

Запускаем Deployment

1. Перейдите в директорию: `1.5.deployment`

```
cd ~/slurm/practice/1.kube-basics-lecture/1.5.deployment/
```

2. Создайте deployment из файла `deployment.yaml`

3. Проверьте что появились два запущенных pod'а и один replicaset.

4. С помощью команды `kubectl edit deployment my-deployment` измените версию образа в deployment'e на `nginx:1.13`

5. Проверьте, что все pod'ы deployment'a заново создались и версия образа в них теперь `nginx:1.13`

6. Скопируйте и отправьте в качестве ответа на данном шаге результат выполнения команды:

```
kubectl get deployment my-deployment -o  
jsonpath='{.status.conditions[1].message}'
```

7. После выполнения задания удалите deployment.

Самостоятельная работа

1. Отредактируйте deployment из предыдущего задания таким образом, чтобы он запускался с **одной** репликой и rolling update проходил без downtime. То есть при обновлении образа всегда должен оставаться один рабочий pod.

Используйте поля `maxSurge` и `maxUnavailable` со значениями, приведенными в видео для деплойментов с одной репликой.

2. Обновите образ с `nginx:1.12` на `nginx:1.13`. Посмотрите в каком порядке запускаются и тушатся pod'ы в момент обновления.

3. Выполните команду (скопировать именно всё):

```
kubectl get deployment my-deployment -o  
custom-columns='NAME:.metadata.name,MAXSURGE:.spec.strategy.rollingUpdate.maxSurge,  
MAXUNAVAILABLE:.spec.strategy.rollingUpdate.maxUnavailable'
```

Это еще одна возможность ключа `-o`. Она позволяет вывести описание объекта с пользовательским набором полей.

Отправьте результат ее выполнения как ответ на этот шаг.

4. Удалите deployment после выполнения задания.