

Испит по Имплементација на системи со отворен код

време за работа: 150 мин.

7 јуни 2018 г.

Генерално упатство и достава на решенијата:

Најавете се на компјутерот со Вашето корисничко име и лозинка. Испитот се состои од практичен дел. Задачите од практичниот дел се поставени на порталот <http://courses.finki.ukim.mk/>.

Пред завршување, потребно е да креирате фолдер именуван во следниов формат: `index_ime_prezime` (пр. `11111_Mitko_Mitkovski`). Во тој фолдер треба да Ви бидат решенијата на задачите. Фолдерот на крај треба да го зипувате (користете 7Zip за да го компресирате фолдерот) и да го прикачите на соодветното место на курсот.

Техничко упатство:

Работа со PHP интерпретерот: Потребно е да користите PHP 7.x интерпретер, кој треба да Ви биде достапен во фолдер во `C:\` (пр. `C:\php71\`). **Внимавајте, може да има повеќе верзии од PHP (пр. `php5.6`, `php5.4`), Ве молиме користете минимална верзија 7.0 на PHP.** За работа со

Базата која ќе ја користите се наоѓа на сервер достапен на мрежа. При логирање треба да ги користите следните податоци:

- **database host** : `praktikum.finki.ukim.mk` или `194.149.136.28`
- **username**: вашиот број на индекс (пр. `123456`). Доколку индексот е во друг формат треба да се тргнат знаците кои не се бројки и да се додадат 0 од напред доколку има потреба за да се стигне до 6 цифри. Според тоа, доколку индексот е `123/09`, тоа ќе соодветствува на `012309`.
- **password**: `Lozinka[br_index]`, каде што `[br_index]` е бројот на Вашиот индекс без средните загради (пр. `Lozinka123456`).
- **schema**: `isok[br_index]`, каде што `[br_index]` е бројот на Вашиот индекс без средните загради (пр. `isok123456`).

Детално упатство за подесување на Laravel 5.x ви е дадено во документот `Laravel-Webstorm.pdf`. Даде-на Ви е и архива `15.7z` која претставува подготвена `Laravel 5.5` околина, во која се инсталирани пакетите `„laravelcollective/html“` и `„marcuscampos/scaffold“`, кои се веќе вклучени во `„/config/app.php“` фајл.

Документација за PHP и Laravel Ви е дадена на <http://devdocs.finki.ukim.mk/>. Дополнително може да ги користите материјалите кои Ви се достапни на соодветниот курс за предметот. Сите останати материјали се забранети.

1 Задача

Напишете функција која што за дадено поле (array) ќе ги најде минималните 2 броеви во полето.

Полето може да има елементи полиња, броеви и стрингови. Бројот на вгнездени полиња не е познат (може да има поле во друго поле, во трето поле, итн...).

пример:

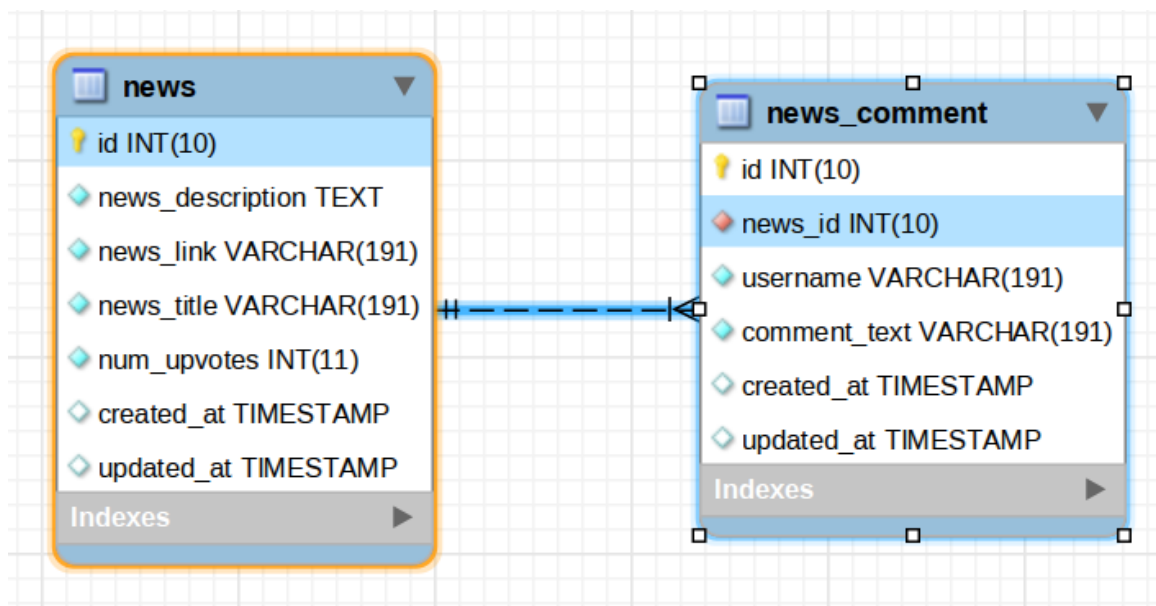
```
$niza = array(
    array(5, 1.25 , 15),
    array('unknown', 0.75 , 25, array(2, 'bla', 5) ),
    array(array('text'), 1.15 , 7)
);
```

Функцијата треба да врати поле со два елементи (0.75 и 1.15)

2 Задача

Со помош на Laravel 5 да се направи веб страна за приказ и гласање на вести.

Дијаграмот за базата на податоци е даден на следната слика. Додатно се креирани соодветни миграции за двете табели како и иницијални податоци.



Сајтот се состои од 2 контролери (за веста и коментарите соодветно) кои треба да ги имаат следните функционалности.

2.1 Почетна страна

При пристап на апликацијата потребно е да има листа на најновите 3 вести (разделени во div елементи поставени еден под друг). За секоја вест се прикажуваат:

- наслов на веста (кое е истовремено хиперлинк кон веста) (`news_title` и `news_link` полињата од базата).
- линкови за `upvote` и `downvote` (`num_upvotes` полето од базата)
- линк кон коментарите на веста (редиректира кон посебна страница)
- време кога веста е внесена во системот (`created_at` полето од базата)

2.2 Листање на коментар

При кликање на линкот кон коментарите потребно е да се прикаже нова страна во која се прикажува:

- наслов на веста (кое е истовремено хиперлинк кон веста) (`news_title` и `news_link` полињата од базата)

- линкови за `upvote` и `downvote` (`num_upvotes` полето од базата)
- описот на веста (`news_description`)
- време кога веста е внесена во системот (`created_at` полето од базата)
- сите коментари кои досега се внесени за дадената вест (земени од `news_comments` табелата) сортирани според датумот на внесување.

Информациите за веста се пристапуваат преку следното URL: `'news/[news_id]'` каде што `[news_id]` претставува `id`-то на веста (колона од табелата `news`).

2.3 CRUD операции за вест

Потребно е да се креираат CRUD операции за секоја вест, следејќи ги стандардите на Laravel. Тоа подразбира внесување на нова вест (со сите податоци), едитирање на податоците за дадена вест и бришење од базата на податоци. Потребно е при секој внес или измена на податоците во базата да се изврши валидација. Насловот на веста треба да има минимум 10 карактери. Линкот на веста треба да биде валидно URL. При приказ на секој лек, потребно е да има линк кон неговите коментари (подзадача 2.4).

2.4 CRUD операции за коментар

За секоја вест да се овозможи додавање на коментар за дадена вест. Пристапот до коментарот треба да биде преку употреба на вгнездени ресурси (`nested resources`). Соодветно, за секој коментар потребно е да се овозможат CRUD операции и нивна соодветна валидација (корисничкото име и коментарот се задолжителни). Коментарите треба да се листаат со помош на `id`-то.

2.5 Приказ на популарна вест

Потребно е во контролерот `news` да се овозможи метод за приказ на најдискутираните 2 вести (односно 2-те вести кои имаат најмногу `upvotes`). Пристапот треба да биде преку URL-то `/news/top`, најдискутираните вести се прикажуваат во листа.