# Heroku хостинг позволяет установить сайт бесплатно

<p>Heroku - это облачная платформа, которая позволяет создавать, разрабатывать, устанавливать, запускать, контролировать и масштабировать сайты. Позиционирует себя как самый быстрый способ перейти от идеи к URL-адресу с приложением, избегая головных болей от работы с инфраструктурой. Благодаря сервису пользователи тратят время и силы только на разработку и развертывание проектов, которые сразу начинают приносить прибыль. </p>

<p>Как облачный хостинг Хероку появился в июне 2007 года, тогда он поддерживал только один язык программирования – Ruby. Сегодня Хероку дополнительно поддерживает Nod.js, Java, PHP, Python, Go, Scala, Clojure. </p>

## <h2>Dynos - сердце Heroku хостинга</h2>

<p>Хероку платформа использует модель контейнера для запуска и масштабирования приложений. </p>

<blockquote>

<p>Система называется масштабируемой, если она способна увеличивать производительность пропорционально дополнительным ресурсам. Масштабируемость можно оценить через отношение прироста производительности системы к приросту используемых ресурсов. Чем ближе это отношение к единице, тем лучше. </p>

<cite><a href="https://ru.wikipedia.org/wiki/Масштабируемость">Википедия</a></cite>

</blockquote>

<p>Контейнеры, используемые в работе, называются «dynos». Это изолированные виртуальные контейнеры Linux, предназначенные для выполнения кода на основе заданной пользователем команды. Приложение может масштабироваться до любого количества контейнеров на основе требований к ресурсам. Возможность управления контейнерами предоставляют Heroku простой способ масштабировать, управлять количеством, размером и типом dynos, которые могут понадобиться проекту в любой момент времени. </p>

<figure">

<img src=" images\container\_model.png " alt="Dynos - строительные блоки, которые управляют приложением Heroku" title="Heroku управляет Dynos для поддержания функциональности всех программ и всей системы">

<figcaption>Принцип работы строительных блоков – dynos </figcaption>

</figure>

## <h2>Как установить сайт бесплатно</h2>

<p>Хостинг позволяет бесплатно разместить до 5 проектов. Сначала надо зарегистрироваться, получив бесплатную учетную запись. </p>

### <h3>Учет работы Dyno</h3>

<p>Учетная запись предоставляет 550 часов в месяц бесплатного использования Dyno. В дополнение к этим базовым часам счета, можно получить дополнительно 450 часов ежемесячной бесплатной работы приложений. Для этого потребуется зарегистрировать платёжную карту. Деньги взиматься не будут, просто надо ее зарегистрировать. Кстати, регистрация карты предоставляет и другие опции. </p>

<p>Халявные dynos обладают уникальным свойством – они умеют ложиться спать после 30-минутного бездействия сайта. Таким образом, расход часов прекращается. Условия бездействия: </p>

<ul>

<li>отсутствие трафика, т.е. веб dynos не действуют; </li>

<li>рабочие dynos не работают. </li>

</ul>

<p>Необходимо выполнение обоих условий. </p>

<p>При появлении трафика dynos просыпаются с небольшой малозаметной задержкой. </p>

<p>После израсходования всех часов в течение месяца, приложения засыпают и спят оставшуюся часть месяца. </p>

### <h3>Как определить остаток бесплатной работы Dyno</h3>

<p>Посмотреть оставшиеся свободные часы можно используя Heroku CLI, набрать команду в командной строке </p>

<p class="main-text” >heroku ps -a <имя приложения>. </p>

<p>Или в аккаунте на странице управления счетами на вкладке Billing. </p>

<p>После расхода 80% запаса свободных часов придет уведомление на электронную почту. Второе уведомление придет после 100% расхода квоты времени, и приложения будут спать до конца текущего месяца. </p>

### <h3>Установка сайта</h3>

<p>Для развертывания приложений следует установить на локальную машину популярную систему контроля версиями Git, интерфейс командной строки Heroku (CLI). CLI - важная часть использования Хероку. Он упрощает создание и управление приложениями Героку непосредственно с терминала. </p>

<p>Чтобы установить сайт с кодом PHP дополнительно потребуется установка Composer (перевод с английского – Композитор). </p>

### <h4>Последовательность действий *для установки сайта*</h4>

<ol>

<li><p>Зайти в системную консоль, набрать команду</p>

<p class="main-text">heroku login</p>

<p>Затем ввести адрес электронной почты, пароль. </p> </li>

<li><p>Если проект не загружен на сервис GitHub, а находится на жестком диске компьютера, зайти в папку для него, или создать ее. </p></li>

<li><p>Создать локальный репозиторий, набрав</p>

<p class="main-text >git init </p></li>

<li><p>Загрузить в эту папку репозитория файлы сайта. </p>

<ol>

<li><p>Создать и загрузить файл <a href="# procfile ">Procfile </a>, другие необходимые для Хероку файлы, например <a href="# composer ">composer.json </a> при загрузке сайта PHP. </p></li>

<li><p>Если файл <span class="main-text">composer.json</span> не пустой, создать файл <span class="main-text">composer.lock</span>, выполнив команду</p>

<p class="main-text >composer update. </p></li>

</ol>

</li>

<li><p>Чтобы добавить и закрепить файлы в репозитории, выполнить последовательно действия: </p>

<p class="main-text >git add . </p>

<p class="main-text >git commit -am "<Комментарий>".</p></li>

<li><p>Создать приложение на Героку, которое готовит хостинг для получения исходного кода: </p>

<p class="main-text >heroku create</p></li>

<li><p>Развернуть свой код</p>

<p class="main-text >git push heroku master</p></li>

<li><p>Заключительное действие - открыть сайт в окне браузера</p>

<p class="main-text >heroku open. </p></li>

</ol>

<p>Всё, сайт успешно установлен на облачный хостинг! </p>

## <h2 id="procfile">Procfile</h2>

<p>Прокфайл определяет команды, которые выполняют приложения в контейнерах динос, объявляет типы процессов. Каждый дино-контейнер относится к одному из объявленных типов процессов и выполняет команду, связанную с этим типом процесса. </p>

<p>Procfile - это простой текстовый файл, который называется точно <span class="main-text">*Procfile*</span> (например, <span class="main-text">*Procfile.txt* </span> не действует). </p>

<p>Прокфайл должен находиться в корневом каталоге приложения. Он не работает, если находится в ином месте. </p>

<p>Procfile объявляет типы процессов в отдельных строках файла, каждая из которых имеет следующий формат: <тип процесса>: <команда>. Каждая новая декларация процесса начинается с новой строки. </p>

<p>Тип процесса <span class="main-text">*web*</span> является особым: это единственный тип процесса, который может получать внешний HTTP-трафик с маршрутизаторов Хероку. Если проект включает веб-сервер, необходимо объявить процесс как <span class="main-text">*web*</span>. </p>

## <h2 id="composer">Composer.json</h2>

<p>Composer.json располагается в корневом каталоге. При его наличии Хероку распознает приложение как PHP. Поддержка PHP будет применяться только тогда, когда проект с программой имеет файл с именем composer.json в корневом каталоге. </p>

<p>Хероку использует Композитор для управления зависимостями в проектах PHP. Если программа имеет какие-либо зависимости с другими внешними библиотеками, <span class="main-text">composer.json</span> файл должен указать их. </p>

<p>Даже если приложение не имеет зависимостей от Композитор, оно должно содержать в composer.json хотя бы пустое поле ({}), чтобы быть распознанным как приложение PHP. </p>

<p>Также этот файл инструктирует Хероку, какую версию PHP использовать. Если такую инструкцию не указать, то Heroku выберет самую последнюю возможную версию PHP 5. </p>

## <h2>Устранение проблем установки</h2>

<p>Размещая приложение на платформу, я обнаружил конфликт между локальным сервером OpenServer и Heroku. Хостинг для PHP проектов требует установить на компьютер средство по управлению зависимостями со сторонними библиотеками – Композитор. </p>

<p>Установил средство, и обнаружилась проблема. При проверке правильности установки командой в системной консоли <span class="main-text">*composer –V*</span> выдало: </p>

<!—noindex-->

<p> «"""" не является внутренней или внешней командой, исполняемой программой или пакетным файлом». </p>

<!—/noindex-->

<p>Проблему решило изменение в настройках OpenServer. Файл <span class="main-text">php.exe</span> при установке Композитор я загрузил в папку \OpenServer\modules\php\PHP-7. А в этой директории OpenServer-а имеются файлы <span class="main-text">composer.phar</span> и <span class="main-text">composer.bat</span> устаревшей версии, и они подхватываются в Композитор. Чтобы их не удалять, я их только переименовал, добавив «1». </p>

## <h2>Полезные команды для сайта на хостинге</h2>

<p>Чтобы попасть в корень проекта на хостинге команда: <span class="main-text">heroku run bash -a=<наименование проекта ></span>.</p>

<p>Удалить проект с хостинга команда: <span class="main-text">heroku apps:destroy --app <наименование проекта ></span>.</p>

## <h2>Вывод</h2>

<p>В этой статье изложено о возможности бесплатного развертывания на хостинге динамичного сайта на языке PHP. Размещение статичного проекта имеет особенности. Об установке простого статичного сайта <a href="//article.php?id=20">здесь </a>. </p>

<p>На основании вышесказанного следует вывод: облачный хостинг Героку – прекрасный ресурс для бесплатного размещения сайтов. </p>