# What's New in Laravel 5.5

# /

# И снова здравствуйте! В настоящее время дата выхода запланирована на август, мы на пороге версии Laravel 5.5. Имея это в виду, давайте начнем с новых функций и дополнений к фреймворку. Как всегда, их довольно много!

# /

# [What's New in Laravel 5.5:](https://laracasts.com/series/whats-new-in-laravel-5-5)Laravel 5.5 Installation and Whoops

**EPISODE 1 • RUN TIME 1:19**

# Пришло время посмотреть что нового в laravel версии 5 точка 5/в настоящий момент это пока не официально доступно поэтому подтянем dev релиз/окей/готово и теперь я хочу открыть это в sublime/первое что мы рассмотрим это whoops интеграцию/я открою это представление в браузере и сразу же вы видите то что никогда не видели/как оказывается laravel раньше использовал этот пакет который называется whoops/но потом они от него избавились и в этой версии они его снова вернули/и это на самом деле очень чистый и понятный способ для отображения ваших ошибок и вы даже можете кликать тут что очень здорово/в это случае laravel cообщает нам что для нашего приложения не определен ключ шифрования/поэтому я сейчас это сделаю/и если вернусь обратно и обновлю мы увидим привычную страницу/так что да это очень полезно/и например если мы сделаем тут опечатку и попробуем запутсить это в браузере то сразу же увидим ошибку и конкретно в какой строке/и это то что он делает/тут больше на самом деле вам нечего запоминать /в этом эпизоде вам достался легкий урок но в следующем мы перейдем к чему то другому/

# /

Давайте начнем с  [установки](https://laravel.com/docs/5.4#installing-laravel) dev версии Laravel. На момент этой записи только так мы можем получить версию 5.5 этого фреймворка. После установки, мы также быстро взглянем на новую интеграцию [Whoops](https://github.com/filp/whoops).

# /

# [What's New in Laravel 5.5:](https://laracasts.com/series/whats-new-in-laravel-5-5)Fresh Migrations

**EPISODE 3 • RUN TIME 1:04**

# Если мы запустим php artisan и пространство имен migrate обновилось и теперь у нас тутт есть migrate fresh и это схоже с migrate refresh но тут есть небольшая разница/давайте я просто покажу это/у меня есть новый проект и я пишу php artisan migrate и это создаст эти таблицы по умолчанию и если я напишу php artisan migrate refresh это откатит все мои таблицы и заново перезальет их/однако если мы напишем migrate fresh это сделает тоже самое но вместо того чтобы отктывать таблицы это просто удалит их и создаст зановго/и в действительности если вы задумаетесь это имено та команда которой не хватало/в особенности когда на продолжительных проектах вы могли попадать в ситуации когда вы пытались откатывать но там есть какой то баг или какое то сделанное изменение которое вызывает ошибку и вы не можете вернуться к началу так что это отличный способ сказать – сбрось это все я собираюсь начать заново и затем запусти все миграции которые у меня есть/

# /

Новая Artisan команда migrate:fresh поставляется как часть laravel 5.5. Это похоже на существующий вариант refresh, однако, вместо того, чтобы откатывать все ваши миграции (и инициировать соответствующий метод downдля каждого), эта новая команда просто “дропнет“ все таблицы и выполнит ваши миграции с нуля.

# /

# [What's New in Laravel 5.5:](https://laracasts.com/series/whats-new-in-laravel-5-5)Automatic Package Discovery

**EPISODE 5 • RUN TIME 2:51**

# laracast

# [What's New in Laravel 5.5:](https://laracasts.com/series/whats-new-in-laravel-5-5)Model Factory Generation

**EPISODE 9 • RUN TIME 1:51**

# Еще одно новшество в этой версии laravel это Model Factory Generation так что если мы просто взглянем на это/тут есть обновление make factory/так вот в чем разница в рпошлом в вашей базе данных в папке factories у вас был бы единственный factory файл и это было бы домом для всех ваших factory/елси вы не знакомы с этим то это что то вроде чертежа для ваших моделей/поэтому вы могли бы наполнить их какими то данными и потом быстро использовать для создания пользователей для ваших тестов на локальной среде разработке/и да этот единственный файл мог стать достаточно большим и запутанным/но вот теперь он может быть разделен на несколько в соответствии с вашими моделями/давайте посмотрим/php artisan make factory/и я все ее работаю с концептом видео/поэтому дадим ему такое название/я только что создал factory для модели видео/и если мы переключимся сюда/ вот наш новый factory файл/и вот тут есть место где мы могли бы начать/здесь я мог бы написать заголовок/долеж быть faker sentence/и архивирован – ожидаем логическое значение/и это все/мы закончили/так что если мы запустим tinker/z мог бы сейчас сказать/я бы хотел factory и в особенности 10 таких/create и вот они созданы/сейчас если бы я сказал – покажи мне все записи – разумеется мы получим 10 рандомных из нашей базы данных/и да теперь в версии 5 и 5 мы разделили эти фактори на 2 и теперь у нас есть этот генератор для быстрого наполнения/

# Удалить 146-148

# Так что теперь это выглядит намного чище /

# /

# В Laravel 5.4, все factory моделей содержались в одном файле. Нужно создать новую factory? Ну, вы знаете способ: скопируйте и вставьте тот, который перед ним, и внесите свои изменения. К счастью, в 5.5, есть более простой способ. Каждая из ваших Eloquent моделей теперь будет иметь соответствующий файл factory модели.

# /

# [What's New in Laravel 5.5:](https://laracasts.com/series/whats-new-in-laravel-5-5)New Blade Directives

**EPISODE 15 • RUN TIME 2:31**

Несколько эпизодов назад мы рассматривали кастомные blade if условия/ и мы уазнали что в в вашем appservice provider вы можете сделать что то подобное blade if и далее что угодно/ если вы авторизовались или если подписались/что угодно что может вернуть boolean / может быть вы хотите проверить среду/чтож мы могли бы определить это так и далее return проверить является ли среда тем что вы передали как входной параметр/и это все/и теперь это просто чтобы освежить память/

**Удалить 034 – 36**

В welcome представлении я мог бы написать/я собираюсь использовать мою новую директиву blade/скажем если среда равна local/отобразить - вы находитесь на локальной среде/и вот теперь если мы посмотрим в браузере это показывать нам что мы на локальной среде/но если мы перейдум в настройки и изменить на что то вроде продуктивная/конечно же мы ничего тут не увидим/так что это прост небольшой обзор/но вот что я еще знаю в ларавел что тут есть пара хелпер функций который делают это в том же формате/так если перейдем обратно и я просто удалю это и теперь мы в welocome представлении и как вы помните несколько эпизодов назад мы говорили оговорили о некоторых кастомных blade eсловиях/таких как guest/так что елси пользователь зашел как гость без авторизации это будет работать/но как оказалось/на момент записи того видео фреймворк добавил поддержку этой функции так что тепреь если я напишу тут endguest/это будет рабоать без каких либо работа с нашей стороны/это доступно прямо в коробке/напишем вы – гость/и если обновимся тут как оказалось мы гости)/мы можем сделать тоже самое для авторизации/и знаете раньше нам бы пришлось писать if check затем отработать/и нам пришлось бы делать это снова и снова но теперьэто было укорочено/и сейчас давайте попробуем/else вы гость/или в этом случае участник/окей обновим тут и мы все еще гости/сейчас давайте авторизируем пользователя/напишем login и мне нужен пользоватлеь но у меня его тут нет/давайте просто сделаем следующее app user create/мы тут просто сгенерировали пользователя и затем авторизовали/и теперь если мы вернемся обратно и обновимся – конечно же мы сейчас участнк/и это все-это все что вам надо знать в рамках этого видео – 2 helper директивы были добавлены одна для авторизации другая для гостя

/

Laravel предлагает две новые полезные директивы Blade в составе версии 5.5: @auth и @guest. Они делают именно то, что вы могли подумать, поэтому попрощайтесь с константой @if (auth()->check()).

/

# [What's New in Laravel 5.5:](https://laracasts.com/series/whats-new-in-laravel-5-5)Route Helpers

**EPISODE 16 • RUN TIME 2:08**

Окей следующий шаг – у нас есть несколько особенных helpers для маршрутов/давайте переключимся в сублайм и мы собираемся разобрать маршрут для редиректа и для представления/и вот что мы сделаем/я забегу немного вперед и запущу команду создать авторизацию и если вы не очень знакомы с этой командой – это автоматически подготовит уровень аутентификации – форму регистрации и форму входа/и как вы видите в коем случае я уже авторизовался но конечно если вы хотите войти вы увидите тут ссылку войти и зарегистрироваться/окей в любом случае это ваш маршрут домашней страницы но иногда вы хотите перенаправить один маршрут на другой/например если мы переключимся обратно вот команда которую я добавил/возможно/я хотел бы что если пользователь посещает старницу слеш me это перенаправит его на дашборд/и традиционно вы бы написали/чтож когда пользователь посещает страницу слеш me мы перенаправим его на слеш home/довольно обычное дело/сейчас если мы вернемся обратно и обновимся/я посещу me как это видно мы были перенаправлены на home страницу/окей/но теперь в ларавел 5 и 5 это было немного упрощено/заменим это на redirect со страницы me на home/стало проще/проверим/посещаем me перешли на home/получили тот же результат но подход гораздо проще/и вот это был route redirect/тот же метод был применен для отображения представлений/если пользователь посещает определенную страницу – нужно отобразить ему представление/напишем тут view так что когда я посещаю домашнюю страницу я хочу отобразить welcome view и опять я избавлюсь от этого/окей вернемся обратно на домашнюю страницу и мы видим тоже самое как было раньше но теперь это выглядит гораздо чище и понятней/и это все что вам надо знать/так что если вы используете ларавел 5 5 теперь вы можете применить это в ваших проектах

/

Теперь вы можете использовать полезные Route::view() и Route::redirect() helper функции в рамках вашего routes.php файла. Подумайте об этом как о сокращении для загрузки представления и выполнения перенаправления, соответственно.

/