

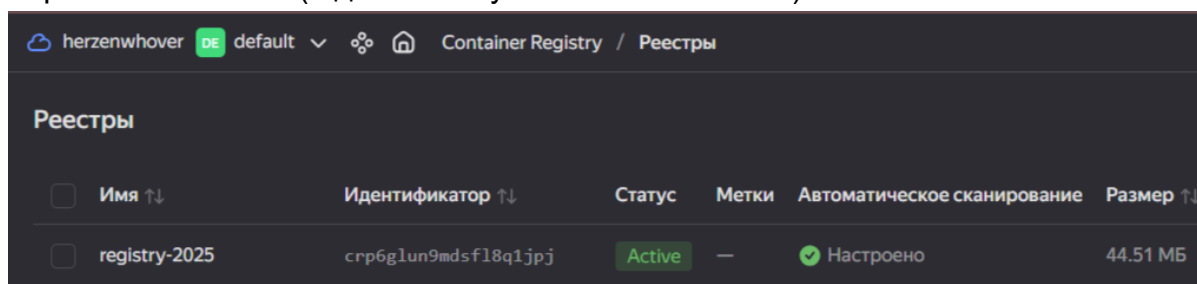
Проверка EA_IBT1.1_КП4_тема12_YandexServerless Docker

Начало работы

- Для начала работы потребовалось установить DockerDesktop и расширения для VScode Docker.
- Был скачен и разархивирован архив с начальным кодом представленный в мул.

Работа с Yandex Cloud и docker

- В сервисе Yandex Cloud был создан Container Registry для хранения и управления Docker-образами в облаке (в данном случае в Yandex Cloud).



The screenshot shows the Yandex Container Registry web interface. At the top, there's a header with the username 'herzenwhoover', a dropdown menu set to 'default', and navigation icons. Below the header, the title 'Container Registry / Реестры' is visible. The main section is titled 'Реестры' and contains a table with the following columns: 'Имя' (Name), 'Идентификатор' (Identifier), 'Статус' (Status), 'Метки' (Tags), 'Автоматическое сканирование' (Automatic scanning), and 'Размер' (Size). There is one entry in the table: 'registry-2025' with identifier 'crp6glun9mdsf18q1jpj', status 'Active', no tags, scanning status 'Настроено' (Configured), and a size of '44.51 МБ'.

Имя	Идентификатор	Статус	Метки	Автоматическое сканирование	Размер
registry-2025	crp6glun9mdsf18q1jpj	Active	—	Настроено	44.51 МБ

- Была выполнена аутентификация в Container Registry. Для этого потребовался OAuth-токен, успешно сгенерированный через документацию к [Yandex Container Registry](#). После чего в корне проекта была введена команда для аутентификации в cmd:
`echo <OAuth-токен>|docker login --username oauth --password-stdin cr.yandex`
- Далее для создания docker образа была использована команда:

```
docker build . -t  
cr.yandex/crp6glun9mdsf18q1jpj/kp12_demo_flask_yandex_serveless:v0.0.1`
```

Где:

cr.yandex — доменное имя сервиса Container Registry

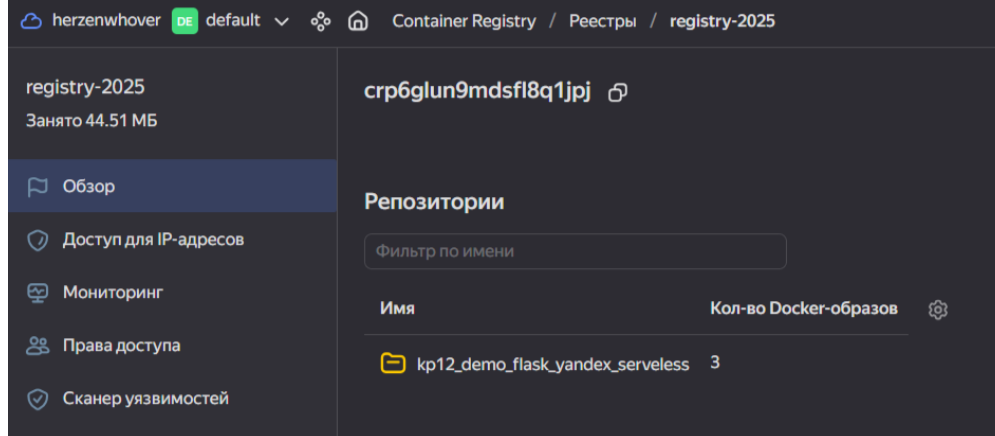
crp6glun9mdsf18q1jpj — идентификатор созданного Container Registry

kp12_demo_flask_yandex_serveless — название образа

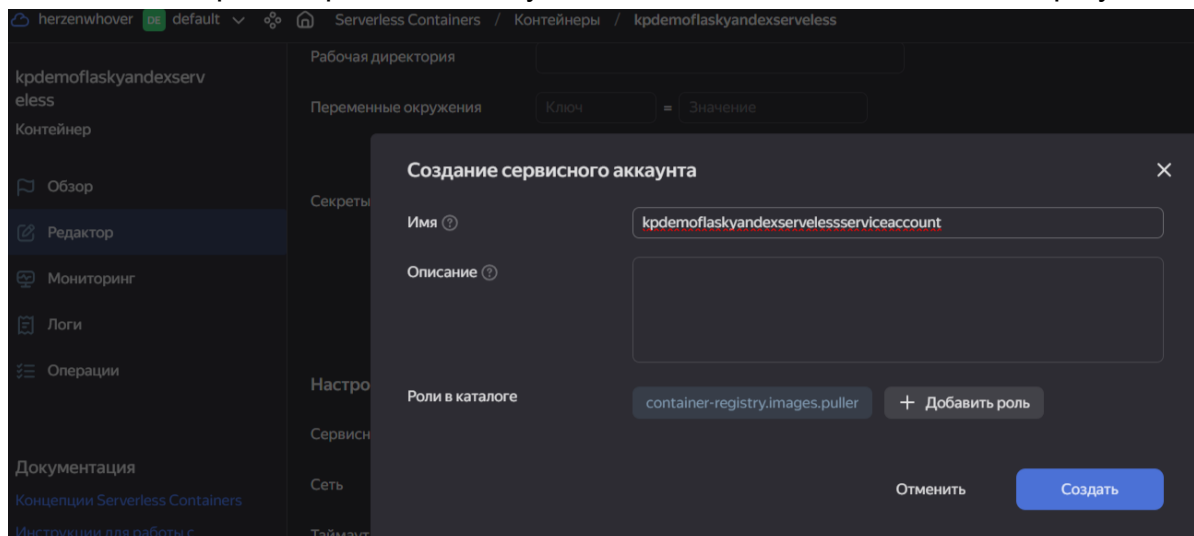
v0.0.1 — тег

Для работы команды требуется предварительно запустить Docker Desktop.

- Данный образ был загружен в реестр командой:
`docker push cr.yandex/crp6glun9mdsf18q1jpj/kp12_demo_flask_yandex_serveless:v0.0.1`



- После создания и загрузки его в реестр, требуется создать сервисный контейнер. Главное в настройках контейнера требуется URL параметра образа, который был создан ранее и создать и выбрать сервисный аккаунт. Обязательно сделать контейнер публичным.



Вывод

Была выполнена работа с использованием Yandex Cloud и Docker. Был создан и запущен сервисный контейнер, с которым можно ознакомиться по ссылке

<https://bbahrlidabbkr6i3e6o7h.containers.yandexcloud.net/>.

Для данной работы не был создан репозиторий с кодом, так как от начального кода он не отличается.