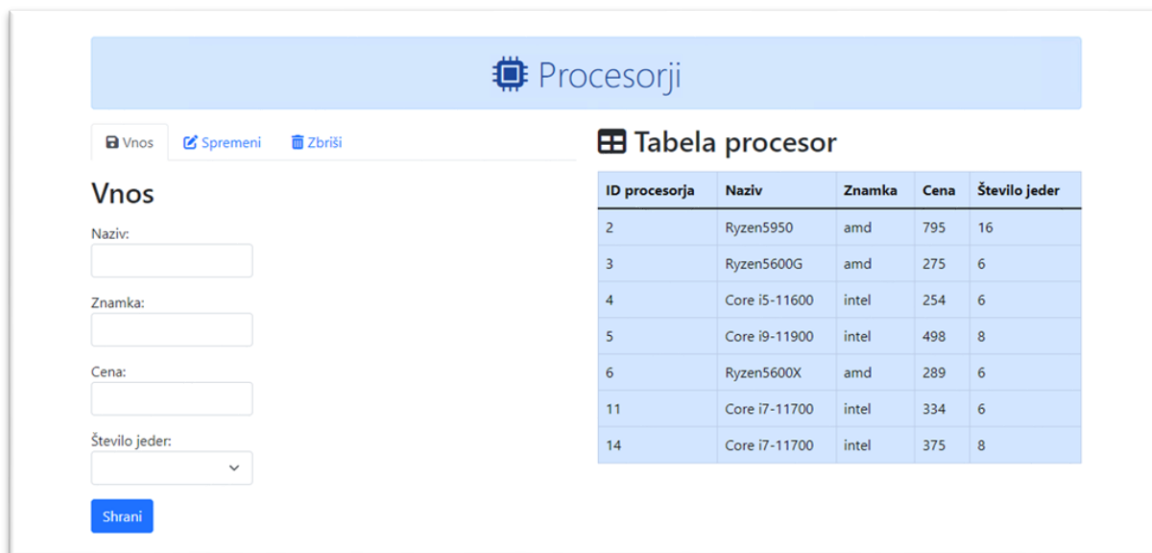
Spletna aplikacija prikazuje uporabo štirih najpogostejših SQL stavkov za delo s podatkovnimi bazami:

- INSERT (vstavljanje podatkov v tabelo),
- UPDATE (spreminjanje podatkov v tabeli),
- DELETE (brisanje podatkov v tabeli) in
- SELECT (prikaz podatkov iz tabele).

Oblikovanje spletne aplikacije

V mapi svojega uporabniškega računa (**omrežni disk M:**) izdelajte novo mapo z imenom **procesor** in jo odprite v programu **PHPStorm**. Kreirajte novo php datoteko (new php file) z imenom **index.php**.

Za oblikovanje spletne strani lahko uporabite ogrodje Bootstrap ali pa s pomočjo ukazov HTML in CSS dosežete ustrezno postavitev elementov (glej sliko):



Slika 2: Izgled spletne aplikacije

Spletna stran ima na vrhu naslov (**Procesorji**), pod njim pa vrstico, ki je razdeljena na dva stolpca, ki sta v razmerju 1:1. V levem so obrazci za vnos podatkov, spreminjanje in brisanje podatkov tabele, v desnem pa izpis podatkov iz tabele (glej sliko zgoraj).

Za dostop do podatkovne baze v PHP-ju definiramo štiri konstante, ki jih uporabljamo pri ukazih za vnos, spreminjanje, brisanje in prikaz podatkov:

```
const DB_SERVER = "localhost";
const DB_USER = "root";
const DB_PASS = "admin";
const DB_NAME = "procesorji";
```

Pri vseh ukazih za upravljane podatkovne baze se najprej povežemo s podatkovnim strežnikom:

```
$conn = mysqli_connect(DB_SERVER,DB_USER,DB_PASS,DB_NAME);
```

Nato zapišemo SQL stavek, ki ga izvedemo z naslednjim ukazom:

```
mysqli_query($conn,$sql);
```

Na koncu ne pozabimo zapreti povezave s podatkovnim strežnikom:

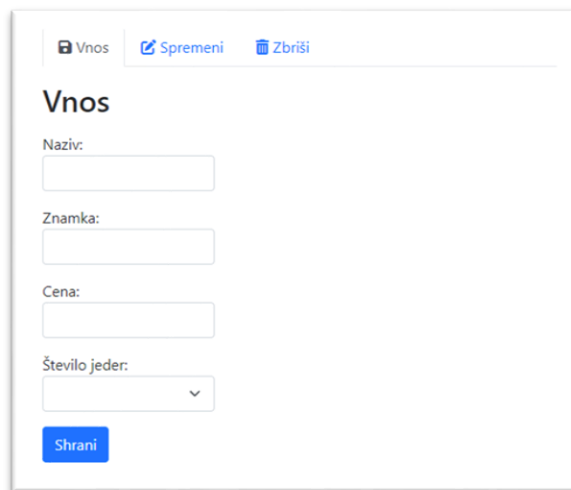
```
mysqli_close($conn);
```

Vnos podatkov

Za vnos podatkov uporabite ustrezne html elemente (**input**). Po vnosu podatkov in kliku na gumb **Shrani**, se podatki shranijo v tabelo **procesor**. Za shranjavanje v tabelo uporabimo SQL ukaz INSERT:

```
$sql = "INSERT INTO procesor (naziv, znamka, cena, stevilo_jeder) VALUES ('$naziv','$znamka','$cena','$st_jeder')";
```

Podatke za vnos preberemo iz obrazca za vnos.



The screenshot shows a web form titled "Vnos" with three action buttons: "Vnos", "Spremeni", and "Zbriši". Below the title, there are four input fields: "Naziv:", "Znamka:", "Cena:", and "Število jeder:". The "Število jeder:" field is a dropdown menu. At the bottom of the form is a blue button labeled "Shrani".

Slika 3: Vnos podatkov v tabelo procesor

Spreminjanje podatkov

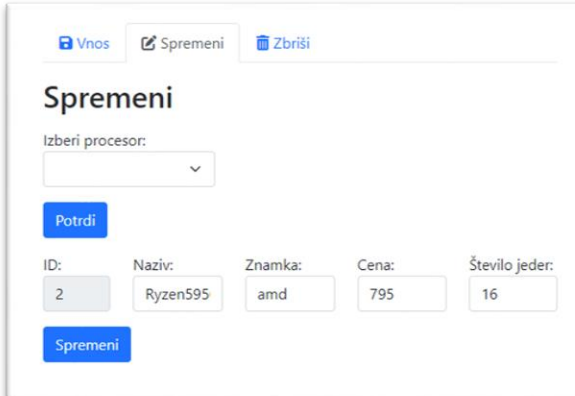
Za spreminje podatkov najprej uporabite html element **select**, s katerim izberemo podatek, ki ga želimo spremeniti. Element select napolnimo s podatki iz tabele. Za to uporabimo SQL ukaz SELECT:

```
<select name="procesor_spremeni" class="form-select" required>
  <option></option>
  <?php
    $conn = mysqli_connect(DB_SERVER,DB_USER,DB_PASS,DB_NAME);
    $sql = "SELECT * FROM procesor";
    $result = mysqli_query($conn,$sql);
    while ($row = mysqli_fetch_array($result))
    {
      echo "<option value='". $row['id']."'>". $row['naziv']."</option>";
    }
    mysqli_close($conn);
  ?>
</select>
```

Po potrditvi izbranega podatka se njegovi podatki zapišejo v vnosna polja, kjer jih lahko spreminjamo. Za spreminjanje podatkov uporabimo SQL ukaz UPDATE:

```
$sql = "UPDATE procesor SET naziv='". $naziv."', znamka='". $znamka."',
cena='". $cena."', stevilo_jeder='". $st_jeder.'" WHERE id='". $id_spremeni.'";
```

Podatke za spreminjanje preberemo iz obrazca za spreminjanje.



ID:	Naziv:	Znamka:	Cena:	Število jeder:
2	Ryzen595	amd	795	16

Slika 4: Spreminjanje podatkov tabele procesor

Brisanje podatkov

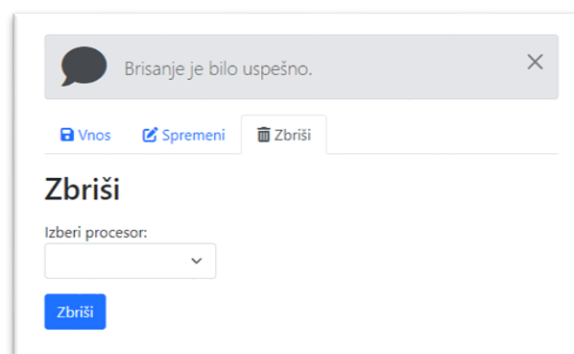
Za brisanje podatkov najprej uporabite html element **select**, s katerim izberemo podatek, ki ga želimo izbrisati. Element select napolnimo s podatki iz tabele. Za to uporabimo SQL ukaz SELECT:

```
<select name="procesor_spremeni" class="form-select" required>
  <option></option>
  <?php
  $conn = mysqli_connect(DB_SERVER,DB_USER,DB_PASS,DB_NAME);
  $sql = "SELECT * FROM procesor";
  $result = mysqli_query($conn,$sql);
  while ($row = mysqli_fetch_array($result))
  {
    echo "<option value='". $row['id']."'>". $row['naziv']."</option>";
  }
  mysqli_close($conn);
  ?>
</select>
```

Po izbiri podatka in kliku na gumb **Zbriši** se njegovi podatki izbrišejo iz tabele. Za brisanje podatkov uporabimo SQL ukaz DELETE:

```
$sql = "DELETE FROM procesor WHERE id='$id';"
```

Po vsaki izvršitvi ukaza (vnos, spreminjanje, brisanje) izpišemo ustrezno obvestilo (glej sliko spodaj):



Slika 5: Brisanje podatkov tabele procesor

Prikaz podatkov iz tabele procesor

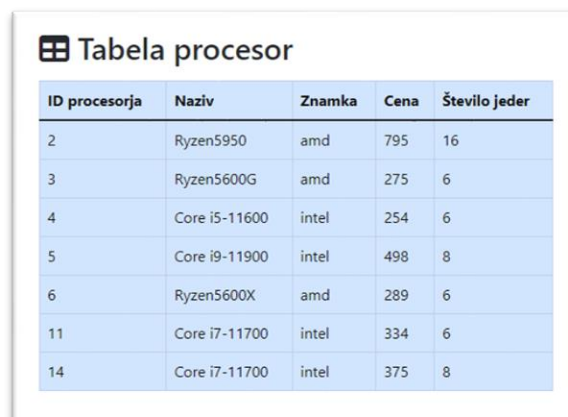
Za prikaz podatkov iz tabele uporabimo SQL stavek SELECT:

```
$sql = "SELECT * FROM procesor";
```

Rezultat poizvedbe shranimo v spremenljivko (**\$result**), ki vsebuje vse zapise iz tabele **procesor**. Po zapisih se sprehodimo z zanko **while**, v kateri preberemo posamezne stolpce tabele in jih izpišemo v vrstico tabele (**<tr>**).

```
$result = mysqli_query($conn,$sql);
while ($row = mysqli_fetch_array($result))
{
    echo "<tr>";
    echo "<td>".$row["id"]."</td>";
    echo "<td>".$row["naziv"]."</td>";
    echo "<td>".$row["znamka"]."</td>";
    echo "<td>".$row["cena"]."</td>";
    echo "<td>".$row["stevilo_jeder"]."</td>";
    echo "</tr>";
}
```

Za oblikovanje tabele lahko uporabite **Bootstrap** temo (glej sliko spodaj):



ID procesorja	Naziv	Znamka	Cena	Število jeder
2	Ryzen5950	amd	795	16
3	Ryzen5600G	amd	275	6
4	Core i5-11600	intel	254	6
5	Core i9-11900	intel	498	8
6	Ryzen5600X	amd	289	6
11	Core i7-11700	intel	334	6
14	Core i7-11700	intel	375	8

Slika 6: Prikaz podatkov iz tabele procesor

Ocenjevanje

Ocenjuje se tako oblika kot funkcionalnost. Pri **obliki** se ocenjuje uporaba ustreznih elementov in slogov ter ujemanje s predlogo:

- osnovna struktura html (**10 točk**)
- razmerje in postavitve blokov (**10 točk**)
- vnos podatkov (**10 točk**)
- izpis podatkov (**10 točk**)
- naslovi (**10 točk**)

Pri **funkcionalnosti** se ocenjuje uporaba ustreznih metod, postopkov in ukazov:

- preverjanje vnosa (**10 točk**)
- povezava s strežnikom (**10 točk**)
- zapis v bazo (**10 točk**)
- izpis podatkov (**10 točk**)
- obrazec (**10 točk**)

Točkovanje:

Oblika			
št. možnih točk	element	točkovnik	točke
10	osnovna struktura	html, head, title, body	
10	bloki	postavitev, razmerje	
10	vnos podatkov	vnosni obrazec	
10	izpis podatkov	ustrezni elementi in oblika (tabela)	
10	naslovi	heading in barva	
50	SKUPAJ		
Funkcionalnost			
št. možnih točk	element	točkovnik	točke
10	vnos	elementi obrazca	
10	povezava	povezava s podatkovno bazo	
10	zapis v bazo	zapis v bazo	
10	izpis tabele	tabela procesor	
10	obrazec	branje podatkov obrazca	
50	SKUPAJ		
100	SKUPAJ		

Kriterij ocenjevanja:

Število možnih točk:		100	Načrtovanje in razvoj spletnih aplikacij Priimek in ime: _____ Datum: _____ Število točk: ____ / 100 Ocena: _____
točke (meje)	točke (zgornja meja)	ocena	
0 - 49	49 % = 49,00	nezadostno (1)	
50 - 62	62 % = 62,00	zadostno (2)	
63 - 75	75 % = 75,00	dobro (3)	
76 - 88	88 % = 88,00	prav dobro (4)	
89 - 100	100 % = 100,00	odlično (5)	