Теория

1. Клиент-сервер

2. Серверные языки программирования

3. Клиентские языки программирования JavaScript

4. PHP

5. Сервер баз данных

6. Шаблоны проектирования

7. Архитектурные шаблоны проектирования

8. Шаблоны проектирования практических задач 22 Laravel

9. Простой

require

require\_once

include

include\_once

10. Шаблонная функция

11. Метод буферизации

ob\_start();

<div>test</div>

$content = ob\_get\_clean();

12. MVC

запрос -> маршрутизатор

13. Маршрутизация

14. Контроллер

15. Модель

16. View

17. Жизненный цикл запроса в MVC

18. HMVC

19. Фрэймворк Laravel

20. CRUD

21. Вставка

-множественная

Order::create($\_POST);

-одиночная

$obj = new Order;

$obj->body = $\_POST['body'];

$obj->user\_id = (int)$user\_id;

...

$obj->save();

redirect()->back();

22. Чтение

- объект

->first()

User::find(12);

User::where('username', $name)->first();

- массива объектов

$admins = User::where('role\_id', 1)->get()

User::all()

$arr = User::where('role\_id', 1)->paginate(10);

@foreach($arr as $one)

{{$one->name}}<br />

@endforeach

{!!$arr->links();!!}

23. Обновление

-одиночная

$obj = Order::find($id);

$obj->body = $\_POST['body'];

$obj->user\_id = (int)$user\_id;

...

$obj->save();

-Множественное

User::where('username', $name)->update($\_POST);

24. Удаление

$obj = Order::find($id);

$obj->delete();

User::where('username', $name)->delete();

25. Связи моделей

26. belongsTo

Product Catalog

public function catalogs(){

return $this->belongsTo('App\Catalog', 'catalog\_id', 'id');

}

27. hasOne

public function products(){

return $this->hasOne('App\Product', 'catalog\_id', 'id');

}

28. hasMany

Catalog Product

public function products(){

return $this->hasMany('App\Product', 'catalog\_id', 'id');

}

29. hasMany(throught)

public function products(){

return $this->hasMany('App\Product', 'catalog\_id', 'id');

}

30. Инструментарий

31. Apache2

.htaccess

32. Менеджер зависимостей

composer self-update

composer init

composer create-project laravel/laravel --prefer-dist

composer install

composer update

33. Система контроля версий

- состояние файлов

новый

измененный

зафиксированный

git add \*

git commit -m "message"

34. Локальный репозиторий git

git init

git add \*

git config --global user.name "Alex"

git config --global user.email "mikhalkevich@ya.ru"

git commit -m "message"

git status

35. Удаленный репозиторий git

github.com

gitlab.com

bitbacket.org

36. Ajax

37. Типы запросов

38. GET

39. POST

40. Обзор рынка IT

Практика

1. Версия для печати

2. Перевод на иностранный языки

3. Рекламные виджеты

4. Отправка оповещения на email

5. Админка

6. Конфигурирование config/site.php

<?

return [

'title'=>'Default TITLE for site'

];

config('site.name')

title

description

keywords

name

7. Миграции

8. Поиск

9. Хлебные крошки

10. Обработка формы с выводом ошибок на экран

11. Vip-товар

12. Vip-каталог

13. Модель товар

14. Модель каталог

15. Определить самый популярный товар

16. Шаблон практического решения

17. Безопасность web-приложения

dDOS

php-include -

sql-injection

xss - добавление скриптов через форму