Univerzitet u Beogradu

Fakultet organizacionih nauka

Katedra za elektronsko poslovanje

PHP OOP MYSQL

Domaći zadatak 1

|  |  |  |
| --- | --- | --- |
| Ime | Prezime | Broj indeksa |
| Lazar | Grebović | 1060/2021 |
| Link ka Github-u | <https://github.com/1lazar/Trke-domaci1.git> | |

Sadržaj

[1 Korisnički zahtev 3](#_Toc1822586367)

[2 Implementacija 3](#_Toc2044682512)

[3 Korisničko uputstvo 8](#_Toc257807579)

# Korisnički zahtev

Ime mog domaćeg su Trke. Tu Vam pokazujem vozače i automobile i pokazujem koji vozači voze koja kola. U domaćem možete da dodate novog vozača ili da ga obrišete već postojećeg vozača u bazi ili da vidite koja kola oni voze, a kod automobila možete novi automobil ili da obrišete već postojeći u bazi.

Slučajevi korišćenja:

**SK1: Slučaj korišćenja – Kreiranje novog vozača**

**Naziv SK**

Kreiranje novog vozača

**Aktori SK**

Administrator

**Učesnici SK**

Administrator i sistem (program)

**Preduslov:** Sistem je uključen i administrator je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa vozačem.

**Osnovni scenario SK**

1. Administrator poziva sistem da kreira novi vozač. (APSO)
2. Sistem kreira novi vozač. (SO)
3. Sistem prikazuje administratoru novi vozač i poruku: “Sistem je kreirao novi vozač“. (IA)
4. Administrator unosi podatke u novi vozač. (APUSO)
5. Administrator kontroliše da li je korektno uneo podatke u novi vozač. (ANSO)
6. Administrator poziva sistem da zapamti podatke o vozaču. (APSO)
7. Sistem pamti podatke o vozaču. (SO)
8. Sistem prikazuje administratoru zapamćeni vozač i poruku: “Sistem je zapamtio vozač”. (IA)
9. Administrator poziva sistem da obradi vozač. (APSO)
10. Sistem obrađuje vozač.(SO)
11. Sistem prikazuje administratoru obrađen vozač i poruku: “Sistem je obradio vozač“.(IA)

**Alternativna scenarija**

3.1 Ukoliko sistem ne može da kreira vozač on prikazuje administratoru poruku: “Sistem ne može da kreira novi vozač”. Prekida se izvršenje scenaria. (IA)

8.1 Ukoliko sistem ne može da zapamti podatke o vozaču on prikazuje administratoru poruku “Sistem ne može da zapamti vozač”. Prekida se izvršenje scenaria. (IA)

11.1 Ukoliko sistem ne može da obradi vozač on prikazuje administratoru poruku: “Sistem ne može da obradi vozač”. (IA)

**SK2: Slučaj korišćenja – Pretraživanje vozača**

**Naziv SK**

Pretraživanje vozača

**Aktori SK**

Administrator

**Učesnici SK**

Administrator i sistem (program)

**Preduslov:** Sistem je uključen i administrator je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa vozačem.

**Osnovni scenario SK**

1. Administrator unosi vrednost po kojoj pretražuje vozače. (APUSO)
2. Administrator poziva sistem da nađe vozače po zadatoj vrednosti. (APSO)
3. Sistem traži vozače po zadatoj vrednosti. (SO)
4. Sistem prikazuje administratoru podatke o vozačima i poruku: “Sistem je našao vozače po zadatoj vrednosti”. (IA)
5. Sistem bira vozača. (APUSO)
6. Administrator poziva sistem da učita vozača. (APSO)
7. Sistem učitava vozača. (SO)
8. Sistem prikazuje administratoru podatke o vozačima i poruku: “Sistem je učitao vozače po zadatoj vrednosti”. (IA)

**Alternativna scenarija**

4.1 Ukoliko sistem ne može da nađe vozača on prikazuje administratoru poruku: “Sistem ne može da nađe vozače po zadatoj vrednosti”.Prekida se izvršavanje scenarija. (IA)

8.1 Ukoliko sistem ne može da učita vozača on prikazuje administratoru poruku: “Sistem ne može da učita vozača ”. (IA)

**SK3: Slučaj korišćenja – Brisanje vozača**

**Naziv SK**

Brisanje vozača

**Aktori SK**

Administrator

**Učesnici SK**

Administrator i sistem (program)

**Preduslov:** Sistem je uključen i administrator je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa vozačem.

**Osnovni scenario SK**

1. Administrator unosi vrednost po kojoj pretražuje vozače. (APUSO)
2. Administrator poziva sistem da nađe vozače po zadatoj vrednosti. (APSO)
3. Sistem traži vozače po zadatoj vrednosti. (SO)
4. Sistem prikazuje administratoru podatke o vozačima i poruku: “Sistem je našao vozače po zadatoj vrednosti”. (IA)
5. Administrator bira vozača. (APUSO)
6. Administrator poziva sistem da učita vozača. (APSO)
7. Sistem učitava vozače. (SO)
8. Sistem prikazuje administratoru podatke o vozaču i poruku: “Sistem je učitao vozača”. (IA)
9. Administrator poziva sistem da obriše podatke o vozaču. (APSO)
10. Sistem briše vozaču. (SO)
11. Sistem prikazuje administratoru poruku: “Sistem je obrisao podatke o vozaču.” (IA)

**Alternativna scenarija**

4.1 Ukoliko sistem ne može da nađe vozače on prikazuje administratoru poruku: “Sistem ne može da nađe vozače po zadatoj vrednosti”. Prekida se izvršenje scenarija. (IA)

8.1 Ukoliko sistem ne može da učita vozača on prikazuje administratoru poruku: “Sistem ne može da učita vozača”. Prekida se izvršavanje scenarija (IA)

11.1 Ukoliko sistem ne može da obriše vozača on prikazuje administratoru poruku “Sistem ne može da obriše vozača”. (IA)

**SK4: Slučaj korišćenja – Kreiranje novog automobila**

**Naziv SK**

Kreiranje novog automobila

**Aktori SK**

Administrator

**Učesnici SK**

Administrator i sistem (program)

**Preduslov:** Sistem je uključen i administrator je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa automobilom.

**Osnovni scenario SK**

1. Administrator poziva sistem da kreira novi automobil. (APSO)
2. Sistem kreira novi automobil. (SO)
3. Sistem prikazuje administratoru novi automobil i poruku: “Sistem je kreirao novi automobil“. (IA)
4. Administrator unosi podatke u novi automobil. (APUSO)
5. Administrator kontroliše da li je korektno uneo podatke u novi automobil. (ANSO)
6. Administrator poziva sistem da zapamti podatke o automobilu. (APSO)
7. Sistem pamti podatke o automobilu. (SO)
8. Sistem prikazuje administratoru zapamćeni automobil i poruku: “Sistem je zapamtio automobil”. (IA)
9. Administrator poziva sistem da obradi automobil. (APSO)
10. Sistem obrađuje automobil.(SO)
11. Sistem prikazuje administratoru obrađen automobil i poruku: “Sistem je obradio automobil“.(IA)

**Alternativna scenarija**

3.1 Ukoliko sistem ne može da kreira automobil on prikazuje administratoru poruku: “Sistem ne može da kreira novi automobil”. Prekida se izvršenje scenaria. (IA)

8.1 Ukoliko sistem ne može da zapamti podatke o automobilu on prikazuje administratoru poruku “Sistem ne može da zapamti automobil”. Prekida se izvršenje scenaria. (IA)

11.1 Ukoliko sistem ne može da obradi automobil on prikazuje administratoru poruku: “Sistem ne može da obradi automobil”. (IA)

**SK5: Slučaj korišćenja – Brisanje vozača**

**Naziv SK**

Brisanje automobila

**Aktori SK**

Administrator

**Učesnici SK**

Administrator i sistem (program)

**Preduslov:** Sistem je uključen i administrator je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa automobilom.

**Osnovni scenario SK**

1. Administrator unosi vrednost po kojoj pretražuje automobile. (APUSO)
2. Administrator poziva sistem da nađe automobile po zadatoj vrednosti. (APSO)
3. Sistem traži automobile po zadatoj vrednosti. (SO)
4. Sistem prikazuje administratoru podatke o automobilima i poruku: “Sistem je našao automobile po zadatoj vrednosti”. (IA)
5. Administrator bira automobila. (APUSO)
6. Administrator poziva sistem da učita automobil. (APSO)
7. Sistem učitava automobil. (SO)
8. Sistem prikazuje administratoru podatke o automobilu i poruku: “Sistem je učitao automobil”. (IA)
9. Administrator poziva sistem da obriše podatke o automobilu. (APSO)
10. Sistem briše automobilu. (SO)
11. Sistem prikazuje administratoru poruku: “Sistem je obrisao podatke o automobilu.” (IA)

**Alternativna scenarija**

4.1 Ukoliko sistem ne može da nađe vozače on prikazuje administratoru poruku: “Sistem ne može da nađe vozače po zadatoj vrednosti”. Prekida se izvršenje scenarija. (IA)

8.1 Ukoliko sistem ne može da učita vozača on prikazuje administratoru poruku: “Sistem ne može da učita vozača”. Prekida se izvršavanje scenarija (IA)

11.1 Ukoliko sistem ne može da obriše vozača on prikazuje administratoru poruku “Sistem ne može da obriše vozača”. (IA)

# Implementacija

CRUD = Create, Read, Update, Delete. Ja sam u svojoj aplikaciji implementirao create,read i delete.

Create operacija sluzi da kreiramo vozaca ili automobila u bazi podataka.

Read operacija sluzi da ucitamo vozace ili automobile iz baze i u koliko zelimo prikazemo u aplikaciji.

Delete operacija sluzi da obrisemo vozaca ili automobila iz baze podataka.

Prilikom klika na dugme on proverava podatke da li su svi unesni pa onda ubacuje,brise ili pretrazuje.

Ove operacije su radjena na osnovu query,php i js.

# Korisničko uputstvo

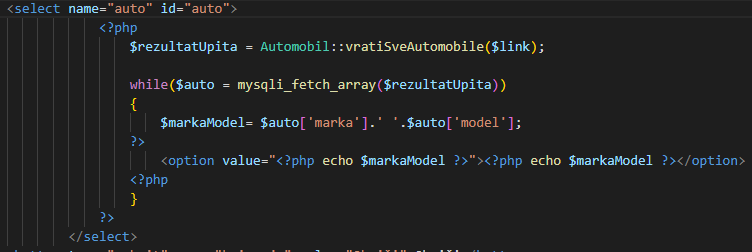


Opis ajaxa ucitajTekst gde uzima tekst iz text.txt i ispisuje na ekran prilikom klika na dugme,a skloniDiv ona samo obrise tj. skloni nas tekst.

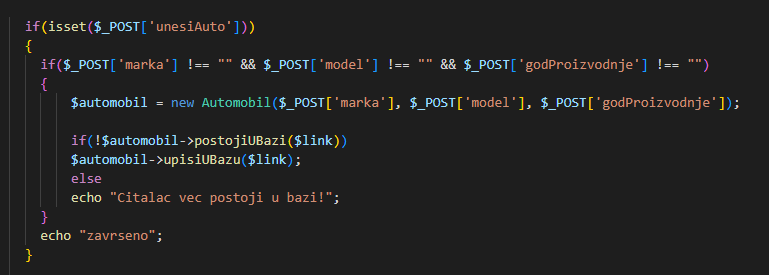


Ovde je prikazan primena ajaxa gde ovaj put ne uzimamo tekst iz .txt vec uzima iz .json i ispisujemo ga na ekranu. Kod filmovi.json izgleda ovako:





U ovom delu kod uzimamo sve automobile iz baze i njih ispisujemo u Combo box-u gde se listuju svi automobili.



Prilikom unosenja automobila u bazu moramo prvo da ispitamo da li je korisnik uneo obavezne podatke ako nije nece ga uneti u bazi. Posle toga uzimamo informacije koji je korisnik uneo i proveravamo da li ga ima u bazi, ukoliko postoji nece se upisati,a u slucaju da nema ga u bazi mi ga upisujemo.



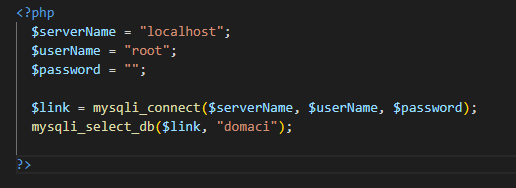
U ovom kodu mi zelimo da obrisemo automobil. Prvo iz Combo box-a izaberemo automobil pa uzimamo iz baze sve automobili. Proveravamo da li izabrani automobil jednak sa nekim automobilom iz baze pa ga obrisemo taj automobil.



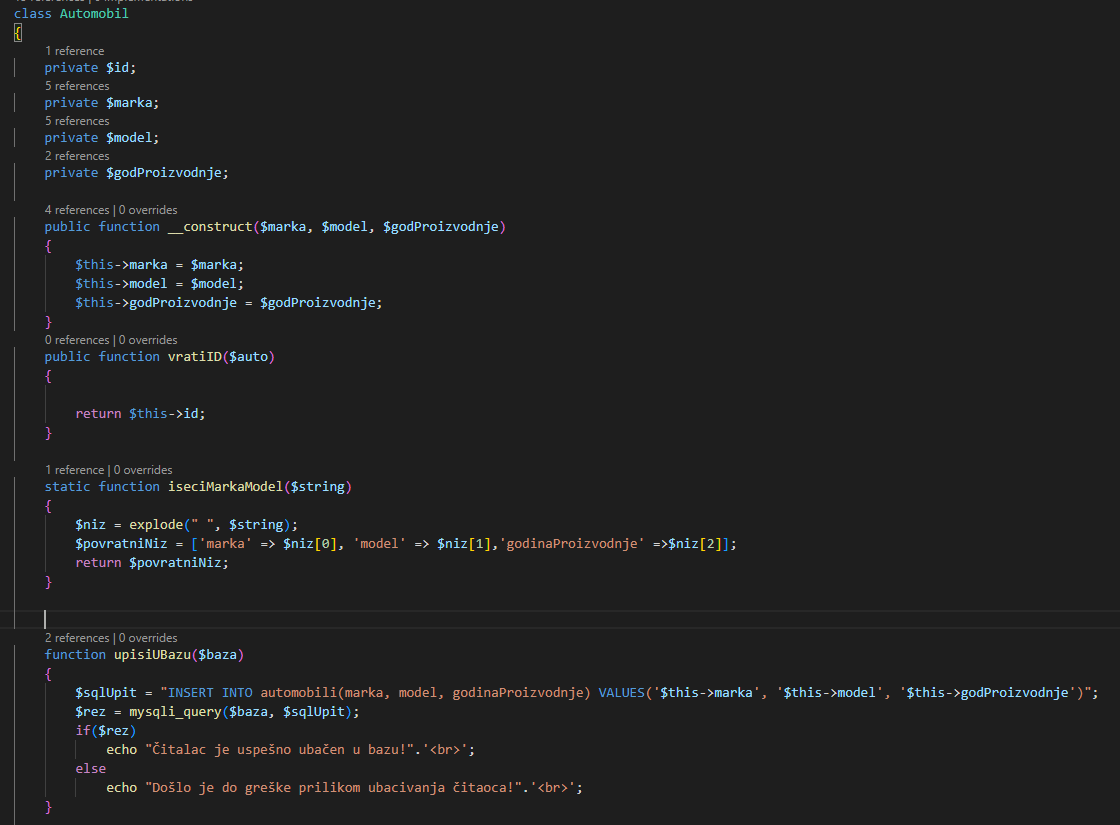
Dodavanjem vozaca u bazi prvo ako smo selektovali iz combo box-a auto onda samo isecemo na marku,model i godinu proizvodnje. Ukoliko nema tog auta korisnik unosi informacije gde mi i proveravamo da li je uneo i onda njih upisujemo u promenljive, pa te informacije unosimo u bazu. Posle toga unosimo vozaca pa uzimamo id automobila i proveravamo da li je korisnik uneo informacije pa onda proveravamo da li tog vozaca imamo u bazi, ako ne unosimo ga u bazu.

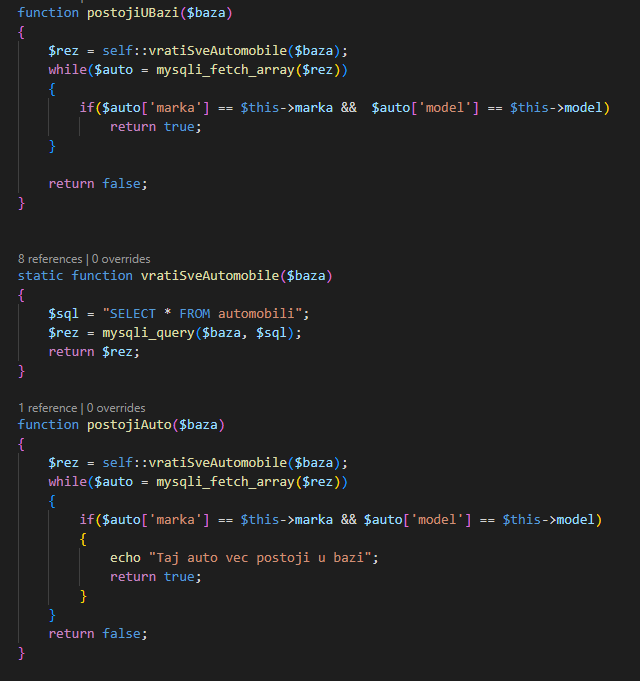


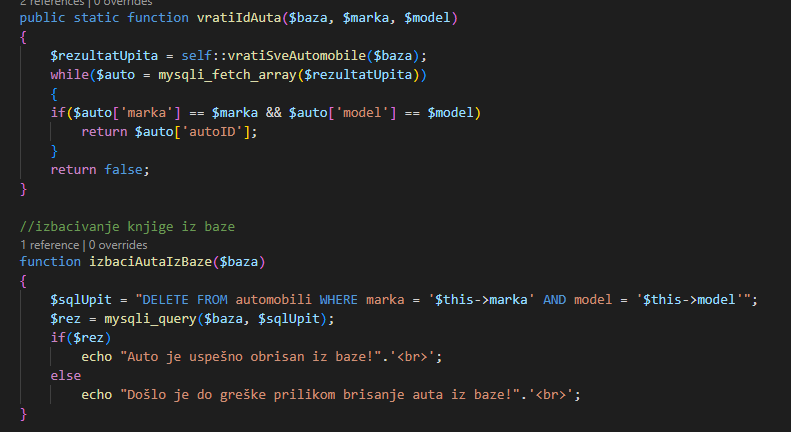
U ovom kodu proveravamo koji auto vozac vozi. Uzimamo izabrani auto i vozace iz baze. Ispisujemo tabelu sa nazivima kolonama. Posle ponalazimo id automobila i posle proveravamo id automobila sa id auta koji vozac vozi i ako se podudare ispisi jos jednu kolonu tabele da imenom i prezimenom vozaca i naziv automobila.



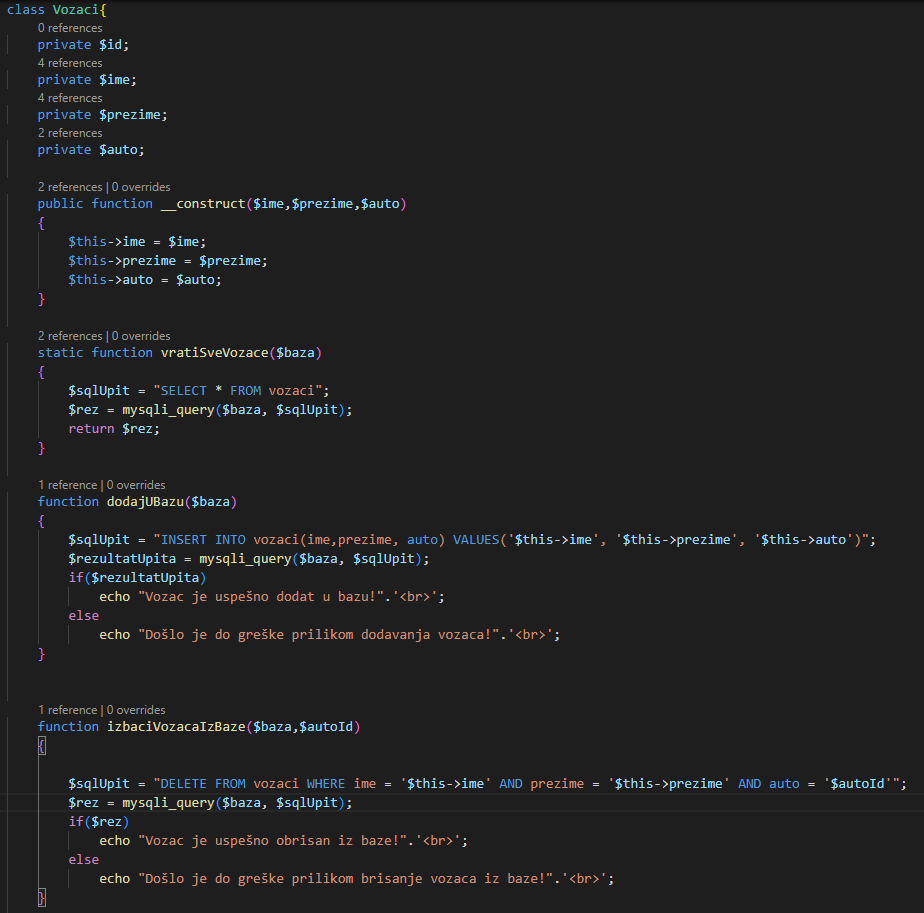
U ovom kodu je prikazano povezivanje nase aplikacije sa bazom podataka.

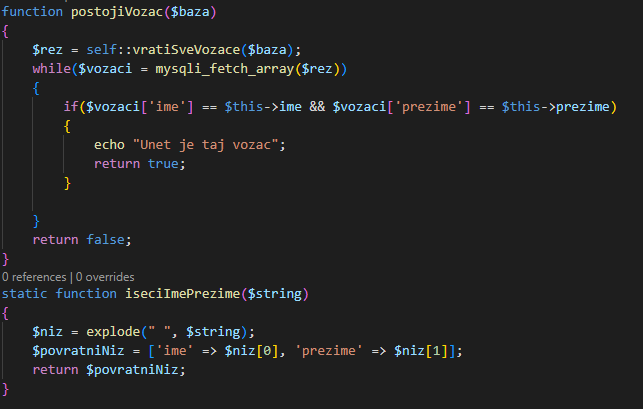






Ovde je definisana klasa Automobil koji sadrzi konstruktor,upisivanje u bazu, izbacivanje iz baze, da li postoji u bazi, da vrati sve automobile iz baze.





Ovde je definisana klasa Vozaci koja sadrzi konstruktor, da vrati sve vozace iz baze, da izbrise vozaca , da proveri da li postoji vozac, da isece string u ime i prezime.



U JavaScript proveravamo unete vrednosti da li su unete sve vrednosti u koliko nisu unete neke vrednosti ispisace se da morate da popunite sve podatke.



Prvo proveravamo da su uneti svi podaci za vozaca pa zatim da li su svi podaci uneti vezano za automobile.