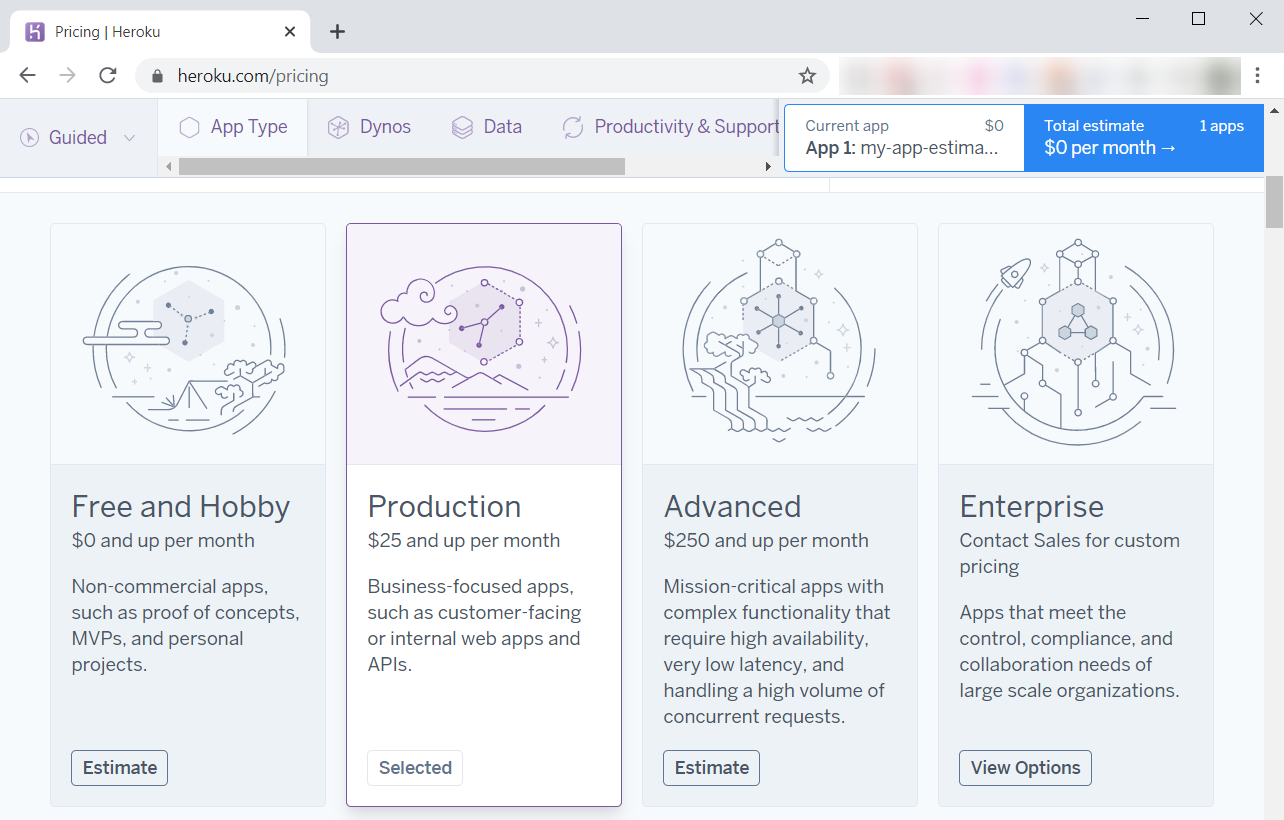
Лекция 22

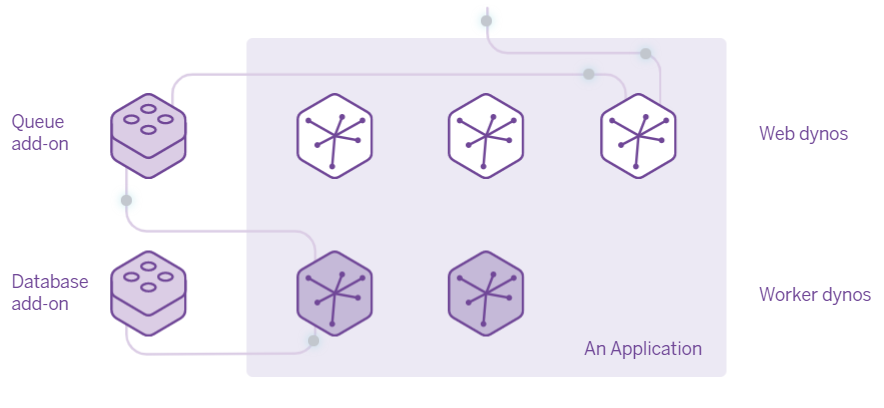
ПСКП, ПОИТ-3

**Heroku**

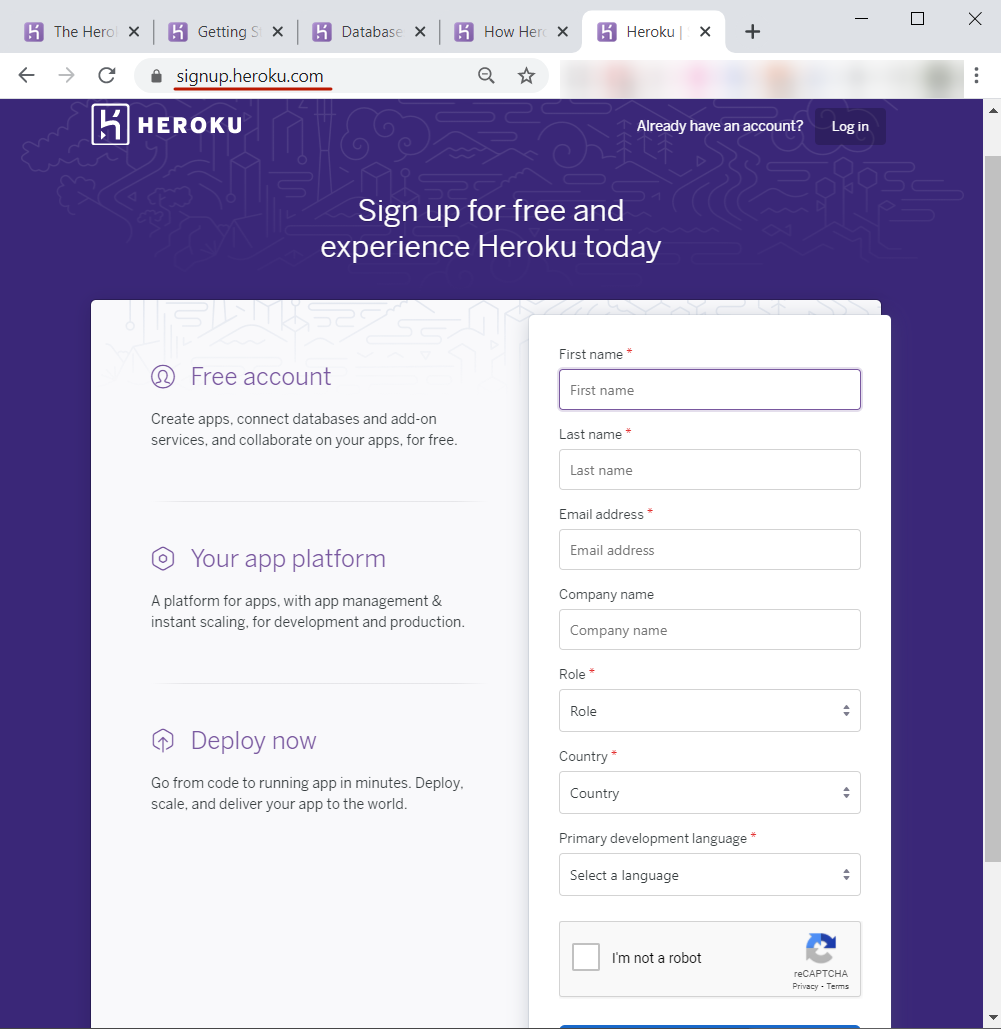
1. **Heroku:** облачный сервис: легкость управления и масштабирования, автодеплой с github.
2. **Heroku:** позволяет развернуть приложения на Node, Ruby, Java, PHP, Python, Go, Scala, Clojure.
3. **Heroku:** <https://devcenter.heroku.com/articles/getting-started-with-nodejs>
4. **Heroku:** <https://www.heroku.com/pricing>

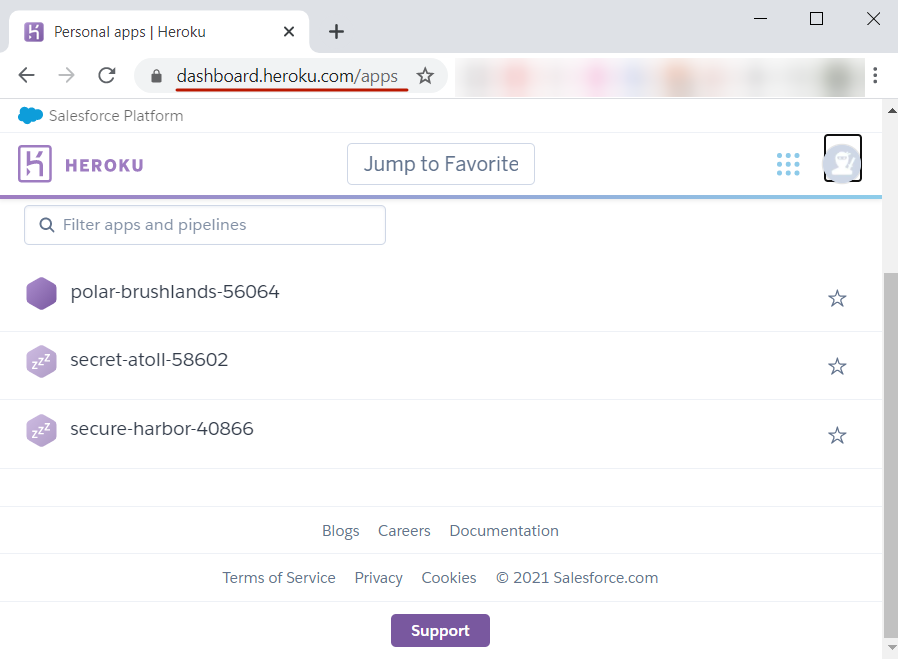


1. **Heroku:** контейнерная система, диносы.
2. **Heroku:** динос - изолированные виртуальные контейнеры Linux, предназначенные для выполнения кода на основе заданной пользователем команды.

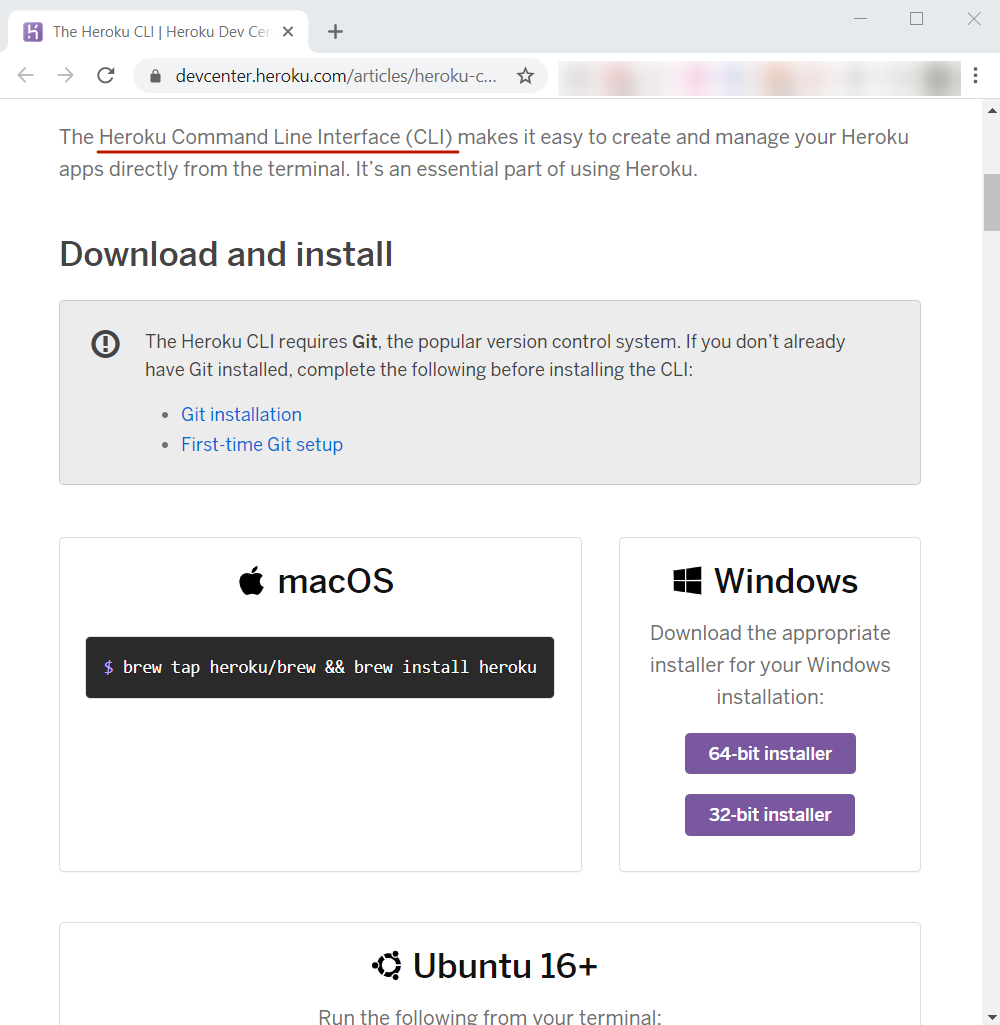


1. **Heroku:** add-ons – сторонние облачные сервисы, используемые для расширения приложений с помощью ряда функций, таких как хранилища данных (Heroku Postgres, Heroku Redis и Apache Kafka), протоколирование, мониторинг и другое.
2. **Heroku:** зарегистрироваться на <https://www.heroku.com/>



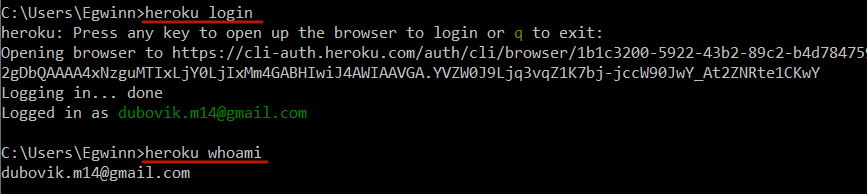


1. **Heroku:** скачать и установить Heroku CLI (предварительно должен быть установлен Git)

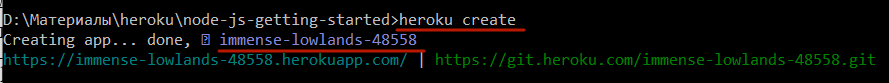


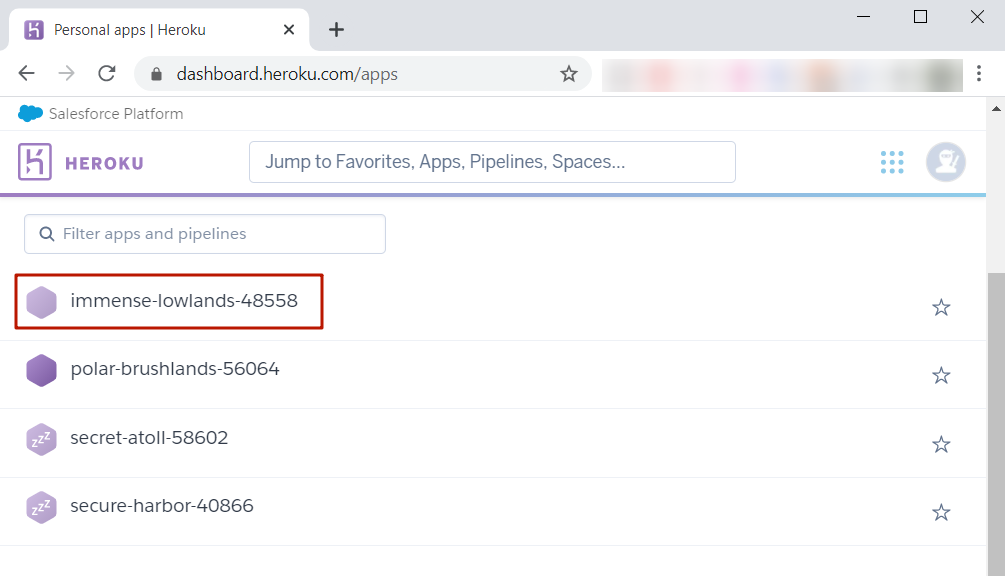


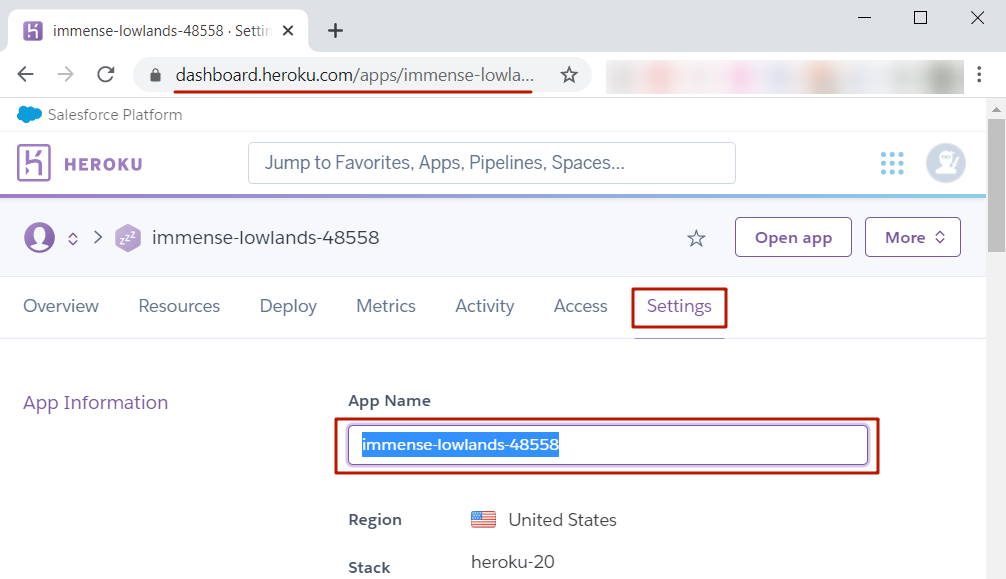
1. **Heroku:** залогиниться, heroku login



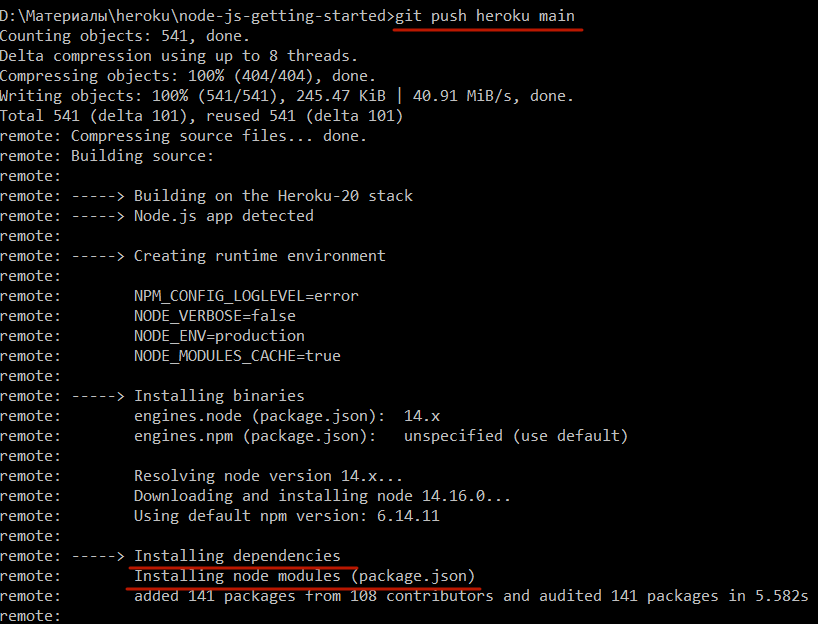
1. **Heroku:** создать новое приложение

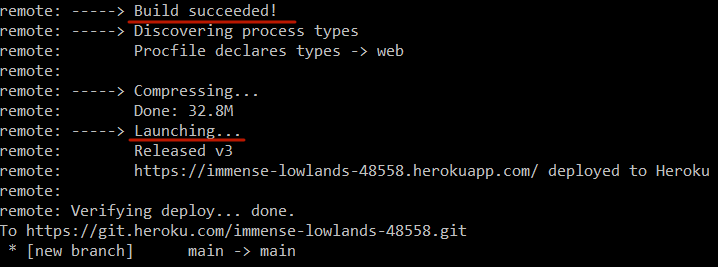


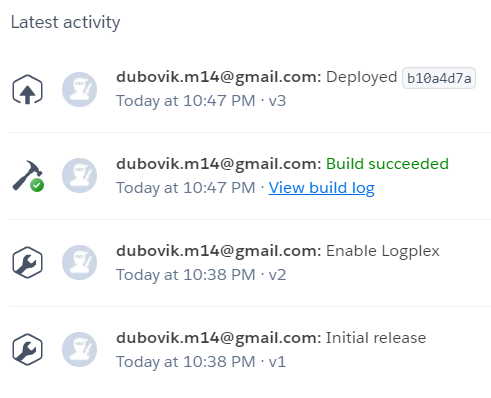




1. **Heroku:** публикация

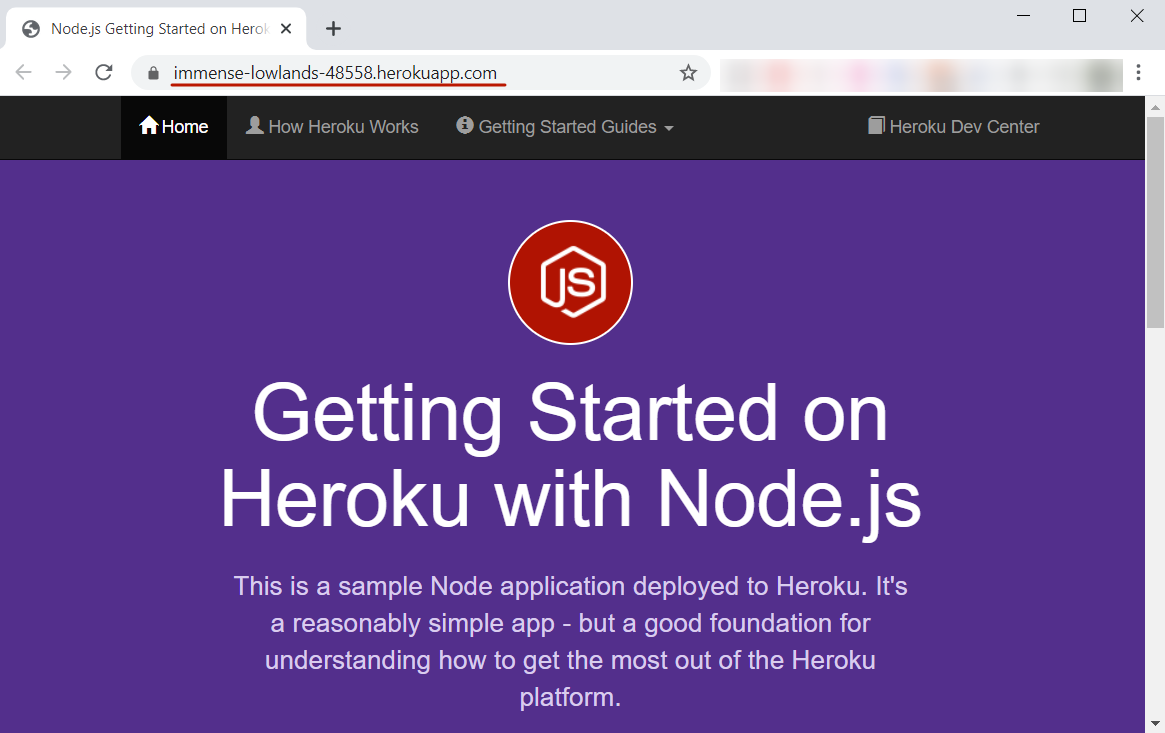






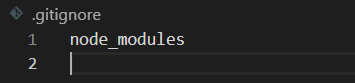
1. **Heroku:** открыть приложение



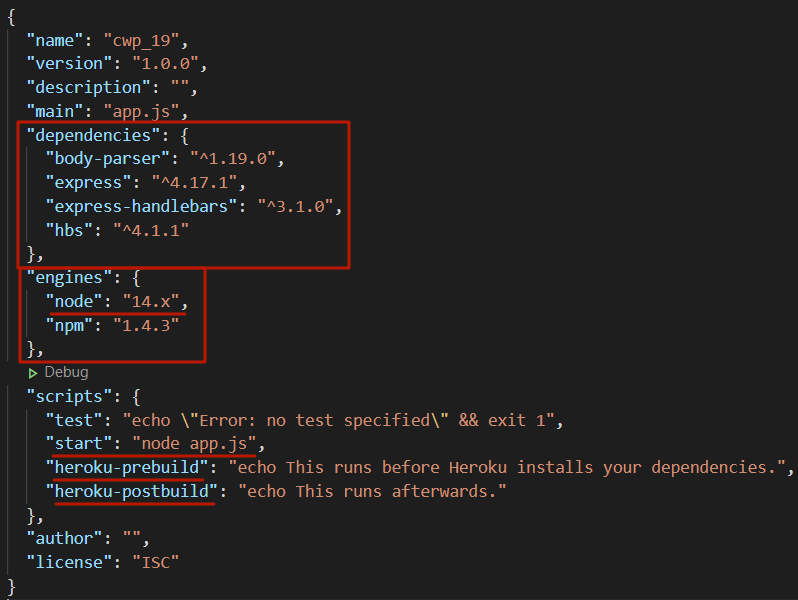


1. **Heroku:** про что не стоит забывать:

* добавить файл .gitignore и обязательно исключить папку node\_modules;



* проверить, чтобы везде использовались относительные пути;
* обязательно должен быть файл package.json, в котором необходимо определить зависимости (dependencies), версию node.js и стартовый скрипт (можно для этого использовать Procfile);

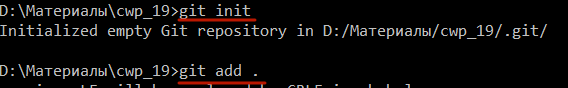


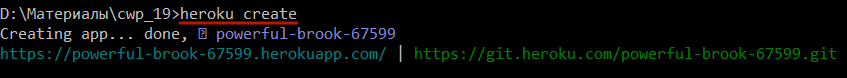


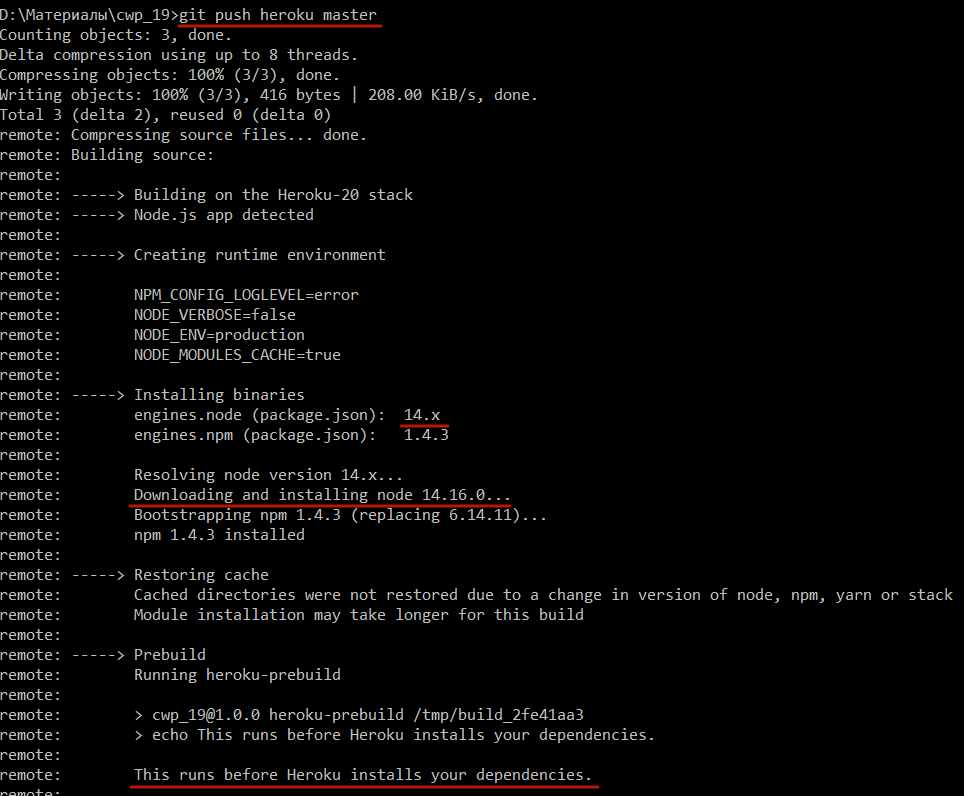
* указать порт для Heroku, использовать переменную среды PORT

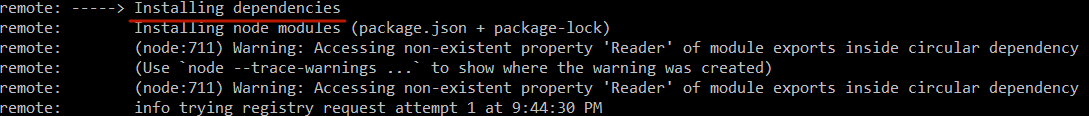


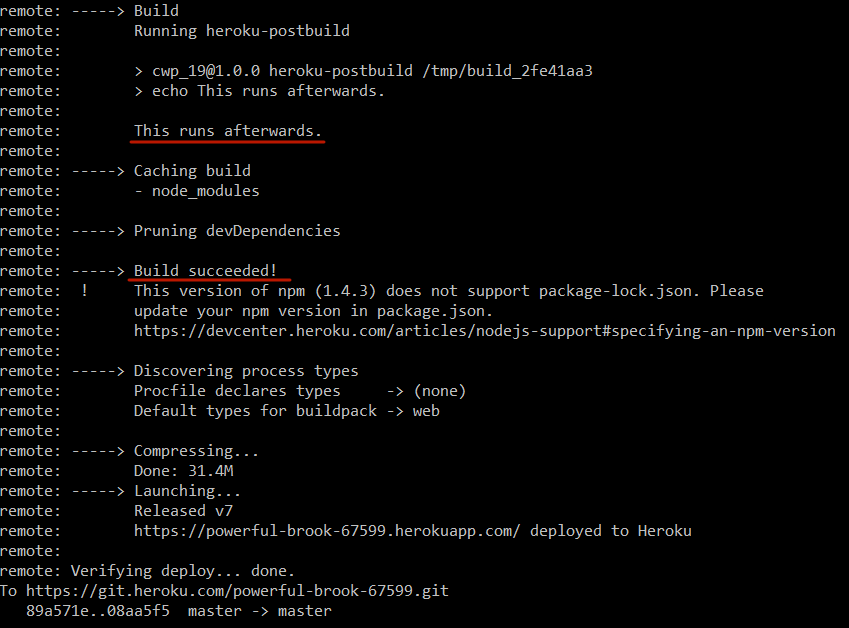
1. **Heroku:** пример



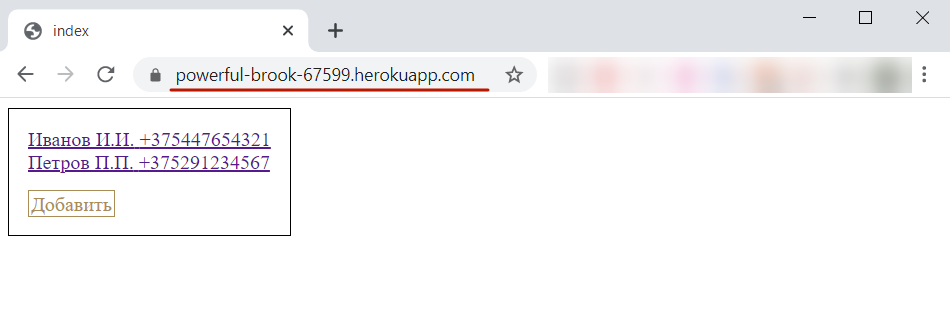




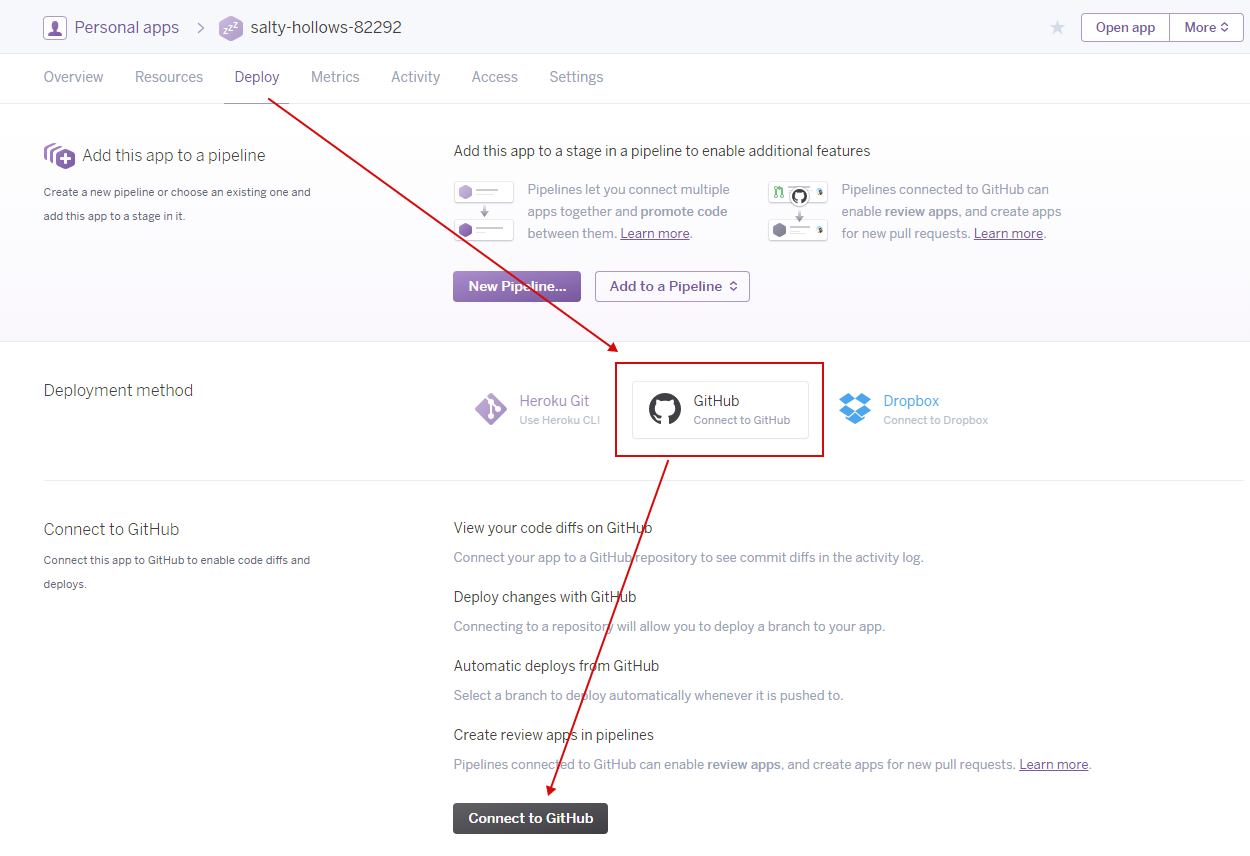


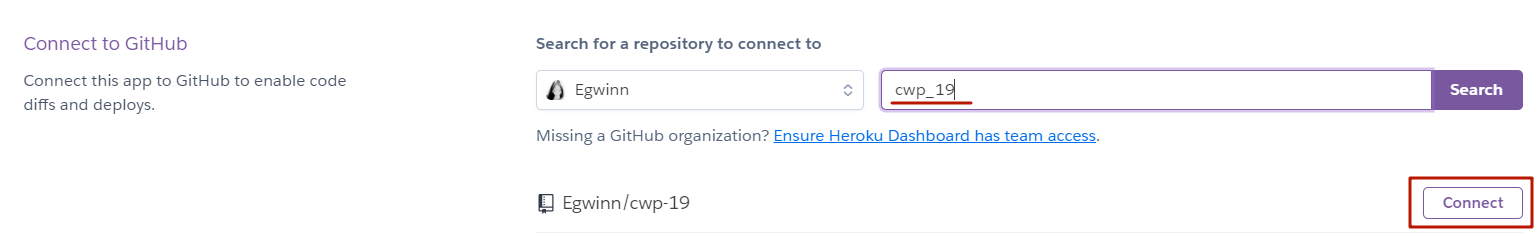


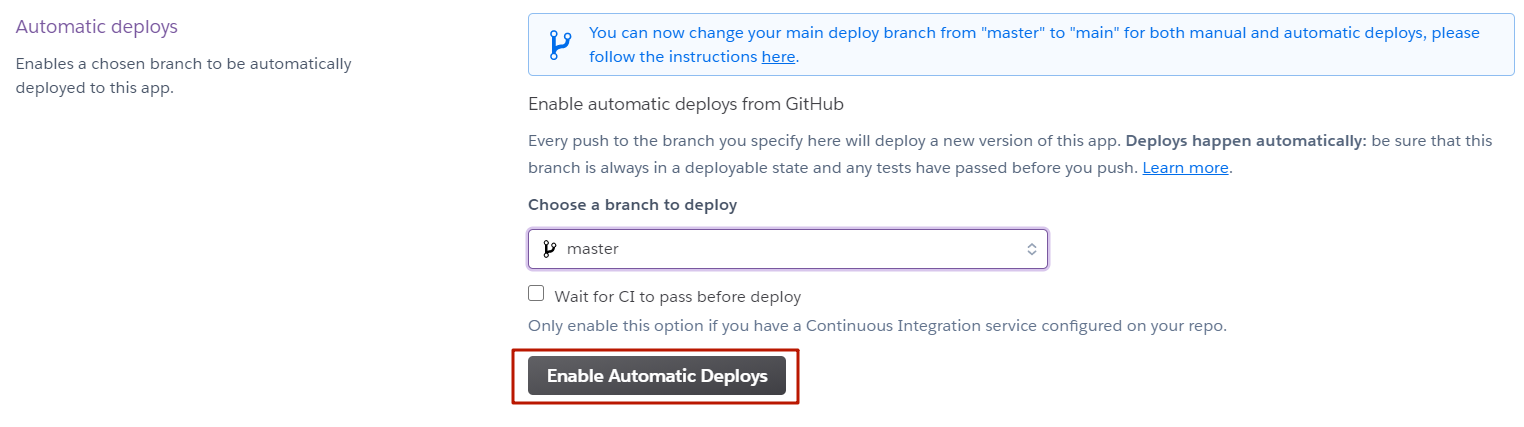


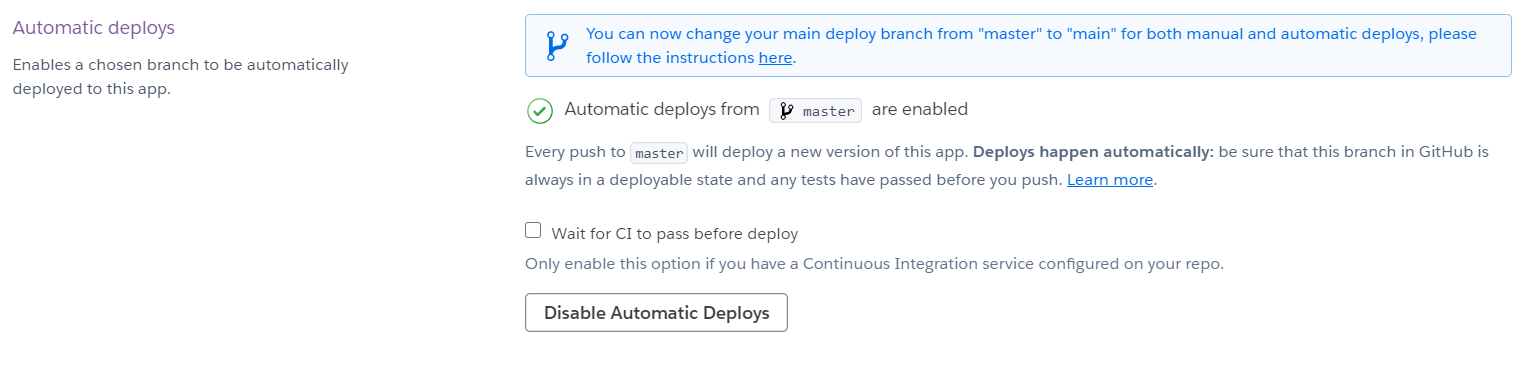


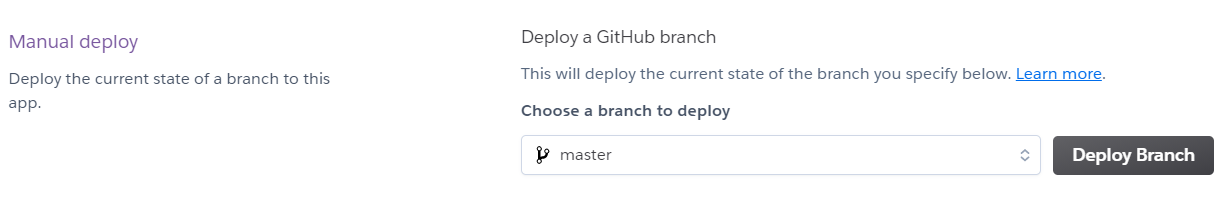
1. **CI/CD:** непрерывная интеграция (CI, Continuous integration) и непрерывное развертывание (CD, Continuous delivery).
2. **CI/CD:** практика программирования; частое слияние feature-веток с master; автоматизированные сборки; автоматизированное выполнение тестов; автоматизированный деплой; отправка отчетов.
3. **Heroku:** CI/CD











1. **Heroku:** add-on



