Бесплатный хостинг на HEROKU для проекта на JAVA.

Сегодня мы поговорим о том, как выложить Java приложение на бесплатный хостинг и этим самим показать свое творение другим людям.

Я долго искал бесплатных хост-провайдеров с поддержкой Java, но не находил ничего, что могло бы подойти для моих требований: выложить приложение на Java как обычный сайт.

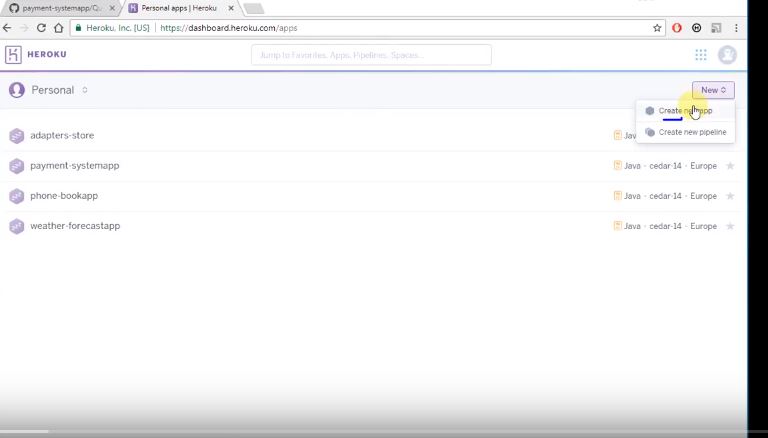
Сейчас в мире есть множество клауд платформ, которые можно использовать в качестве площадки для сайта на Java, но удовольствие это зачастую не из дешевых. Но, все же мне удалось найти клауд провайдера у которого есть бесплатный, естественно ограниченный тарифный план для приложений на джава.

[Heroku](https://www.heroku.com/) — облачная платформа, которая поддерживает очень много языков программирования в том числе и Java. Данная платформа имеет бесплатный тарифный план, который позволяет загружать не более 5 приложений на один аккаунт и после 30 минут не активности приложения — оно «засыпает». Для учебных целей этого плана более чем достаточно. Для коммерческих целей можно воспользоваться другими тарифными планами.

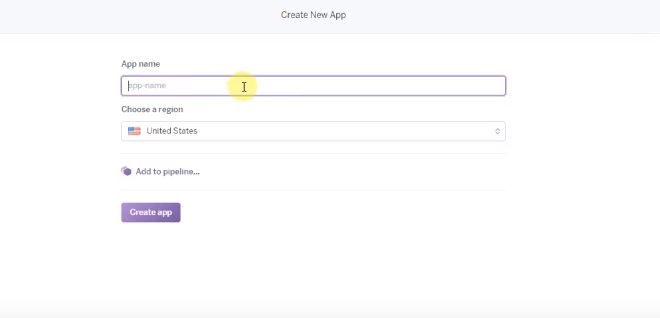
Именно на Heroku мы и попробуем выложить приложение на java. Для этих целей можно использовать приложение, которое мы писали в цикле статей: [Простое приложение на java](http://java-master.com/category/java-%D0%B4%D0%BB%D1%8F-%D0%BF%D1%80%D0%BE%D0%B4%D0%B2%D0%B8%D0%BD%D1%83%D1%82%D1%8B%D1%85/java-web/).

Первое, что нам нужно для того, чтобы задеплоить java приложение на бесплатный хостинг — это зарегистрироваться на heroku. Создать аккаунт очень просто: следуем инструкциям на сайте провайдера.

Далее нужно в личном кабинете создать новое приложение: New -> Create new app



После этого у Вас должно появиться окно для выбора названия приложения и региональности:

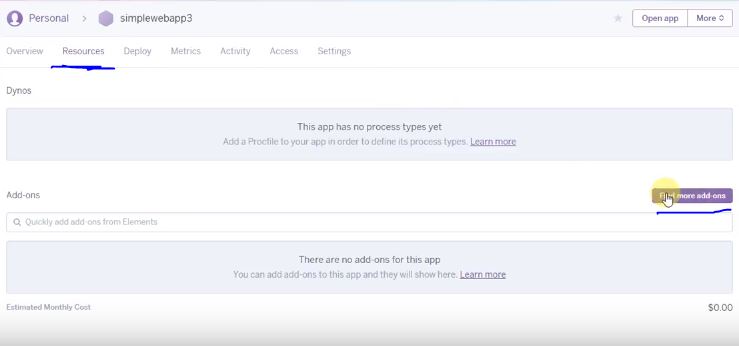


Имя можете задать любое. По поводу региона: лучше выбрать Европу — пинг будет меньший.

Теперь, если Ваше приложение использует базу данных ее нужно создать в хероку. Бесплатно клауд-провайдер поддерживает Postgres базу данных. Где-то была статья в Интернете, что можно бесплатно использовать любую базу данных, но автор не показал как, а мои попытки использовать другие базы не увенчались успехом. Поэтому, будем использовать Postgres.

Если у Вас приложение на другой базе данных например MySql — ничего страшного. Скрипты вставки, выборки, удаления и обновления данных идентичны. Нужно всего лишь немного изменить скрипт создания таблиц. А именно: если в MySql AUTOINCREMENT для автоматического увеличения целочисленный переменной, то в Postgres — SERIAL. Это все что Вам нужно переделать в скрипте создания таблиц.

Прежде всего, создадим базу на хероку. Для этого нужно нажать на вкладку Resources созданного приложения и нажать на кнопку Find more add ons:

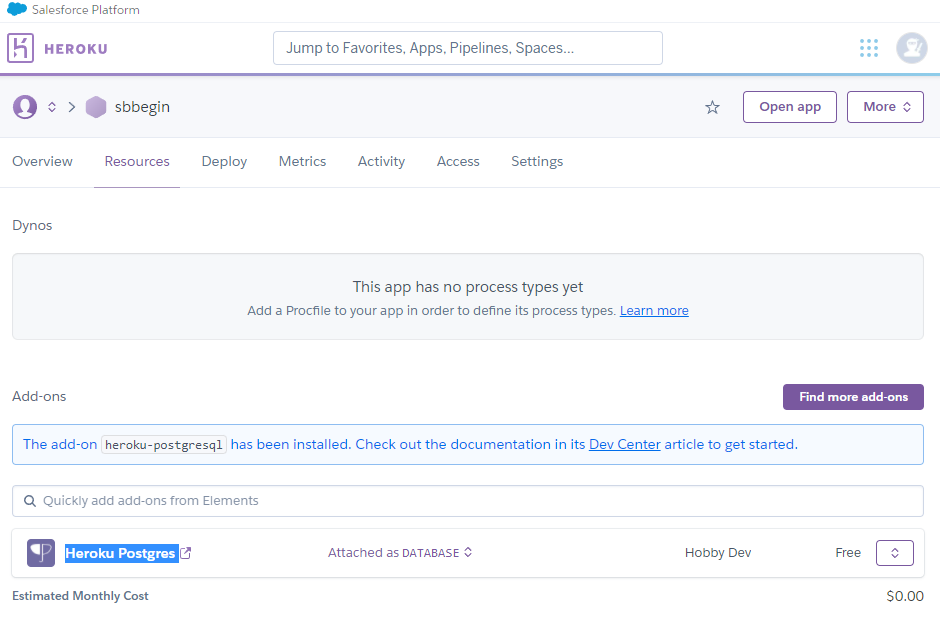


Далее откроется окно где будет множество приложений.

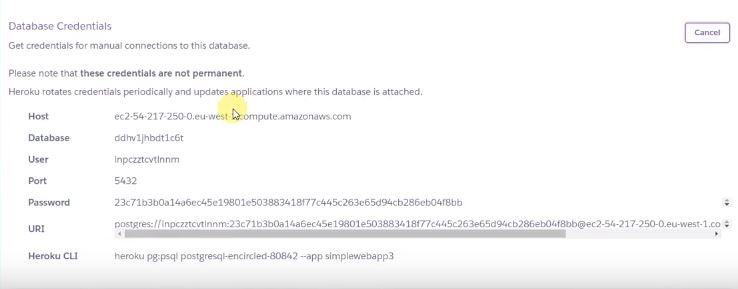
Вам нужно будет найти Postgres.

При нажатии на Heroku postgres откроется окно где будет кнопка install heroku postgres. Нажимаем ее, выбираем приложение, к которому мы хотим добавить базу данных и жмем Submit Order Form. База данных создана.

Для того чтобы посмотреть пользователя, пароль и имя базы данных нужно нажать на созданную базу данных,



выбрать вкладку Settings и нажать кнопку “view credentials”:



Если у Вас не установлена БД PostgreSQL, то вначале переключить проект на нее.

Далее для запуска на Heroku:

1. сработало закомментирование всех строк в "application.properties" (т.е. Heroku сам подключает наш проект к своей БД). Правда потом раскомментировал строку “*spring.jpa.hibernate.ddl-auto=update”* и нормально работало.
2. поменял в html-файлах строки загрузки моих js-файлов со скриптами:

с *<script type="text/javascript" src="https://localhost/js/admin.js"></script>* на *<script type="text/javascript" src="https://sbbegin.herokuapp.com/js/admin.js"></script>,* где “*sbbegin.herokuapp.com”* это адрес моего сайта, который можно посмотреть на закладке “setting” моего проекта на Heroku.

1. поменял в js-файлах строки обращения к файлам серверной части: с *url: "https://localhost:443/admin/findAll"* на *url: "https://sbbegin.herokuapp.com/admin/findAll",* при этом убрав порт (и здесь Heroku сам подключает нужный ему порт).
2. После четвертого удаления/изменения кода/загрузки нового кода на Heroku вылетела ошибка циклической ссылки - ошибка создания bean-компонента с именем «getPasswordEncoder», что такой бин уже создается. Помогло добавление аннотации @Lazy над бином getPasswordEncoder() в файле WebSecurityConfig.java добавил аннотацию @lazy.

В других статьях предлагают сделать следующие шаги для запуска проекта на Heroku (т.е. самостоятельно прописывать подключение к БД Heroku). Но у меня такое не сработало (ругалось на подключение к https и закомментирование строк с подключением к https в файле “application.properties” не помогало):

Далее переходим в наше приложение и ищем где мы подключаемся к базе.

вместо

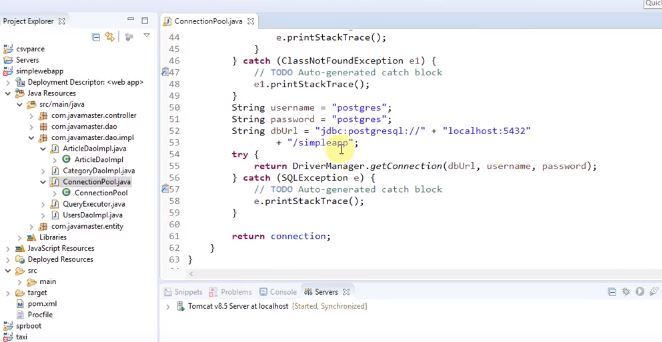
#spring.datasource.url=jdbc:postgresql://localhost:5432/forflyway?serverTimezone=UTC

#spring.datasource.username=root  
#spring.datasource.password=qwa12345

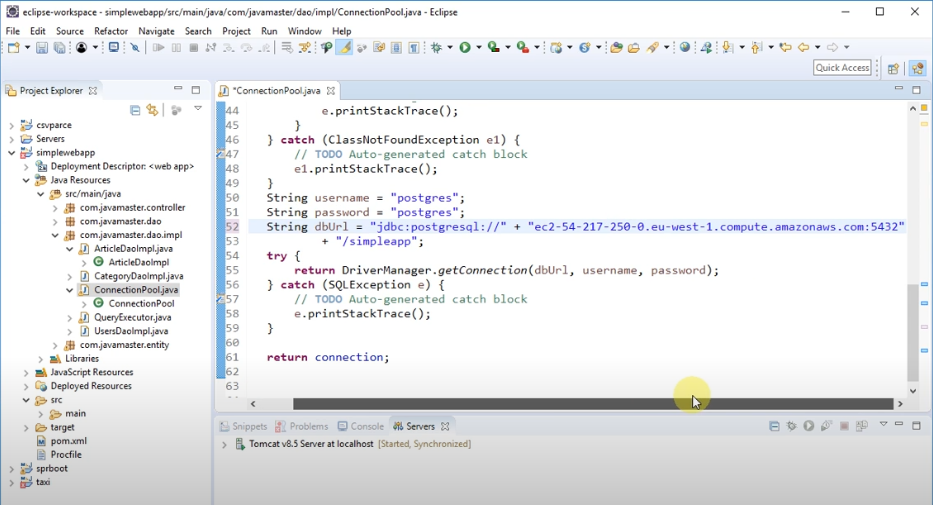
написать

spring.datasource.url=${SPRING\_DATASOURCE\_URL:jdbc:postgresql://localhost:5432/forflyway?serverTimezone=UTC}  
spring.datasource.username=${SPRING\_DATASOURCE\_USERNAME:root}  
spring.datasource.password=${SPRING\_DATASOURCE\_PASSWORD:qwa12345}

где SPRING\_DATASOURCE\_URL/ USERNAME/ PASSWORD – это служебные переменные в HEROKU, которые “автоматом подтягивают” данные о созданной у них БД. Хотя можно и вручную повставлять эти данные, взяв их из верхней таблички(и тогда не нужны ${}).

У автора этой статьи иначе устроено подключение к БД, поэтому он меняет по своему:

Просто замените пользователя, пароль и урл к базе на те, который Вам создал клауд.



Для того, чтобы подключиться к удаленной базе с локальной машины добавьте еще один параметр к урл: sslmode=require как показано на рисунке выше.

Далее нужно сделать изменения в нашем pom.xml. Так как мы используем постгрес — не забудьте добавить драйвер подключения именно для этой базы данных.

|  |  |
| --- | --- |
| [Код](https://java-master.com/java-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BD%D0%B0-%D0%B1%D0%B5%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D0%BD%D1%8B%D0%B9-%D1%85%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3/" \l "codesyntax_1" \o "Нажмите, чтобы показать или скрыть блок кода) |  |

**<dependency>**  
            **<groupId>**org.postgresql**</groupId>**  
            **<artifactId>**postgresql**</artifactId>**  
            **<version>**9.4.1208**</version>**  
            **</dependency>**

Далее нужно добавить плагин для хероку, который позволит без проблем запустить приложение.

|  |  |
| --- | --- |
| [Код](https://java-master.com/java-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BD%D0%B0-%D0%B1%D0%B5%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D0%BD%D1%8B%D0%B9-%D1%85%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3/" \l "codesyntax_2" \o "Нажмите, чтобы показать или скрыть блок кода) |  |

**<plugin>**  
                **<groupId>**org.apache.maven.plugins**</groupId>**  
                **<artifactId>**maven-dependency-plugin**</artifactId>**  
                **<version>**2.3**</version>**  
                **<executions>**  
                    **<execution>**  
                        **<phase>**package**</phase>**  
                        **<goals>**  
                            **<goal>**copy**</goal>**  
                        **</goals>**  
                        **<configuration>**  
                            **<artifactItems>**  
                                **<artifactItem>**  
                                    **<groupId>**com.github.jsimone**</groupId>**  
                                    **<artifactId>**webapp-runner**</artifactId>**  
                                    **<version>**8.0.30.2**</version>**  
                                    **<destFileName>**webapp-runner.jar**</destFileName>**  
                                **</artifactItem>**  
                            **</artifactItems>**  
                        **</configuration>**  
                    **</execution>**  
                **</executions>**  
            **</plugin>**

С pom.xml готово. Осталось добавить еще один файл настроек к нашему приложению: Procfile. Для этого в корне приложения добавьте файл с именем Procfile и без расширения(а когда спросит с каким форматом его ассоциировать, то из предложенного списка выбрать “**txt**”). В него положите следующее содержимое:

|  |  |
| --- | --- |
| [Код](https://java-master.com/java-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BD%D0%B0-%D0%B1%D0%B5%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D0%BD%D1%8B%D0%B9-%D1%85%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3/" \l "codesyntax_3" \o "Нажмите, чтобы показать или скрыть блок кода) |  |

web:    java $JAVA\_OPTS -jar target/dependency/webapp-runner.jar --port $PORT target*/\*.war*

Теперь наше приложение готово к отправке в cloud. Одним из преимуществ Heroku является поддержка деплоя приложений через github. Это очень удобно. Вам нужно всего лишь выложить приложение на github, «приконектить» гитхаб с хероку и нажать кнопку деплоя.

Сейчас я не буду расписывать, как выложить приложение на гитхаб. Вот видео с подробными инструкциями:

Когда Вы выложили приложение на гитхаб — нужно зайти в кабинет heroku, выбрать приложение и перейти на вкладку deploy. Выберите Github из списка deploy method. Далее нужно приконектить свой аккаунт, выбрать репозиторий с приложением и нажать кнопку deploy branch. Теперь Вам осталось подождать пока приложение задеплоиться и можно на него переходить.  Оно будет иметь имя***http://имя\_приложения.herokuapp.com/***

Вот и все. На первый взгляд может показаться не очень просто, но это того стоит. Вы сможете показать людям результаты Ваших трудов не в виде кода на гитхаб, а в виде приложения, просмотр которого вызывает больше удовольствия чем читать код.

Это не единственный способ отправки java приложения на бесплатный хостинг. Тем более есть и другие клауд-провайдеры. Amazon например предоставляет пользование своими сервисами бесплатно один год. Можно попробовать использовать Amazon. Я в свое время открыл для себя Heroku и пользуюсь им до сих пор.

Как всегда, перед статьей я записал видео с подробными инструкциями по отправке Java приложения на бесплатный хостинг Heroku.