|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Информатика, искусственный интеллект и системы управления

КАФЕДРА Системы обработки информации и управления\_\_\_\_\_\_\_\_\_\_\_

**ОТЧЕТ**

**К ЛАБОРАТОРНОЙ РАБОТЕ № 3**

**По дисциплине «Методы поддержки принятия решений»**

**ОБУЧЕНИЕ ПОЛНОСВЯЗНОЙ НЕЙРОННОЙ СЕТИ**

Студент ИУ5-71Б  С.С. Кондрахин

(Группа) (Подпись, дата) (И.О.Фамилия)

Преподаватель  Д.О. Ишков

(Подпись, дата) (И.О.Фамилия)

Москва – 2023

**СОДЕРЖАНИЕ**

[1. Запуск кода 3](#_Toc151735844)

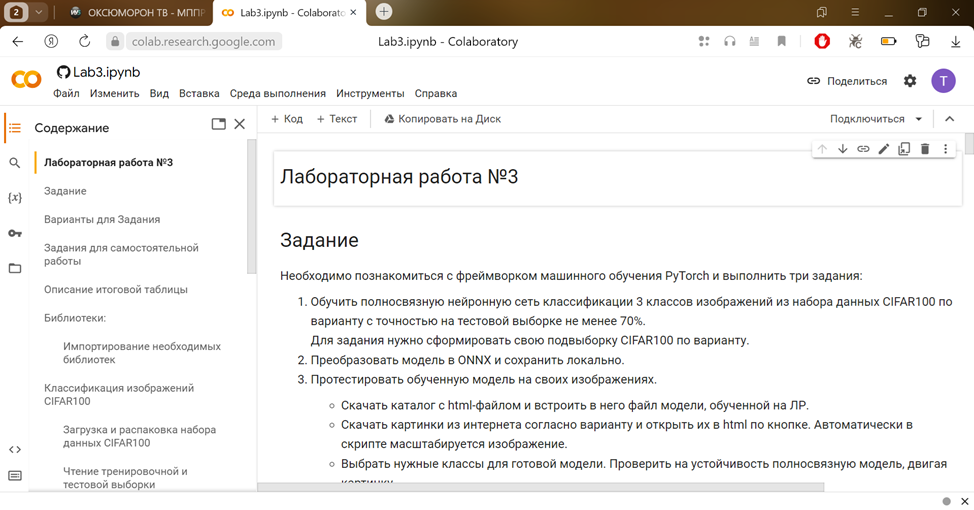
[2. Сборка папки 10](#_Toc151735845)

[3. Работа с моделью 14](#_Toc151735846)

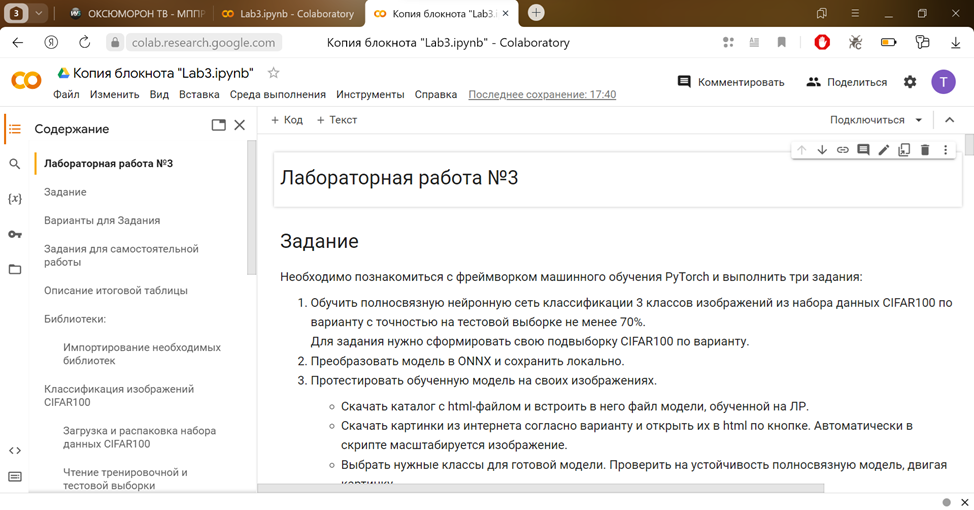
[4. Улучшение модели 25](#_Toc151735847)

# Запуск кода

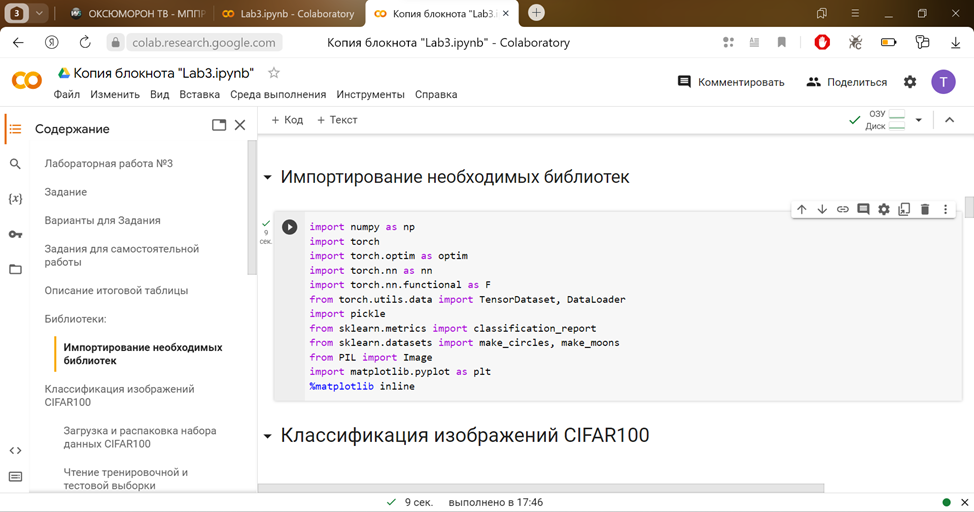
Откроем файл лабораторной работы в Google Colab: <https://colab.research.google.com/github/iu5git/MPPR/blob/main/notebooks/Lab3.ipynb>



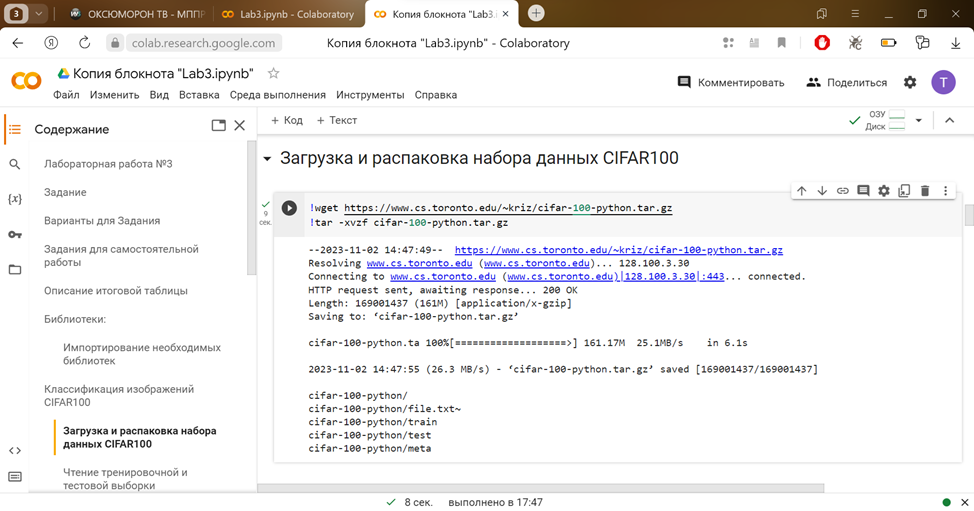
Скопируем файл на диск, чтобы могли сохраняться наши изменения в файле, нажав на кнопку «Копировать на Диск».



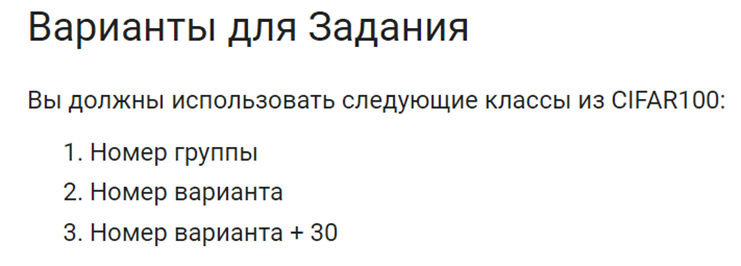
Теперь последовательно запустим все шаги. Сначала импортируем все необходимые библиотеки.

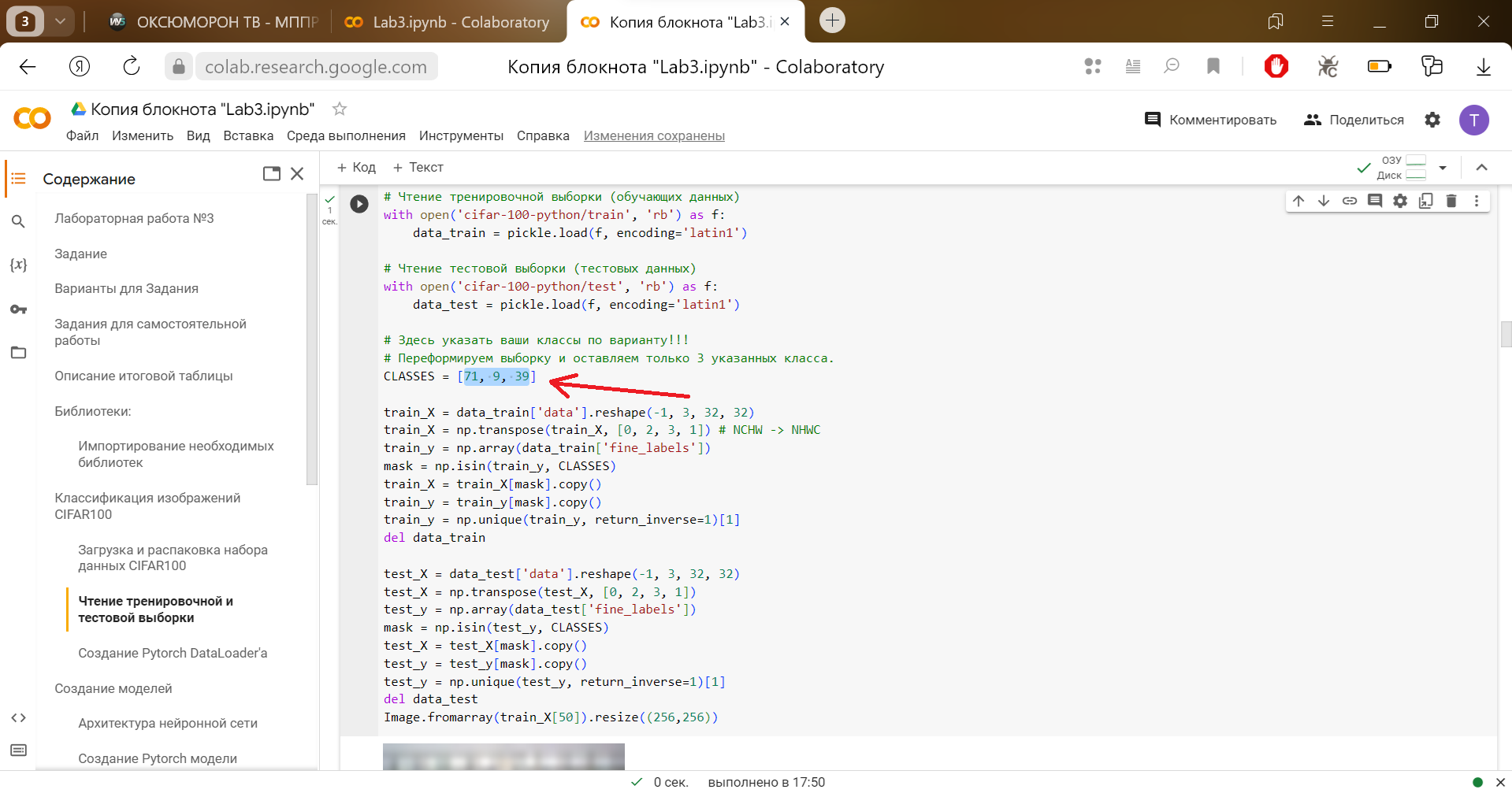


Загружаем и распаковываем набора данных CIFAR100.

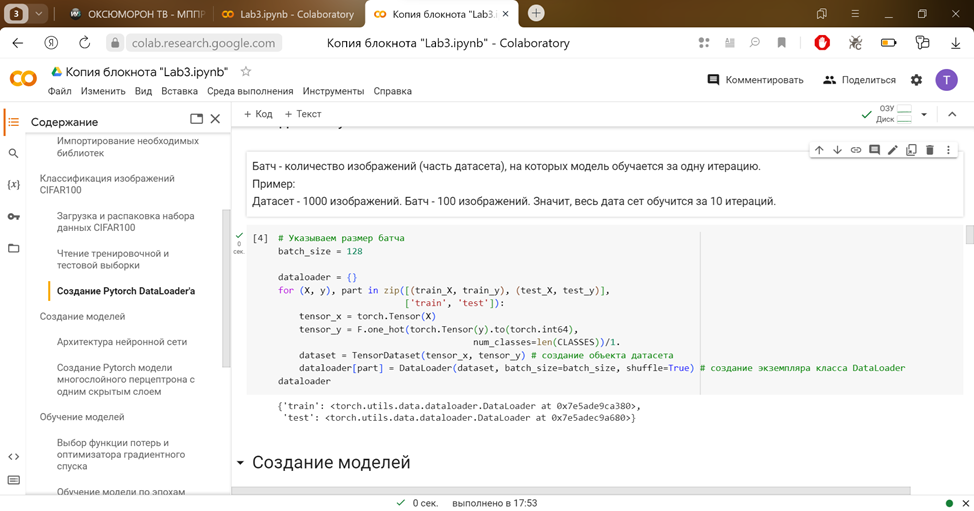


Разбиваем датасет на обучающие и тестовые данные. !!!Также необходимо указать классы по своему варианту.!!!

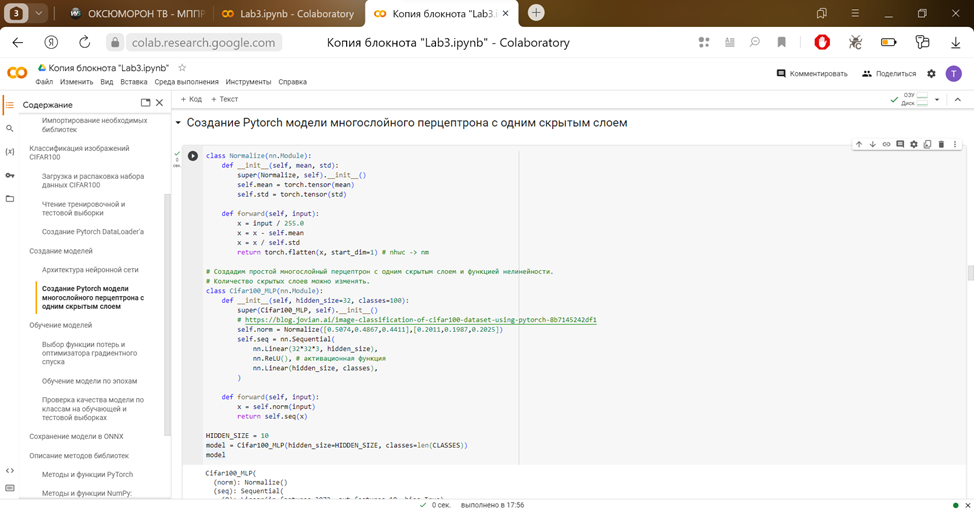




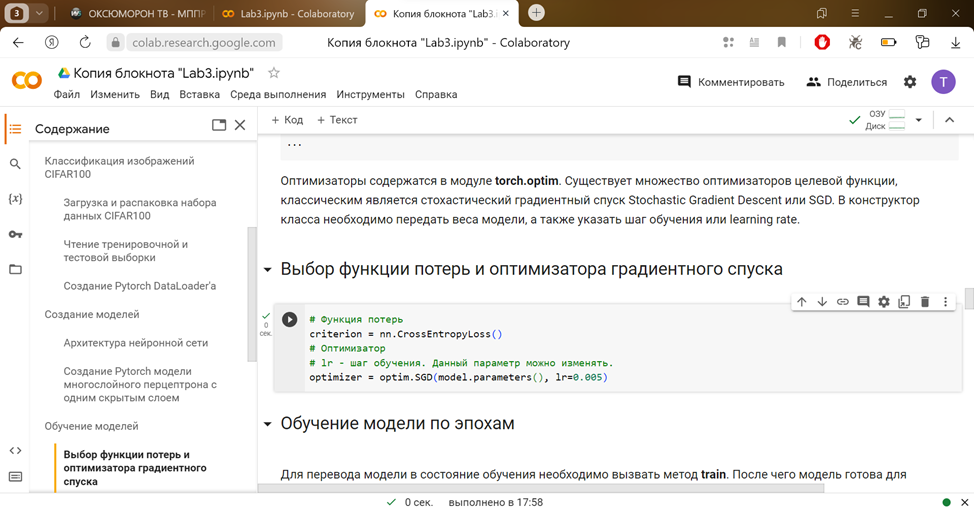
Задаем размер батча.



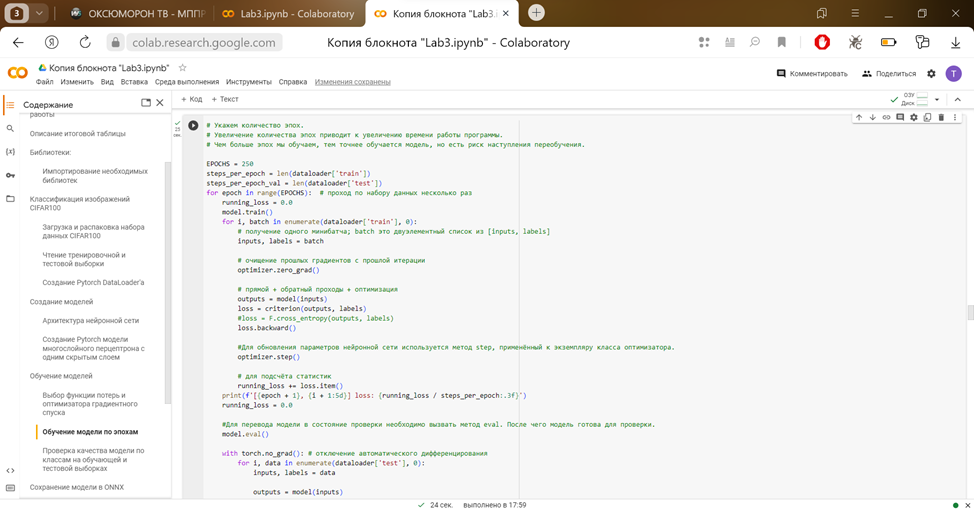
Создаем модель.



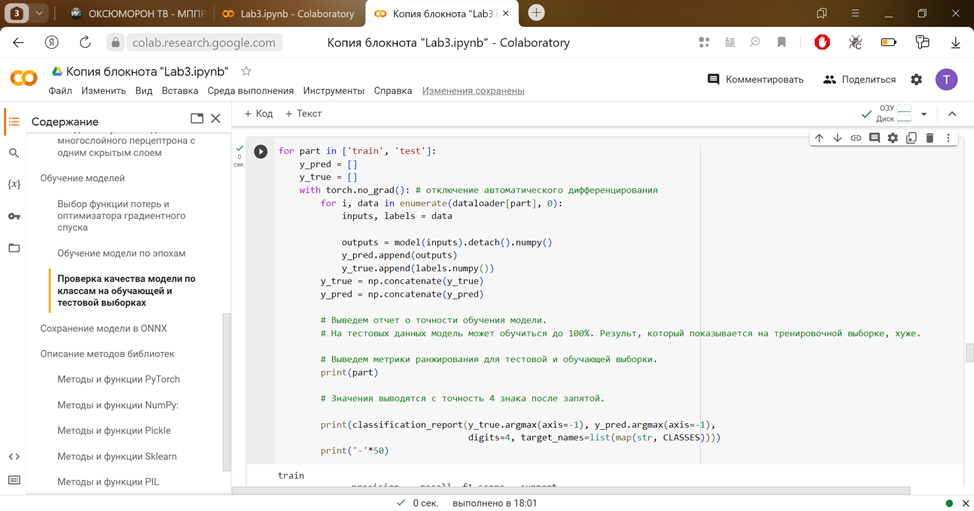
Выбираем функции потерь и оптимизатора градиентного спуска.



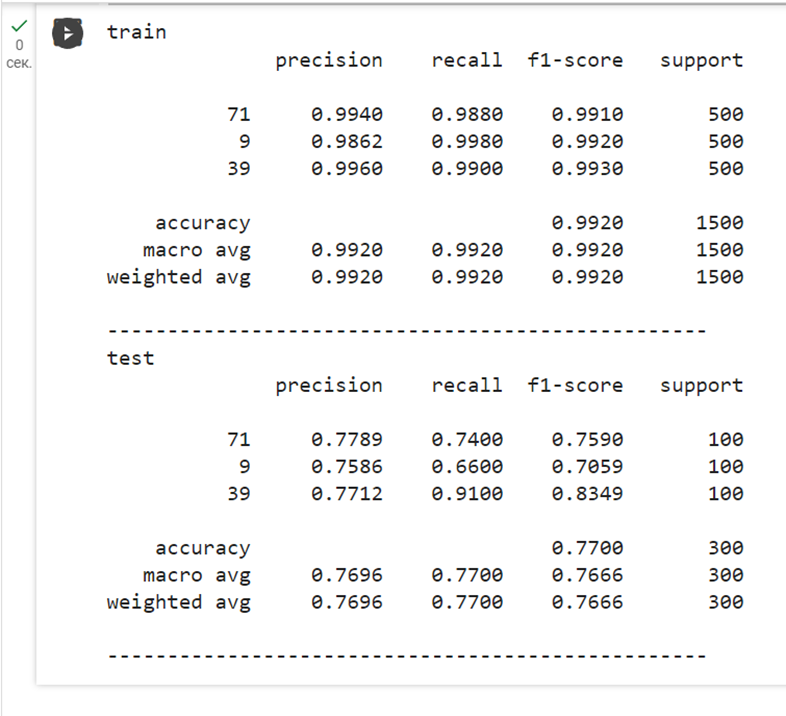
Обучаем модель по эпохам.



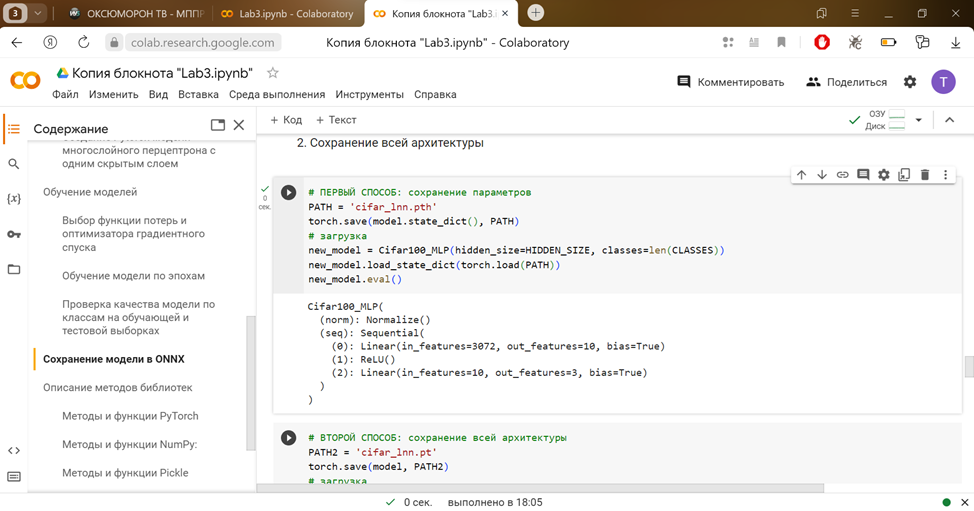
Проверка качества модели по классам на обучающей и тестовой выборках.

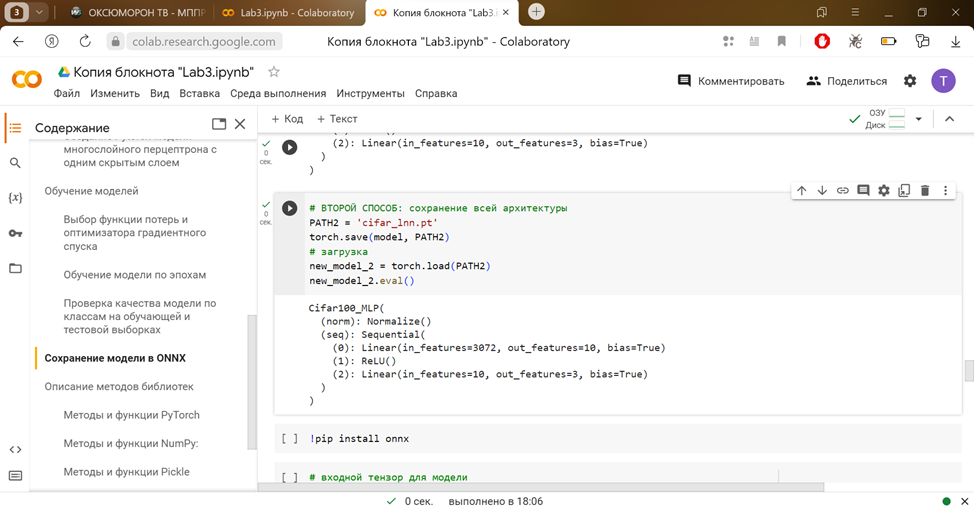


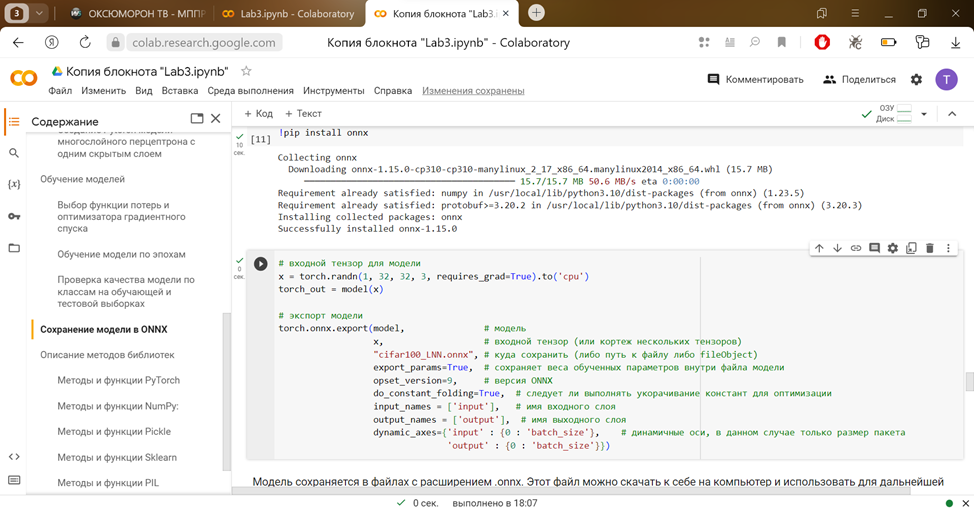
!!!По условию точность модели на тестовых данных должна быть более 70%.!!! В данном случае значение accuracy равно 77%, а значит наше условие выполняется.



Сохранение модели в ONNX.

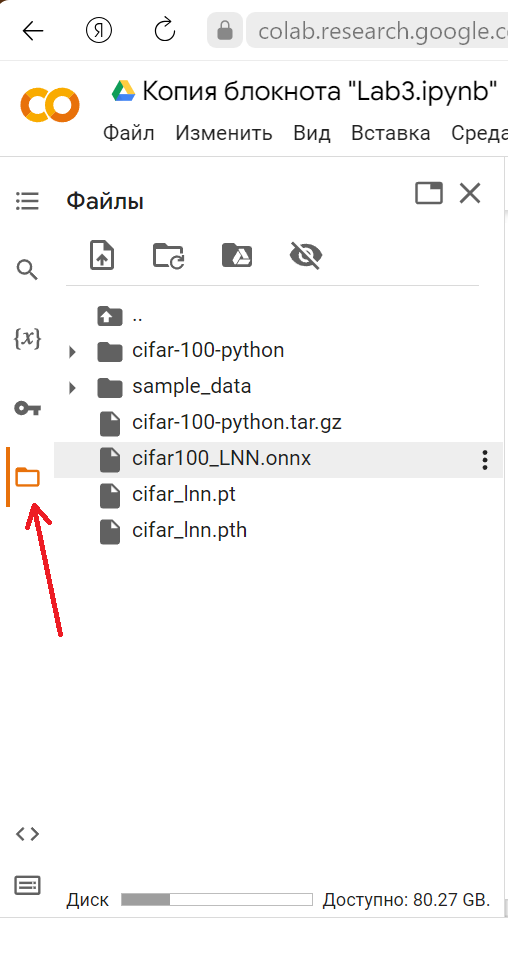


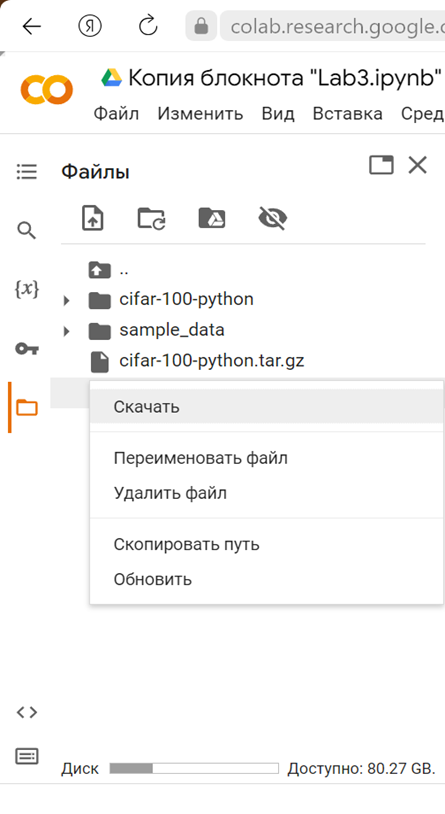




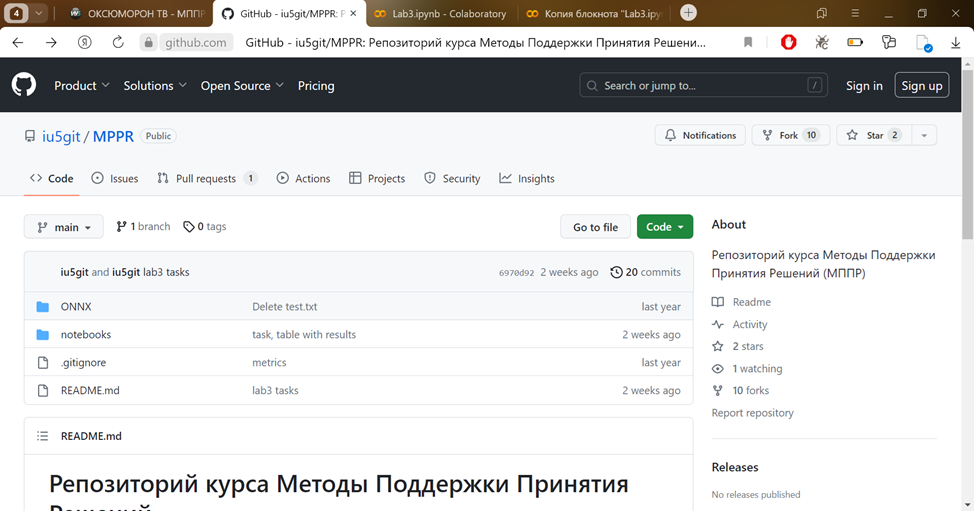
# Сборка папки

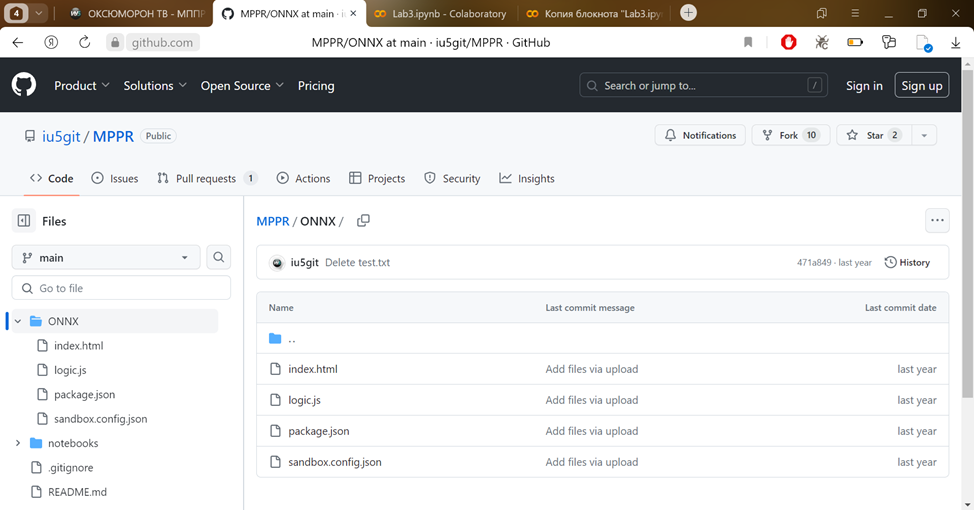
После запуска всех шагов нужно нажать на значок папки в левой части документа и скачать файл с расширением onnx.



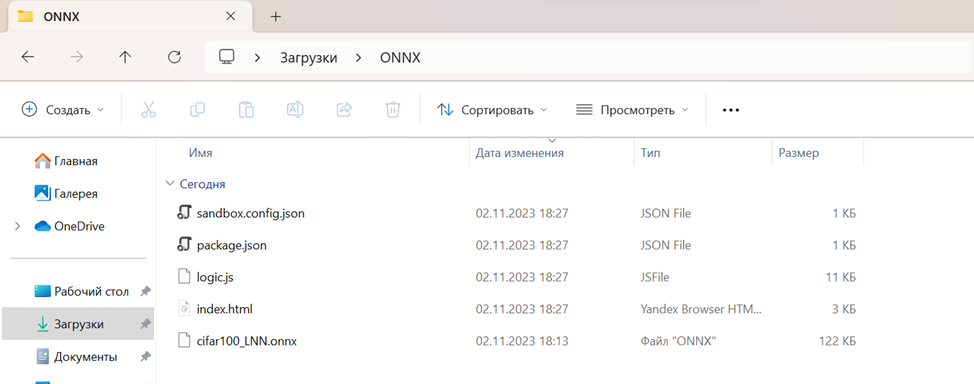


Теперь нужно перейти по ссылке<https://github.com/iu5git/MPPR> и скачать папку ONNX со всем ее содержимым.



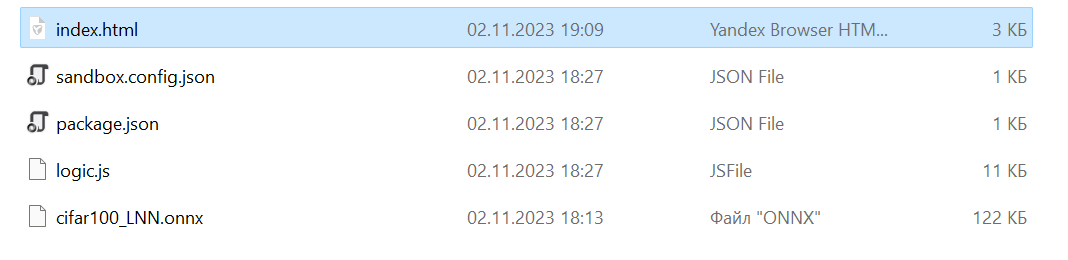


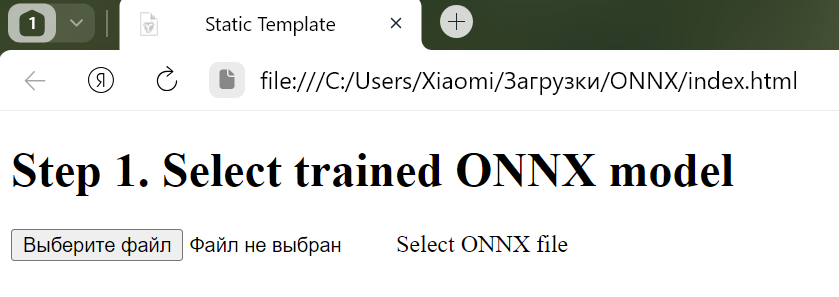
После того, как скачаете папку, нужно перенести ранее скачанный файл c расширением onnx в созданную папку. По итогу у вас должно получится данное содержимое.



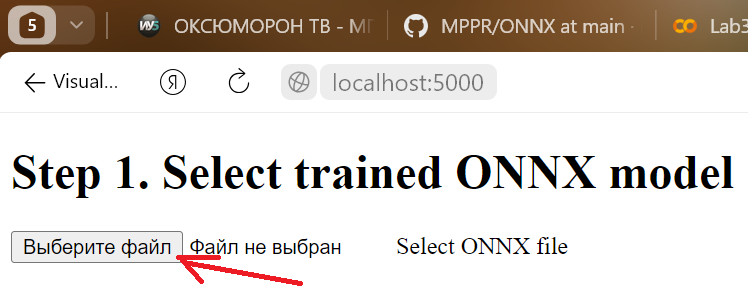
# Работа с моделью

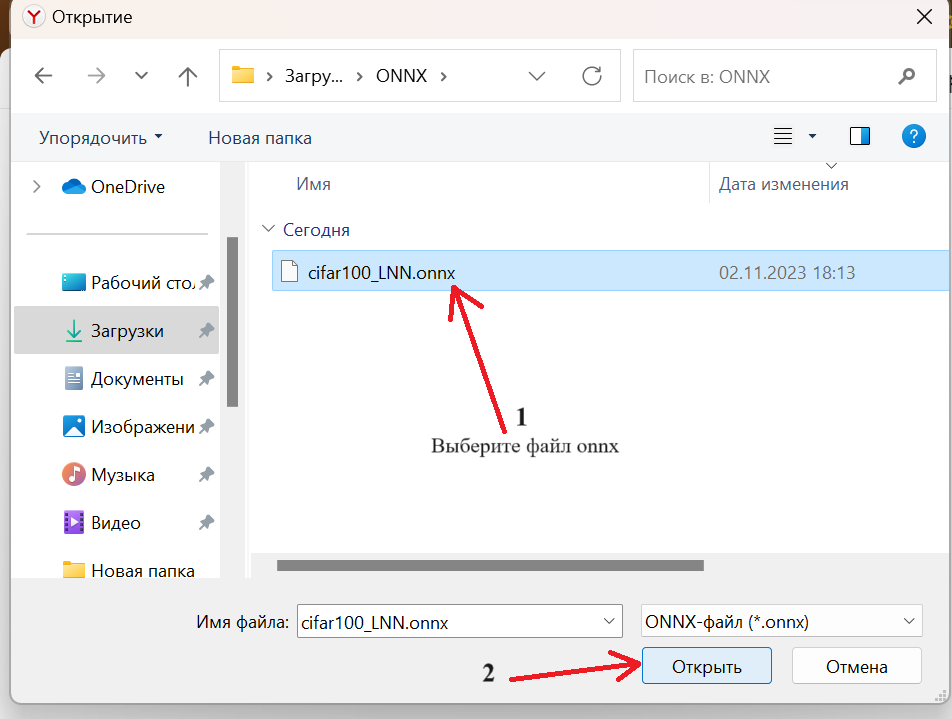
Теперь надо запустить html файл.

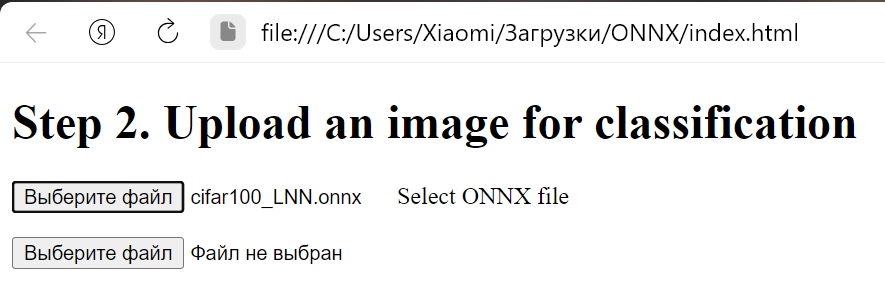




Далее нажимаем на кнопку «Выберите файл» и выбираем файл с расширением onnx.

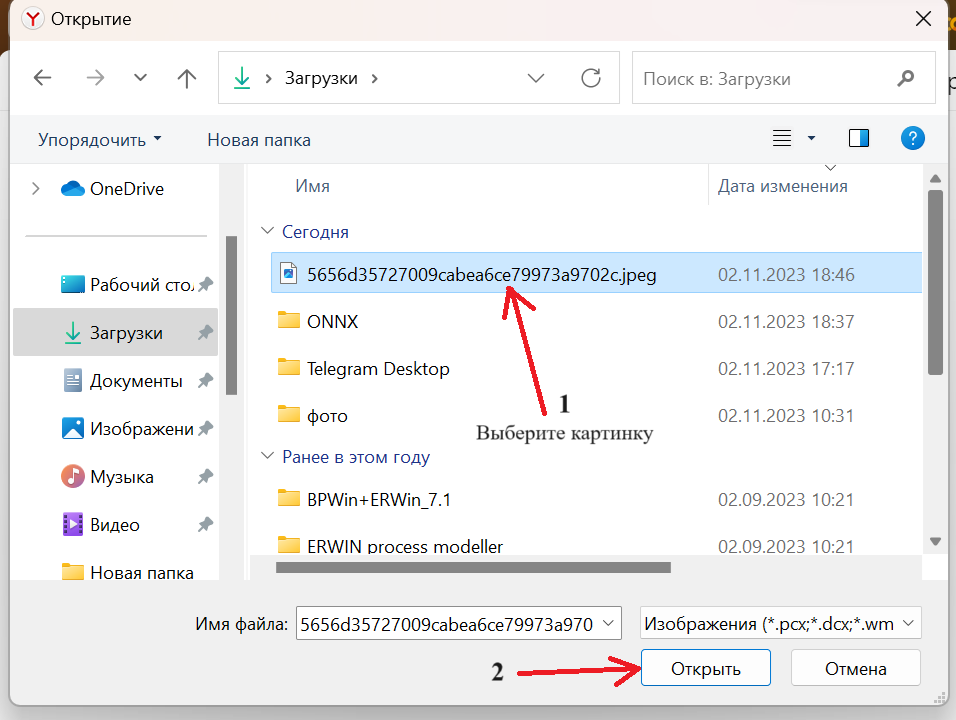


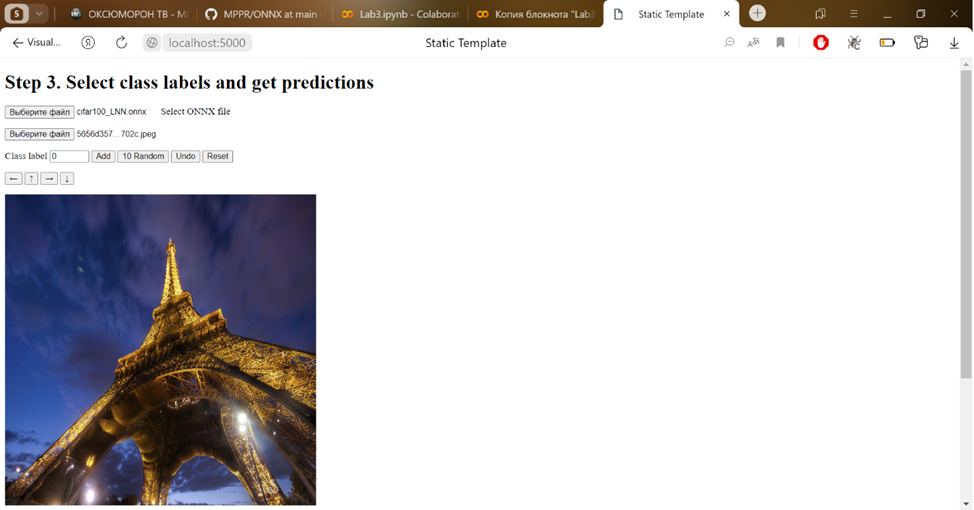




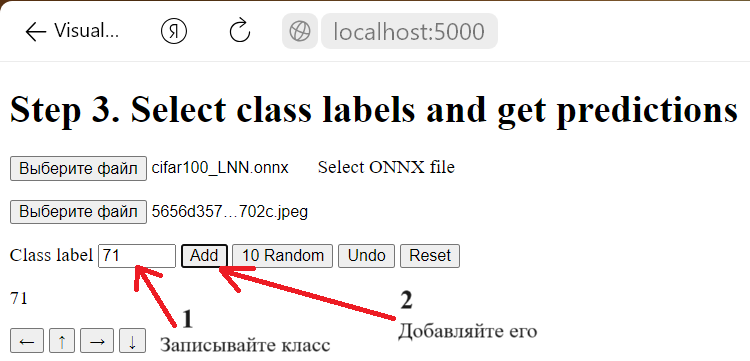
Теперь скачайте абсолютно любую фотографию из Интернета, потом нажмите на кнопку «Выберите файл» и выберите скачанную фотографию.







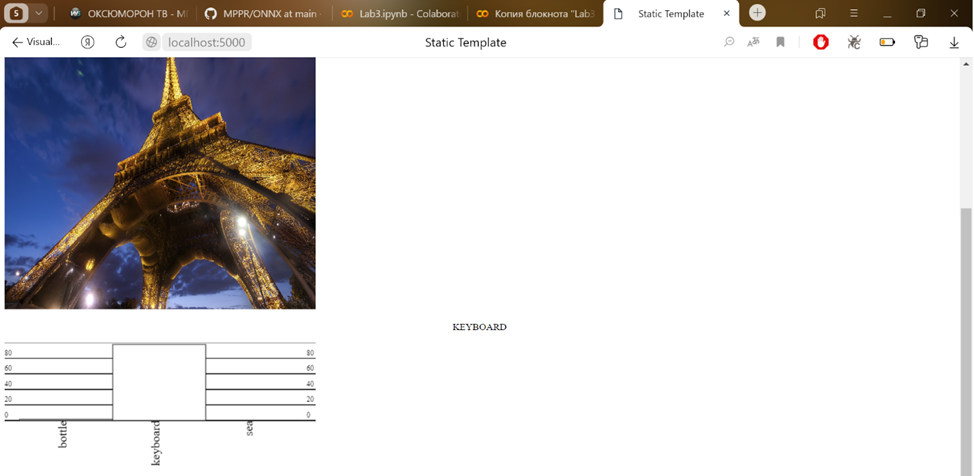
!!!Необходимо добавить классы по своему варианту, чтобы модель могла распознавать картинки.!!! В поле «Class label» вбивайте по очереди три класса и жмите на кнопку «Add», чтобы их добавить.



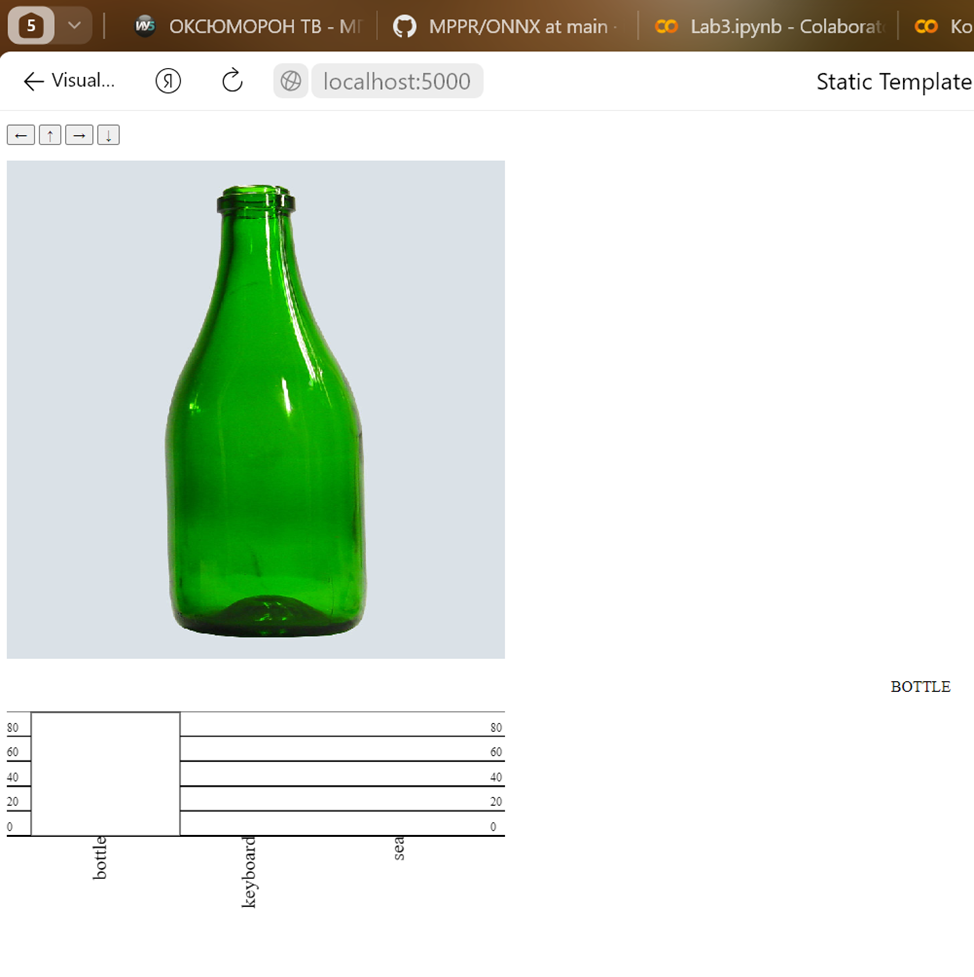


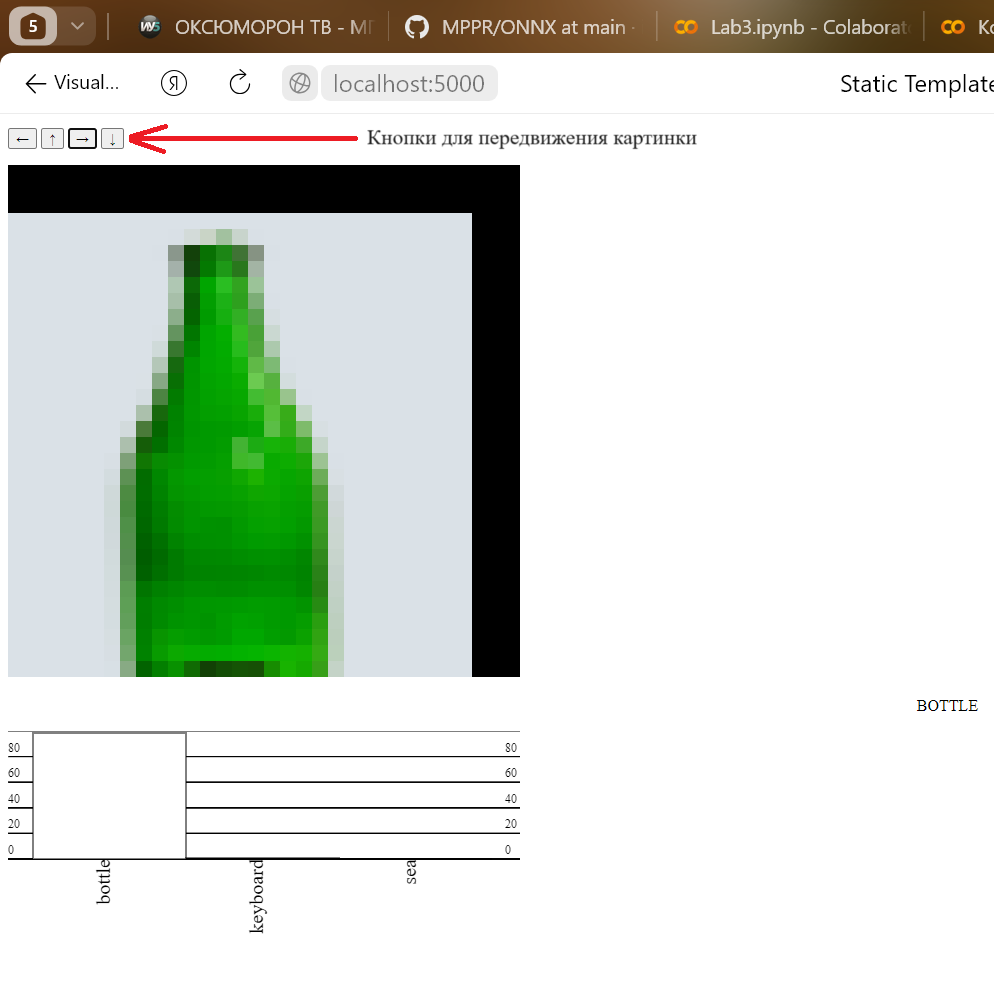


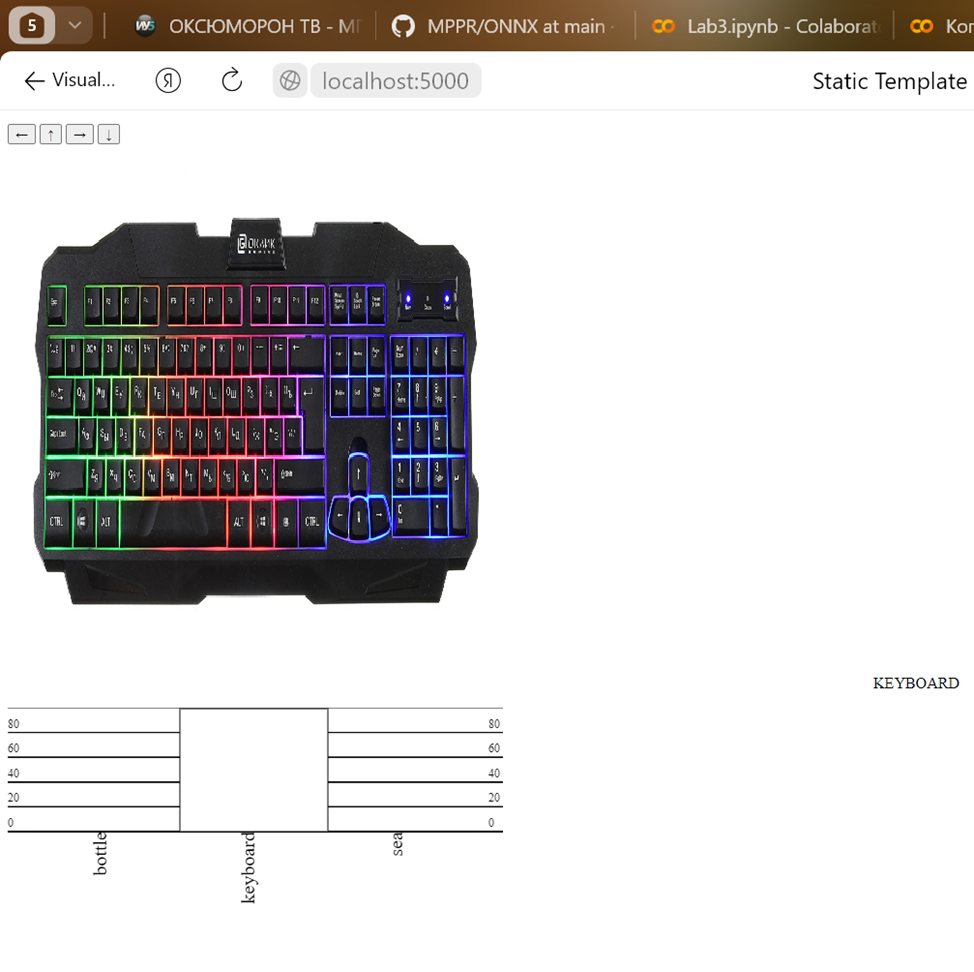
Теперь вы можете узнать, какие классы вам достались по варианту, пролистав страницу вниз.



!!!Далее вы скачиваете уже те фотографии, которые соответствуют вашим классам. В данном случае это бутылка, клавиатура и море и по очереди открываете их, как описано ранее. Помимо этого, вы также должны передвигать картинку, чтобы проверить, насколько хорошо ваша модель обучилась и может распознавать картинки в сдвиге. Двигать ее можно, нажимая на стрелки либо на своей клавиатуре, либо на кнопки стрелок, расположенных над картинкой.!!!

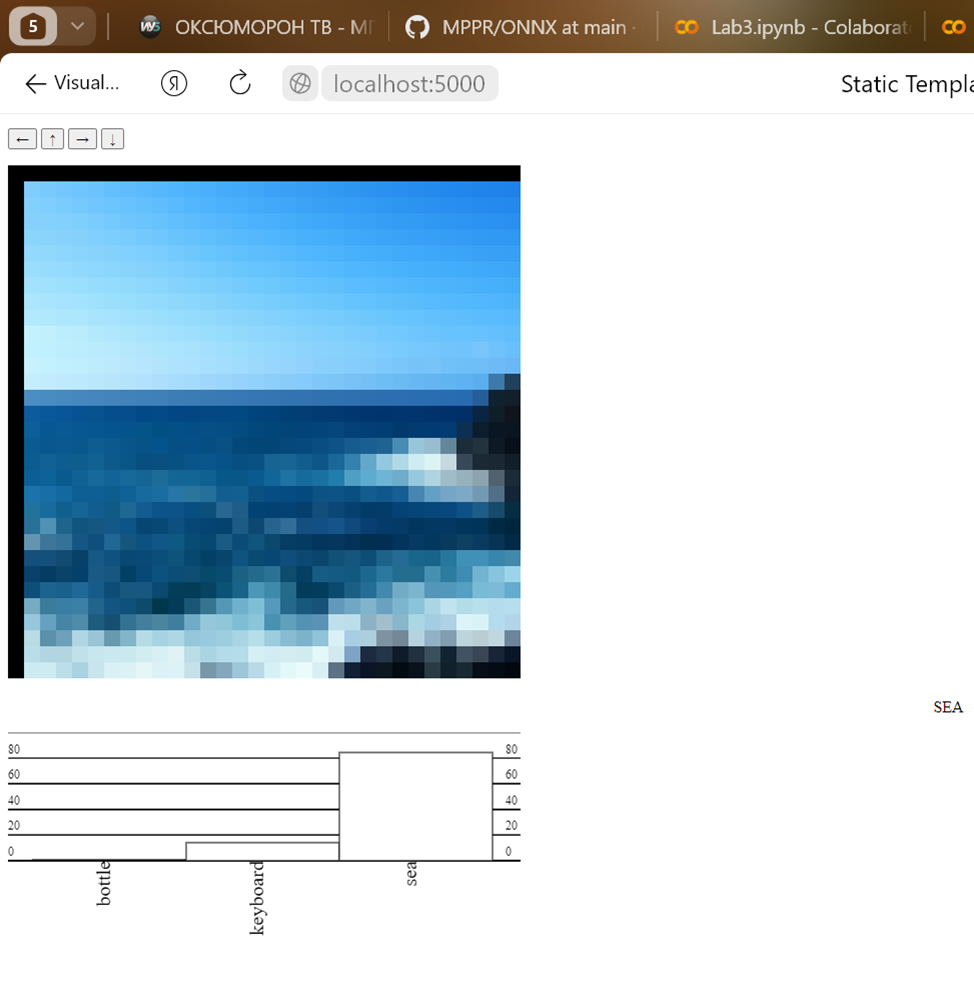










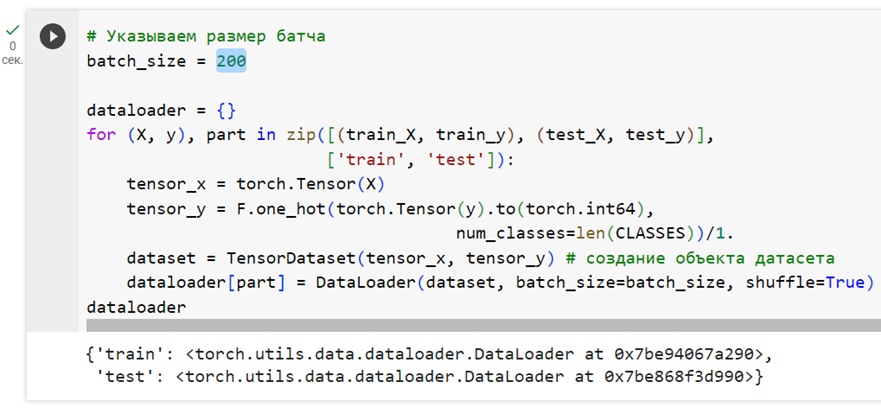


# Улучшение модели

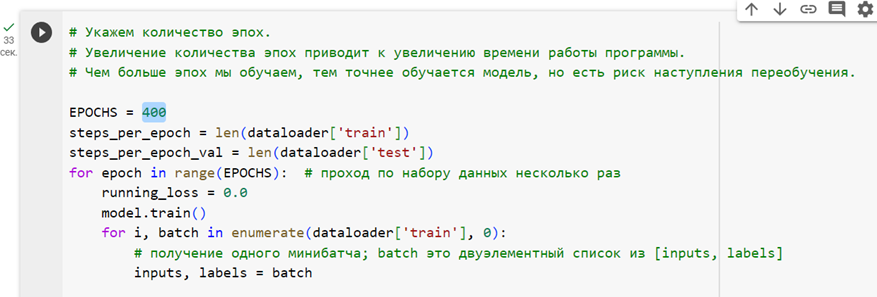
Теперь попробуем улучшить нашу модель. Для этого нужно изменить размер батча, уменьшить скорость обучения и увеличить общее количество итераций, а также поменять количество нейронов и слоев. Также создадим таблицу для учета результатов вариантов обучения.

Далее будет показано, где именно нужно менять конфигурации и параметры в коде. !!!После внесения изменений необходимо выполнять запуск кода с того момента, где вы вносили правки, т.е. не с самого начала.!!! Ниже будет приведена таблица с вариантами обучения.

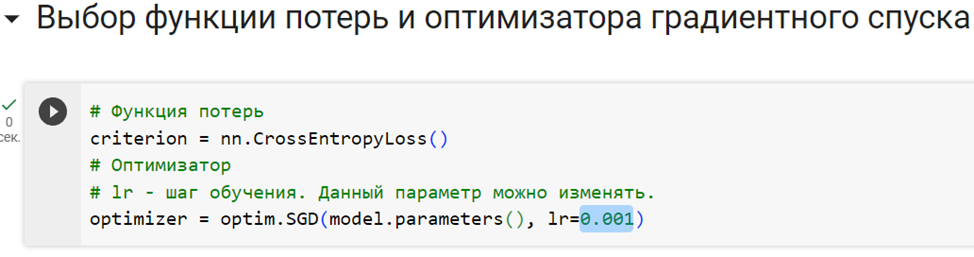
Изменение размера батча (batch\_size).



Изменение количества эпох (epochs).



Изменения шага (lr).



Добавление слоев.



**Таблица с результатами для вариантов обучения**

|  |  |  |  |
| --- | --- | --- | --- |
| **Конфигурация нейросети** | **Гиперпараметры** | **Точность** | **Комментарий** |
| FC(10), FC(3) | lr = 0,005, batch\_size = 128, epochs = 250 | test = 76%,  train = 99,2% | Базовый вариант |
| FC(10), FC(3) | lr = 0,005, batch\_size = 200, epochs = 400 | test = 77,3%,  train = 99,1% | Увеличили размер батча и количество эпох, тем самым сохранив общее количество итераций. Результаты слабо улучшились, хотя при том же количестве итераций обучаем на большем наборе, что дает возможность хорошего улучшения в теории. Возможно, проблема в переобучении модели. |
| FC(10), FC(3) | lr = 0,001, batch\_size = 200, epochs = 800 | test = 77,3%,  train = 91,6% | Уменьшили шаг (скорость) обучения и увеличили общее количество итераций для повышения точности. Результат на тестовой выборке не изменился, хотя все еще сильно хуже результата на обучающей выборке, а та в свою очередь ухудшилась. |
| FC(30), FC(10), FC(3) | lr = 0,005, batch\_size = 250, epochs = 135 | test = от 76% и до 79%,  train = 85,6% | Модель была изменена путем добавления еще одного скрытого слоя. Количество нейронов в скрытых слоях изменено на 30 и 10. Усложнив модель, добились снова незначительного прироста качества модели. Каждый раз обучая модель, она выдает разные, но хорошие значения метрики. Для получения более точных данных о качестве модели, возможно, стоит применить кросс-валидацию. Большинство значений гиперпараметров модели ведет к ее переобучению, и не удается получить результат лучше, чем этот, не прибегая к ресурсоемкому перебору их значений. |