Используемая лекция

https://www.youtube.com/watch?v=lr1rYnUubpQ

*предыстория, почему нужен docker:*

Сервер - компьютер, мощность и ресурсы которого увеличены про сравнению с обыкновенным ПК, для того, чтобы их хватало на использование приложения или программы.

(ресурсы такие как оперативная память, процессор, жесткий диск)

Если программа работает на сервере, то увеличение обращений к ней и ее инструментам влекут за собой увеличение нагрузки. При использовании всеми частями программы (например, БД и программой для кэш сервера) одних ресурсов сервера, возможно их неравномерное распределение, которое влияет на работу программы и на ожидаемый результат. Для примера: при направлении большого кол-ва запросов к программе, она «забирает» ресурс у БД, следовательно, уже при необходимости получить данные от БД, ресурса сервера, который остался для БД может не хватить.

Сервер

Виртуальная машина:

разделение ОС на одном сервере, для выделения определенного кол-ва ресурсов и мощностей.

каждая программа на сервере использует только те ресурсы и те мощности железа, которые ей выделены ОС-ой.

ВМ

ВМ

ВМ

По результату использования виртуальных машин на сервере мы вместо 1 ос используем несколько, в зависимости от того, сколько требуется нашему приложению (грубо говоря, выглядит это как для каждой программы, которая работает на сервере, используется свой сервер, хотя по факту сервер один)

Итого для чего это нужно?

-Изоляция программы от других программ на одном и том же сервере

- Распределение ресурсов сервера между программами

Проблемы: настройка этих виртуальных машин

Ранее программы и настройки сервера необходимо было произвести 1 раз (т.к. ос система одна), а при использовании виртуальных машин такая настройка соответственно усложняется.

Плюс, при необходимости использовать дополнительный сервер, время и ресурс настройки соответственно увеличатся.

**Docker**

Что такое docker и для чего он используется?

Программа, использование которой помогает разворачивать приложение на виртуальной машине, на которой также будут использоваться все необходимые для приложения инструменты (те, что указаны в настройках) с указанным в настройках распределением ресурсов.

В чем преимущество использования docker?

С помощью созданного image, в котором хранится все необходимое для запуска приложения, мы можем развернуть приложение на любом компьютере.

Для того, чтобы запустить приложение необходимо:

- установить программу докер

- загрузить в нее нужный image

- запустить в докер загруженный image

Результат: на машине разворачивается виртуальная машина, которая содержит в себе те необходимые элементы, приложения, определенные настройки по ресурсам, которые мы определили как необходимые для работы нашего приложения.

Само приложение запускается в docker container.

Для того, чтобы можно было развернуть приложение и необходимые для него ресурсы необходим docker image.

**docker image** (образ) - архив, в котором хранятся все необходимые компоненты, описания и команды, необходимые для запуска приложения

В него входит:

- урезанная ОС только с необходимыми элементами

- код приложения

- среда исполнения кода

- зависимости и доп. программы

- настройки приложения

- конфигурация запуска приложения

Используется для автоматического развертывания виртуальной машины на компьютере с указанными настройками и зависимостями

**docker container** - это запущенное приложение на основе шаблона image. Выше смотри описание, что входит в image, есть пункт "код приложения".

Работает на той ОС, которая актуальная на виртуальной машине. Грубо говоря, в image прописана глобальная настройка, а контейнер - это именно работающее приложение, которое использует необходимые, настроенные image, ресурсы.

Преимущества контейнеров:

- малый размер, используют только необходимый минимум компонентов для работы;

- моментальный запуск;

- сниженное потребление ресурсов.

Что необходимо, чтобы работать с docker?

Для того, чтобы запускать приложение через docker, необходимо в приложении создать и использовать **DockerFile.txt**

**DockerFile -** файл, на основании которого создается наш image; это инструкция для построения image + команды для запуска приложения.

В этом файле написано, что нужно сделать, а Docker сам сделает image для приложения.

командой docker build \*\*\*\*\*{тут дополнения по названию проекта и где хранится файл} передаем dockerfile для создания image

**DockerHub** - облачное хранилище для image.

Под стандартные программы, например, Redis, Postgres, на DockerHub можно скачать уже готовые image в облачном хранилище.

Итого по всем элементам:

DockerFile в корне приложения, на основании которого Docker создает image, после загрузки image запускается Docker Container.