

Docker – это набор инструментов, которые позволяют различными командами упаковать ваше приложение или вообще любой программное обеспечение со всеми зависимостями. Можно запускать это приложение в любой операционной системе в так называемом контейнере. Под приложением здесь подразумевается все что угодно это может быть база данных это может быть сервер авторизации, это может быть веб-приложение, веб-сервер и т.д. Можно упаковать что угодно в контейнер и запускать его в любых операционных системах

Docker предоставляет в том числе и готовые образы для любого функционала (веб-сервер, БД и др.)

Основные понятия:

Image(образ) - это шаблон или blueprint для создания контейнера. Он содержит все необходимые файлы, библиотеки, зависимости и конфигурации для запуска определенного приложения.

Примеры: Образ Nginx, образ MySQL, образ WordPress.

Container (контейнер) - это экземпляр образа, который запускается и работает как изолированная среда. Он содержит все необходимое для запуска приложения, включая операционную систему, зависимости и конфигурации.

Примеры: Запущенный экземпляр образа Nginx, запущенный экземпляр образа MySQL.

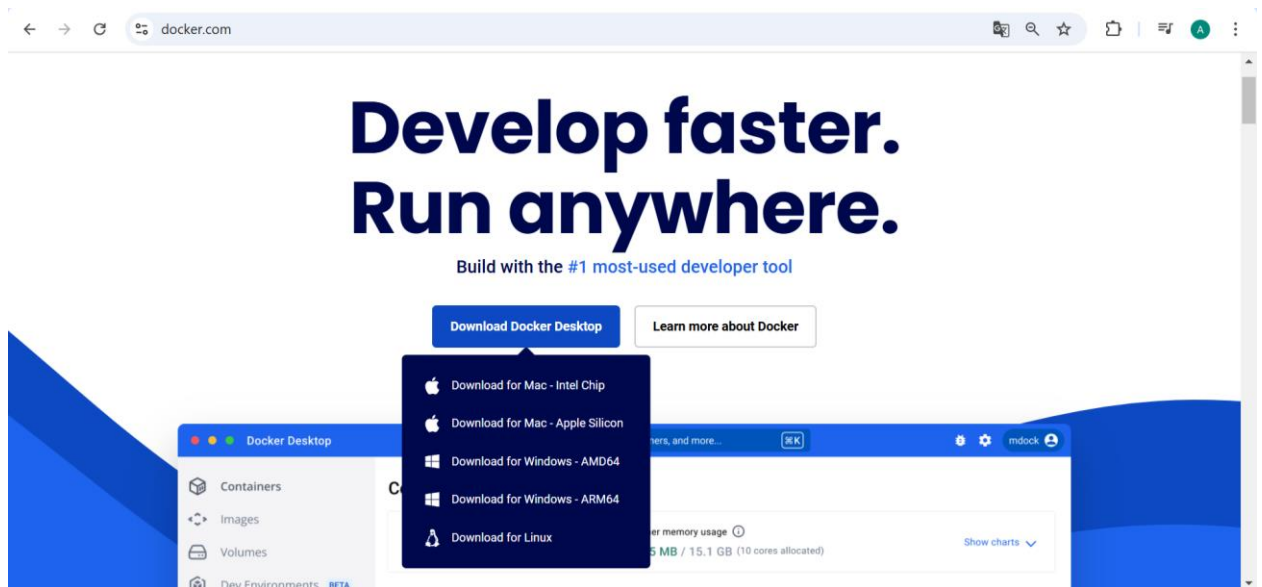
Daemon (служба) – процесс на хосте, который принимает команды, управляет Docker компонентами

Client (Клиент) – интерфейс для взаимодействия с Docker Daemon., позволяет отправлять команды Docker Daemon и управлять контейнерами, может быть использован через командную строку (`docker`) или через графический интерфейс.

Docker Engine – комбинация Docker Daemon и Docker Client обеспечивает полный набор функций для управления Docker контейнерами, запускается на хост-машине и позволяет запускать, управлять и масштабировать контейнеры.

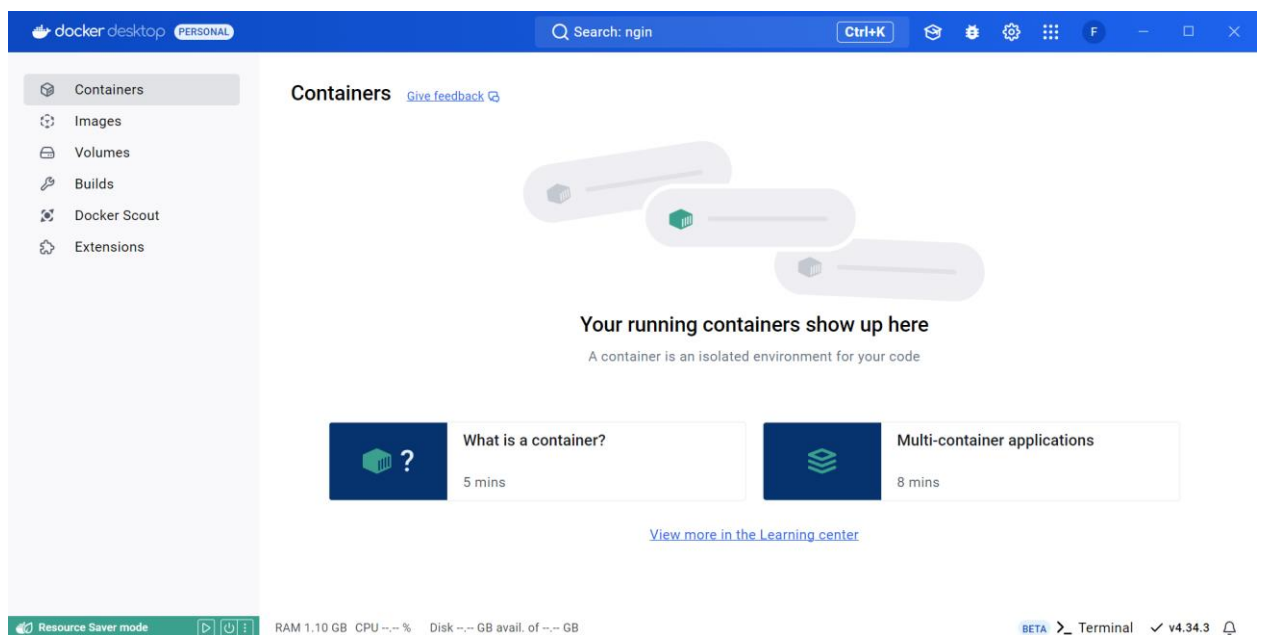
Установка:

На сайте <https://www.docker.com/> необходимо выбрать версию, в соответствии с операционной системой. Документация - <https://docs.docker.com/desktop/>



Docker изначально построен на операционной системе Linux. При установке Docker Desktop на Windows фактически внутри window будет работать виртуальная машина Linux.

Запустить Docker Desktop. Зарегистрироваться и войти под своим аккаунтом. Если все успешно, то экран должен выглядеть примерно следующим образом



Проверить, что все успешно можно еще одним способом: запустить командную строку (от имени администратора) и ввести команду `docker version`

```
Выбрать Администратор: Командная строка
Microsoft Windows [Version 10.0.19045.5011]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

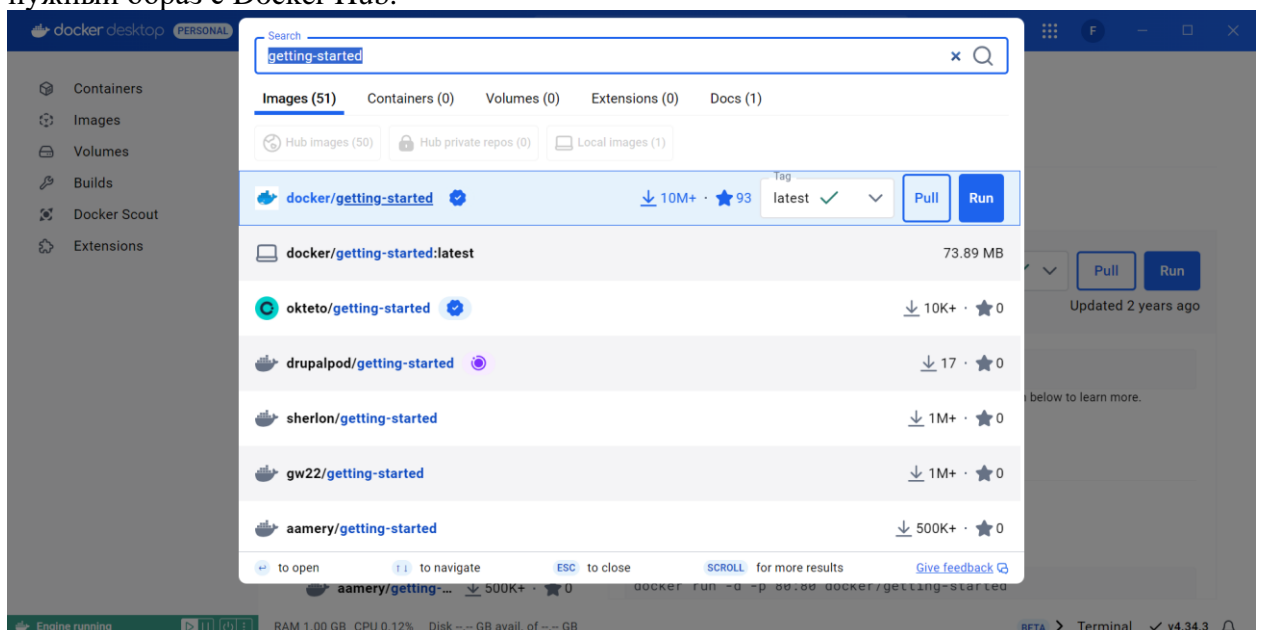
C:\Windows\system32>docker version
Client:
 Version:           27.2.0
 API version:       1.47
 Go version:        go1.21.13
 Git commit:        3ab4256
 Built:             Tue Aug 27 14:17:17 2024
 OS/Arch:           windows/amd64
 Context:           desktop-linux

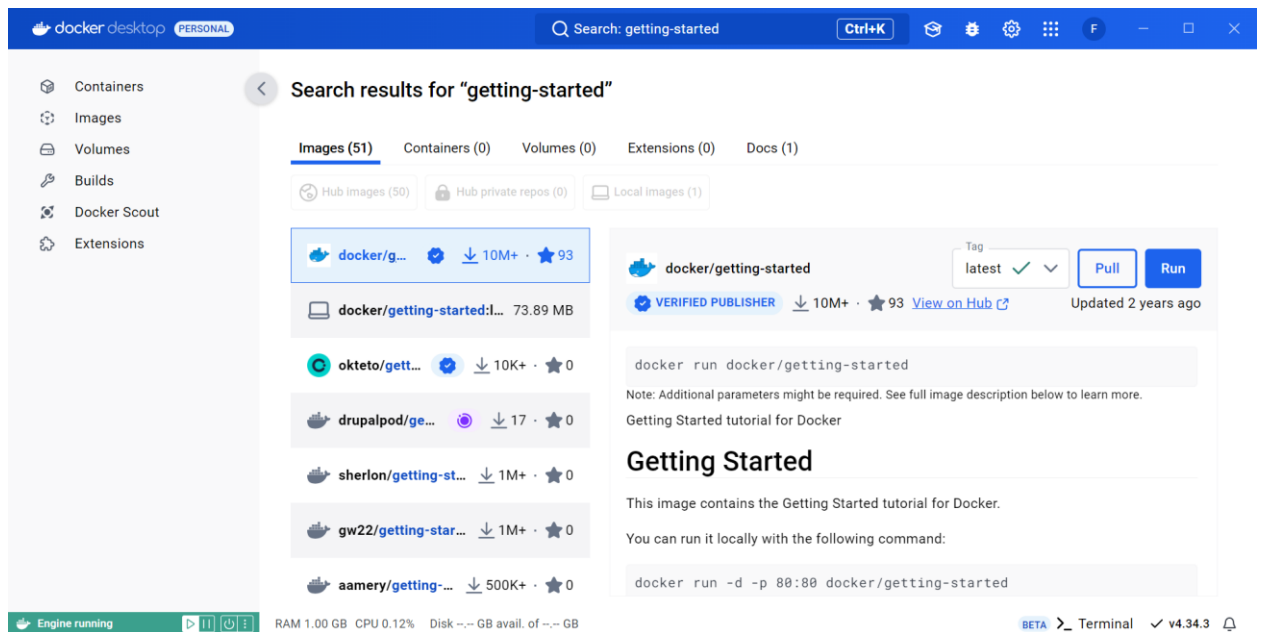
Server: Docker Desktop 4.34.3 (170107)
Engine:
 Version:           27.2.0
 API version:       1.47 (minimum version 1.24)
 Go version:        go1.21.13
 Git commit:        3ab5c7d
 Built:             Tue Aug 27 14:15:15 2024
 OS/Arch:           linux/amd64
 Experimental:      false
containerd:
 Version:           1.7.20
 GitCommit:         8fc6bcbff51318944179630522a095cc9dbf9f353
runc:
 Version:           1.1.13
 GitCommit:         v1.1.13-0-g58aa920
docker-init:
 Version:           0.19.0
```

Обзор Docker Desktop

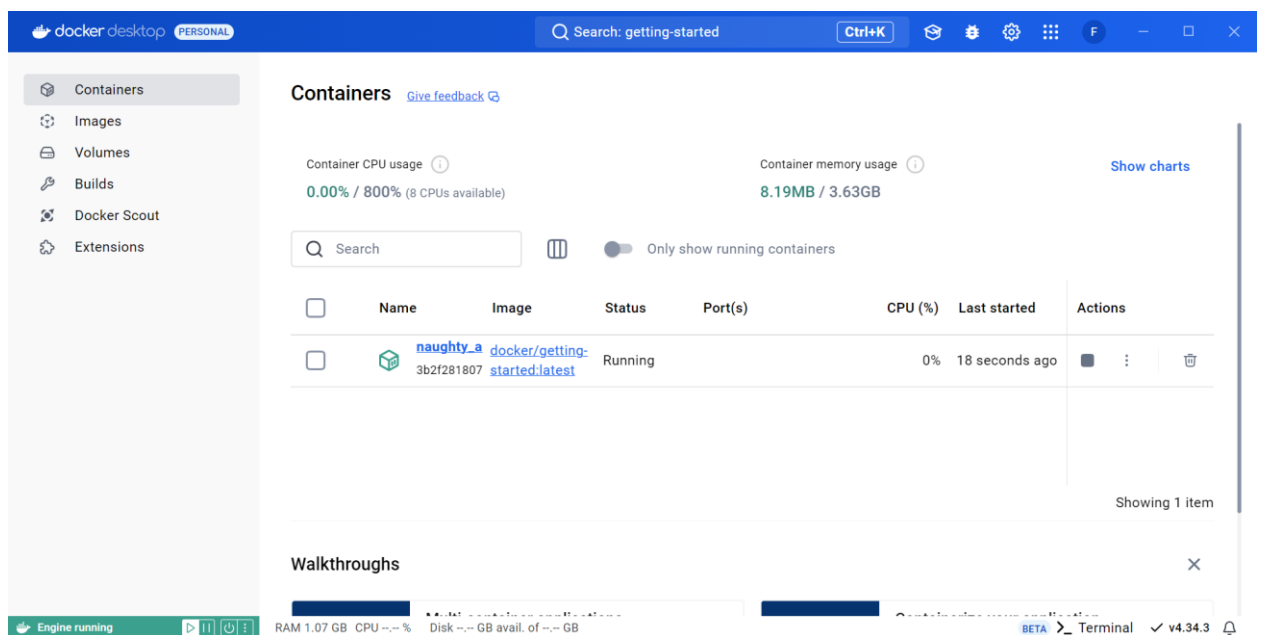
Во вкладке Containers показаны все запущенные контейнеры. На основе одного образа можно запускать несколько контейнеров.

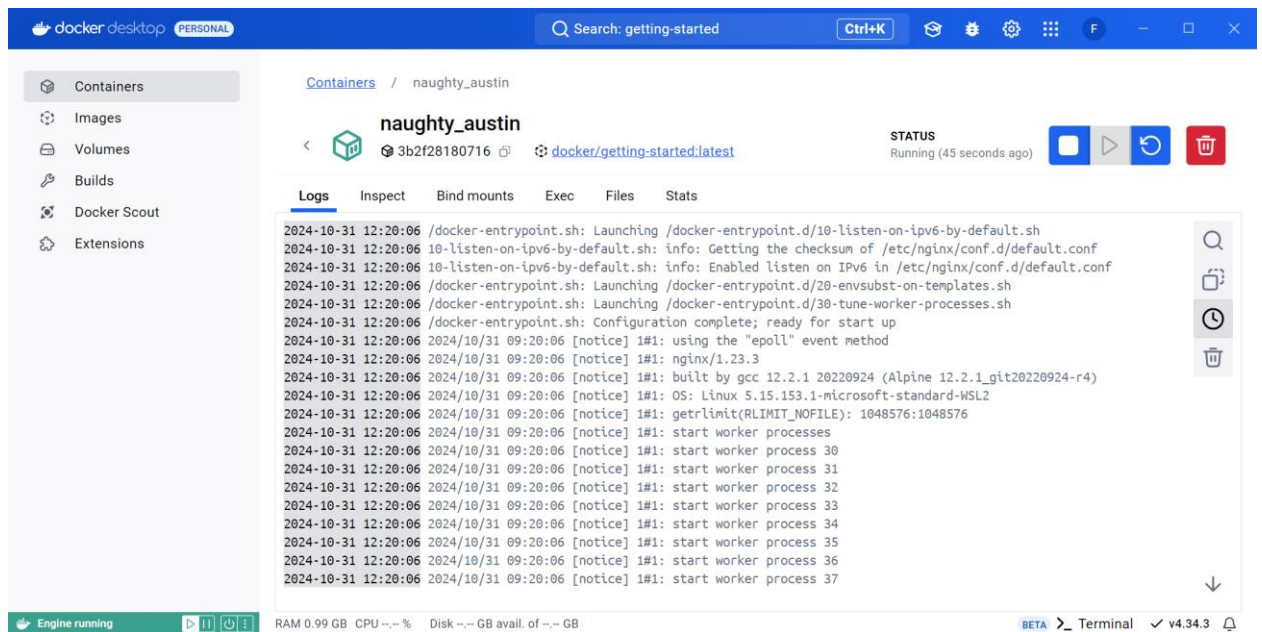
Во вкладке Images показаны все образы. Можно воспользоваться поиском и скачать нужный образ с Docker Hub.





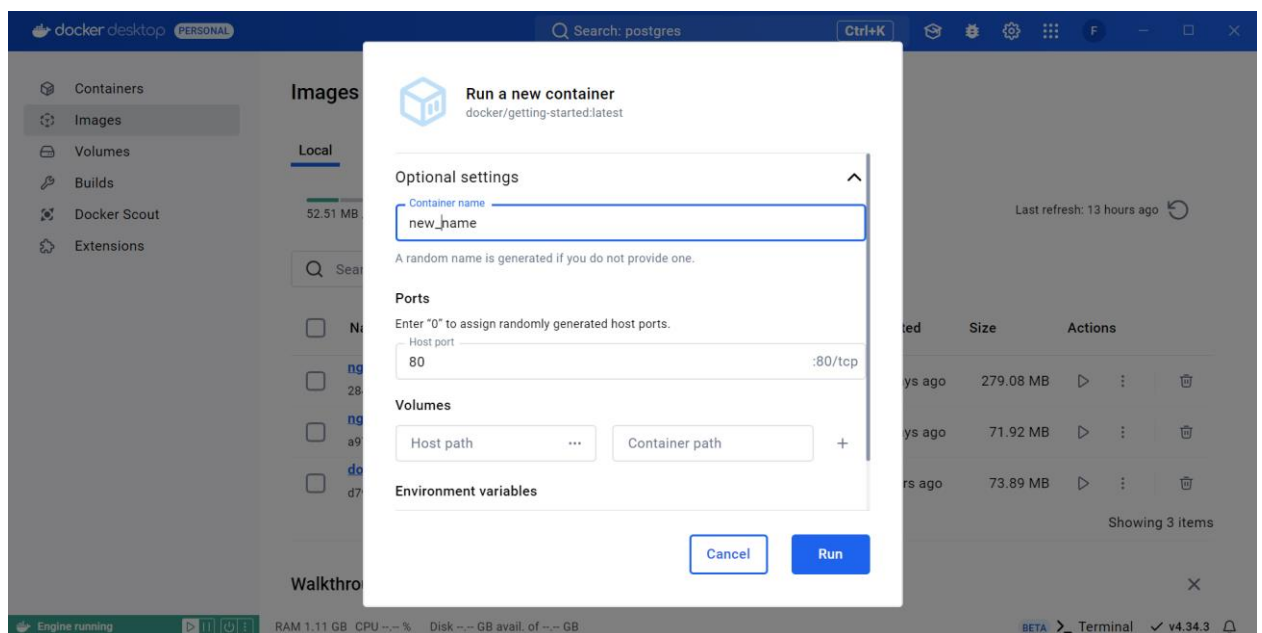
Найдем и скачаем образ getting-started
Запустим его





Существуют alpine образы. Они используют базовый слой операционной системы с минимальными пакетами.

Указание портов (запуск с параметрами)



Перейдя в браузер по адресу localhost

localhost/tutorial/

docker Labs Getting Started

Search

docker/getting-started
2.9k Stars · 6.4k Forks

Getting Started

- Getting Started
- Our Application
- Updating our App
- Sharing our App
- Persisting our DB
- Using Bind Mounts
- Multi-Container Apps
- Using Docker Compose
- Image Building Best Practices
- What Next?

Getting Started

The command you just ran

Congratulations! You have started the container for this tutorial! Let's first explain the command that you just ran. In case you forgot, here's the command:

```
docker run -d -p 80:80 docker/getting-started
```

You'll notice a few flags being used. Here's some more info on them:

- `-d` - run the container in detached mode (in the background)
- `-p 80:80` - map port 80 of the host to port 80 in the container
- `docker/getting-started` - the image to use

Pro tip

Задание: вывести версию скачанного docker, вывести справочную информацию о docker. Необходимо установить nginx, заранее просмотрев все доступные образы (docker search). Запустить контейнер на порту (в примере используется 80, необходимо выбрать другой) с name равным фамилии студента (в примере будет my_nginx) и проверить, что он работает, посмотреть логи, удалить образ.