**Тест 5 — Модульный тест**



**Тестовый контроль «Модульный тест»**

**Вопросы с выбором**

**1. Что такое Laravel?**

a. A) Библиотека для работы с API

b. B) Фреймворк для разработки на PHP

c. C) Язык программирования

d. D) CMS-система

**Ответ:** b. B) Фреймворк для разработки на PHP

**2. Какая команда используется для создания нового проекта Laravel?**

a. A) php artisan create

b. B) composer create-project

c. C) laravel new

d. D) php artisan new

**Ответ:** c. C) laravel new

**3. Какой файл конфигурации используется для настройки базы данных в Laravel?**

a. A) config/database.php

b. B) database/config.php

c. C) app/database.php

d. D) config/app.php

**Ответ:** a. A) config/database.php

**4. Какой командой можно запустить сервер Laravel на локальной машине?**

a. A) php artisan serve

b. B) laravel serve

c. C) php artisan run

d. D) serve

**Ответ:** a. A) php artisan serve

**5. Что такое Artisan в Laravel?**

a. A) Командный интерфейс для работы с базой данных

b. B) Веб-сервер

c. C) Инструмент для автоматизации задач

d. D) Среда для тестирования

**Ответ:** c. C) Инструмент для автоматизации задач

**6. Какая директива Blade используется для вставки данных в представление?**

a. A) {{ }}

b. B) @include

c. C) {{ include }}

d. D) {!!!!}

**Ответ:** a. A) {{ }}

**7. Что такое миграции в Laravel?**

a. A) Скрипты для работы с пользовательским интерфейсом

b. B) Инструменты для работы с базой данных и создания таблиц

c. C) Команды для обновления фреймворка

d. D) Методы для оптимизации кода

**Ответ:** b. B) Инструменты для работы с базой данных и создания таблиц

**8. Какой метод используется для возврата JSON-ответа в контроллере Laravel?**

a. A) json()

b. B) response()->json()

c. C) jsonResponse()

d. D) jsonResponse()->send()

**Ответ:** b. B) response()->json()

**9. Какую команду использовать для выполнения миграций в Laravel?**

a. A) php artisan migrate

b. B) php artisan make:migration

c. C) php artisan db:migrate

d. D) php artisan db:make-migration

**Ответ:** a. A) php artisan migrate

**10. Что такое маршруты (routes) в Laravel?**

a. A) Это статические пути для доступа к файлам

b. B) Это объекты для работы с базой данных

c. C) Это пути, которые определяют, как обрабатываются HTTP-запросы

d. D) Это таблицы базы данных

**Ответ:** c. C) Это пути, которые определяют, как обрабатываются HTTP-запросы

**11. Какой шаблон используется в Laravel для создания представлений?**

a. A) Blade

b. B) Twig

c. C) PHP

d. D) Smarty

**Ответ:** a. A) Blade

**12. Какой метод используется для создания нового ресурса в Laravel?**

a. A) php artisan make:model

b. B) php artisan make:controller

c. C) php artisan make:resource

d. D) php artisan make:entity

**Ответ:** c. C) php artisan make:resource

**13. Как установить Laravel через Composer?**

a. A) composer install

b. B) composer create-project --prefer-dist laravel/laravel

c. C) composer laravel

d. D) composer setup laravel

**Ответ:** b. B) composer create-project --prefer-dist laravel/laravel

**14. Что такое Eloquent в Laravel?**

a. A) Это система шаблонов для представлений

b. B) Это библиотека для работы с базой данных

c. C) Это фреймворк для работы с формами

d. D) Это система для работы с файлами

**Ответ:** b. B) Это библиотека для работы с базой данных

**15. Какой метод используется для загрузки аутентифицированного пользователя в Laravel?**

a. A) Auth::user()

b. B) User::auth()

c. C) Auth::login()

d. D) Auth::check()

**Ответ:** a. A) Auth::user()

**16. Что такое Vue.js?**

a. A) Фреймворк для разработки на PHP

b. B) Библиотека для создания пользовательских интерфейсов

c. C) CMS-система

d. D) Язык программирования

**Ответ:** b. B) Библиотека для создания пользовательских интерфейсов

**17. Какой метод используется для создания нового компонента в Vue.js?**

a. A) Vue.component()

b. B) Vue.createComponent()

c. C) Vue.newComponent()

d. D) Vue.addComponent()

**Ответ:** a. A) Vue.component()

**18. Какой метод используется для передачи данных от родителя к дочернему компоненту в Vue.js?**

a. A) props

b. B) data

c. C) methods

d. D) computed

**Ответ:** a. A) props

**19. Что такое REST API?**

a. A) Протокол для передачи данных по сети

b. B) Архитектурный стиль для создания веб-сервисов

c. C) Язык программирования

d. D) Фреймворк для разработки на PHP

**Ответ:** b. B) Архитектурный стиль для создания веб-сервисов

**20. Какой HTTP метод используется для создания нового ресурса в REST API?**

a. A) GET

b. B) POST

c. C) PUT

d. D) DELETE

**Ответ:** b. B) POST

**21. Что такое PWA (Progressive Web App)?**

a. A) Фреймворк для разработки на PHP

b. B) Веб-приложение, которое использует современные веб-технологии для предоставления пользовательского опыта, аналогичного нативным приложениям

c. C) CMS-система

d. D) Язык программирования

**Ответ:** b. B) Веб-приложение, которое использует современные веб-технологии для предоставления пользовательского опыта, аналогичного нативным приложениям

**22. Какой файл используется для настройки сервис-воркера в PWA?**

a. A) service-worker.js

b. B) app.js

c. C) manifest.json

d. D) index.html

**Ответ:** a. A) service-worker.js

**23. Что такое WordPress?**

a. A) Фреймворк для разработки на PHP

b. B) CMS-система

c. C) Язык программирования

d. D) Библиотека для создания пользовательских интерфейсов

**Ответ:** b. B) CMS-система

**24. Какой файл используется для настройки плагинов в WordPress?**

a. A) functions.php

b. B) plugins.php

c. C) wp-config.php

d. D) themes.php

**Ответ:** a. A) functions.php

**25. Какой метод используется для регистрации нового плагина в WordPress?**

a. A) register\_plugin()

b. B) add\_action()

c. C) wp\_register\_plugin()

d. D) register\_activation\_hook()

**Ответ:** d. D) register\_activation\_hook()

**26. Что такое Composer?**

a. A) Фреймворк для разработки на PHP

b. B) Менеджер зависимостей для PHP

c. C) CMS-система

d. D) Язык программирования

**Ответ:** b. B) Менеджер зависимостей для PHP

**27. Какая команда используется для установки пакета с помощью Composer?**

a. A) composer install package

b. B) composer require package

c. C) composer add package

d. D) composer setup package

**Ответ:** b. B) composer require package

**28. Какой файл используется для управления зависимостями в Composer?**

a. A) composer.json

b. B) package.json

c. C) dependencies.json

d. D) config.json

**Ответ:** a. A) composer.json

**Открытые вопросы**

**29. Объясните, что такое middleware в Laravel и как его можно использовать.**

**Открытый ответ:**

Middleware в Laravel — это механизм, который позволяет фильтровать HTTP-запросы, выполняемые вашим приложением. Middleware может использоваться для выполнения различных задач, таких как аутентификация, логирование, проверка прав доступа и т.д.

**Пример использования middleware:**

// Создание middleware

php artisan make:middleware CheckAge

// Регистрация middleware в ядре приложения (app/Http/Kernel.php)

protected $routeMiddleware = [

'check.age' => \App\Http\Middleware\CheckAge::class,

];

// Применение middleware к маршруту

Route::get('user/profile', function () {

//

})->middleware('check.age');

**30. Какая разница между GET и POST запросами в контексте маршрутов Laravel?**

**Открытый ответ:**

* GET-запросы используются для получения данных с сервера. Они кэшируются, могут быть закладками и сохраняются в истории браузера. Данные передаются через URL.
* POST-запросы используются для отправки данных на сервер. Они не кэшируются, не могут быть закладками и не сохраняются в истории браузера. Данные передаются в теле запроса.

**Пример маршрутов:**

Route::get('/user', 'UserController@index');

Route::post('/user', 'UserController@store');

**31. Что такое контроллеры в Laravel и для чего они используются? Приведите пример.**

**Открытый ответ:**

Контроллеры в Laravel используются для группировки логики обработки запросов. Они помогают организовать код и сделать его более читаемым и поддерживаемым.

**Пример контроллера:**

// Создание контроллера

php artisan make:controller UserController

// Пример метода в контроллере

public function index()

{

$users = User::all();

return view('users.index', compact('users'));

}

**32. Как в Laravel настроить связь между моделями "один ко многим"? Приведите пример.**

**Открытый ответ:**

Связь "один ко многим" устанавливается, когда одна модель имеет множество связанных моделей.

**Пример:**

// Модель Post

class Post extends Model

{

public function comments()

{

return $this->hasMany(Comment::class);

}

}

// Модель Comment

class Comment extends Model

{

public function post()

{

return $this->belongsTo(Post::class);

}

}

**33. Что такое зависимость инъекций (Dependency Injection) в Laravel? Приведите пример, как это используется в контроллерах.**

**Открытый ответ:**

Dependency Injection — это паттерн, который позволяет внедрять зависимости в класс через конструктор или методы. В Laravel это используется для упрощения тестирования и управления зависимостями.

**Пример:**

// Внедрение зависимости в контроллер

public function \_\_construct(UserRepository $users)

{

$this->users = $users;

}

**34. Как создать кастомную миграцию в Laravel? Опишите процесс создания и применения миграций.**

**Открытый ответ:**

Миграции используются для управления структурой базы данных.

**Пример создания и применения миграции:**

// Создание миграции

php artisan make:migration create\_users\_table

// Пример миграции

public function up()

{

Schema::create('users', function (Blueprint $table) {

$table->id();

$table->string('name');

$table->string('email')->unique();

$table->timestamps();

});

}

// Применение миграции

php artisan migrate

**35. Объясните, как можно валидировать данные в Laravel при помощи встроенных правил валидации. Приведите пример.**

**Открытый ответ:**

Валидация данных в Laravel может быть выполнена с помощью встроенных правил валидации.

**Пример:**

public function store(Request $request)

{

$validated = $request->validate([

'name' => 'required|max:255',

'email' => 'required|email|unique:users',

]);

// Сохранение данных

}

**36. Как работают события и слушатели (Events & Listeners) в Laravel? Опишите, как их можно использовать для обработки асинхронных задач.**

**Открытый ответ:**

События и слушатели позволяют реализовать асинхронную обработку задач.

**Пример:**

// Создание события

php artisan make:event UserRegistered

// Создание слушателя

php artisan make:listener SendWelcomeEmail --event=UserRegistered

// Регистрация слушателя в EventServiceProvider

protected $listen = [

'App\Events\UserRegistered' => [

'App\Listeners\SendWelcomeEmail',

],

];

**37. Как в Laravel можно реализовать аутентификацию пользователей? Опишите основные шаги.**

**Открытый ответ:**

Аутентификация пользователей в Laravel может быть реализована с помощью встроенных механизмов.

**Основные шаги:**

1. Установка пакета аутентификации: composer require laravel/ui
2. Генерация аутентификационных маршрутов и представлений: php artisan ui vue --auth
3. Настройка базы данных и миграций.
4. Регистрация маршрутов аутентификации в web.php.

**38. Как можно использовать пагинацию для вывода результатов в Laravel? Приведите пример использования пагинации в Eloquent.**

**Открытый ответ:**

Пагинация позволяет разбивать большие наборы данных на страницы.

**Пример использования пагинации в Eloquent:**

// Получение данных с пагинацией

$users = User::paginate(15);

// Отображение данных в представлении

return view('users.index', compact('users'));

// В представлении

{{ $users->links() }}

**Вопросы на сопоставление**

**39. Сопоставьте компоненты Laravel с их назначением.**

**Термины:**

1. Middleware
2. Eloquent ORM
3. Blade
4. Миграции
5. Контроллеры

**Назначения:**

A) Управляет взаимодействием с базой данных и предоставляет активные модели для работы с таблицами. B) Обрабатывает HTTP-запросы, выполняя дополнительные действия до или после основного запроса. C) Шаблонный движок, используемый для генерации HTML в представлениях. D) Позволяет управлять схемой базы данных, создавая или обновляя таблицы через версии. E) Управляют бизнес-логикой приложения и обрабатывают запросы, возвращая ответы.

**Ответ:**

1. Middleware — B) Обрабатывает HTTP-запросы, выполняя дополнительные действия до или после основного запроса.
2. Eloquent ORM — A) Управляет взаимодействием с базой данных и предоставляет активные модели для работы с таблицами.
3. Blade — C) Шаблонный движок, используемый для генерации HTML в представлениях.
4. Миграции — D) Позволяет управлять схемой базы данных, создавая или обновляя таблицы через версии.
5. Контроллеры — E) Управляют бизнес-логикой приложения и обрабатывают запросы, возвращая ответы.

**40. Сопоставьте компоненты Laravel с их функциональностью.**

**Термины:**

1. Artisan
2. Service Providers
3. Routes
4. Eloquent Relationships
5. Validation

**Функциональность:**

A) Обеспечивает механизм для управления различными действиями в приложении через командный интерфейс. B) Определяет, какие URL-адреса обрабатывает ваше приложение и какой контроллер или метод будет вызван для обработки запроса. C) Механизм для работы с базой данных, позволяющий строить связи между моделями, такими как "один к одному", "один ко многим" и другие. D) Отвечает за выполнение логики проверки данных, например, проверку email на правильность или наличие обязательных полей. E) Это классы, которые помогают интегрировать и настраивать различные компоненты приложения и сторонние пакеты.

**Ответ:**

1. Artisan — A) Обеспечивает механизм для управления различными действиями в приложении через командный интерфейс.
2. Service Providers — E) Это классы, которые помогают интегрировать и настраивать различные компоненты приложения и сторонние пакеты.
3. Routes — B) Определяет, какие URL-адреса обрабатывает ваше приложение и какой контроллер или метод будет вызван для обработки запроса.
4. Eloquent Relationships — C) Механизм для работы с базой данных, позволяющий строить связи между моделями, такими как "один к одному", "один ко многим" и другие.
5. Validation — D) Отвечает за выполнение логики проверки данных, например, проверку email на правильность или наличие обязательных полей.