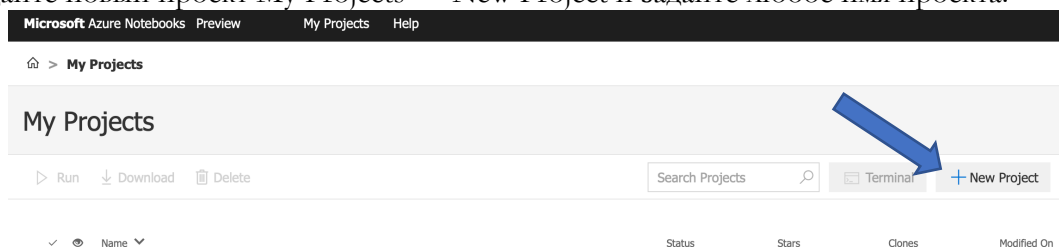
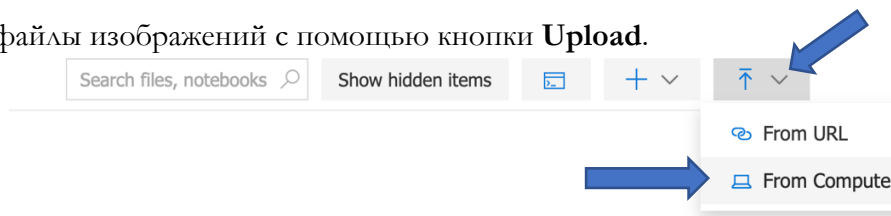


Настройка и создание облачного сервиса

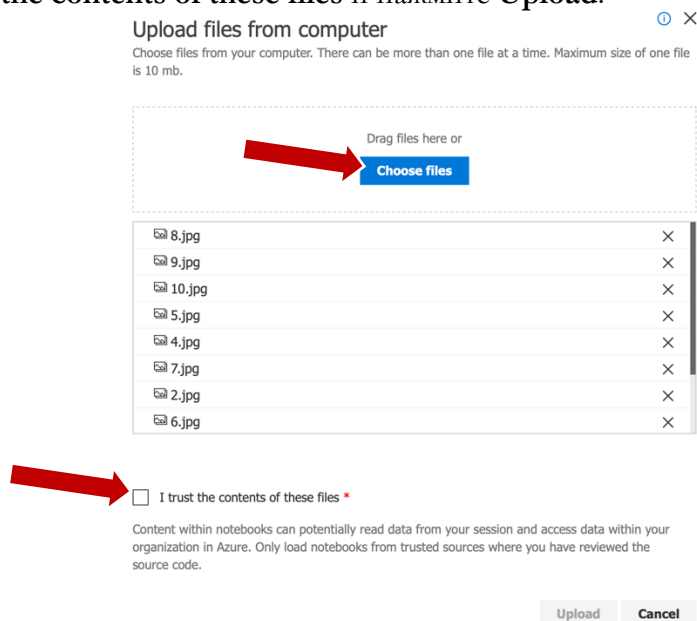
1. Перейдите по адресу: <https://notebooks.azure.com/>
2. Пройдите регистрацию или авторизуйтесь уже имеющимся аккаунтом (например, Office 365)
3. Создайте новый проект My Projects -> New Project и задайте любое имя проекта.



4. Загрузите файлы изображений с помощью кнопки **Upload**.



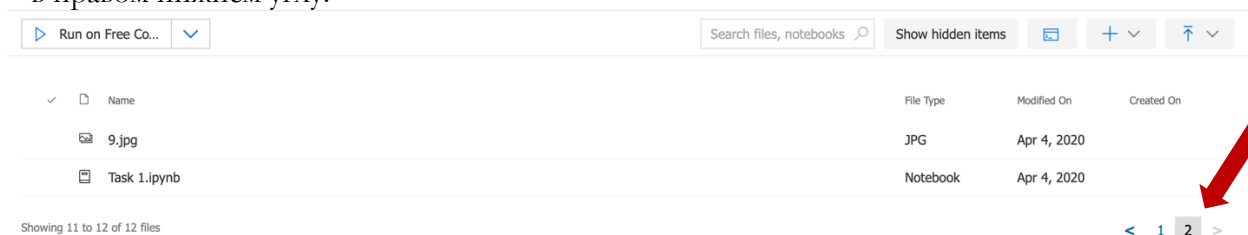
5. Выберите все изображения на вашем компьютере с помощью кнопки **Choose files** (скачайте изображения предварительно на компьютер из упражнения), далее поставьте галочку напротив пункта **I trust the contents of these files** и нажмите **Upload**.



6. После загрузки файлов нажмите **Done**.

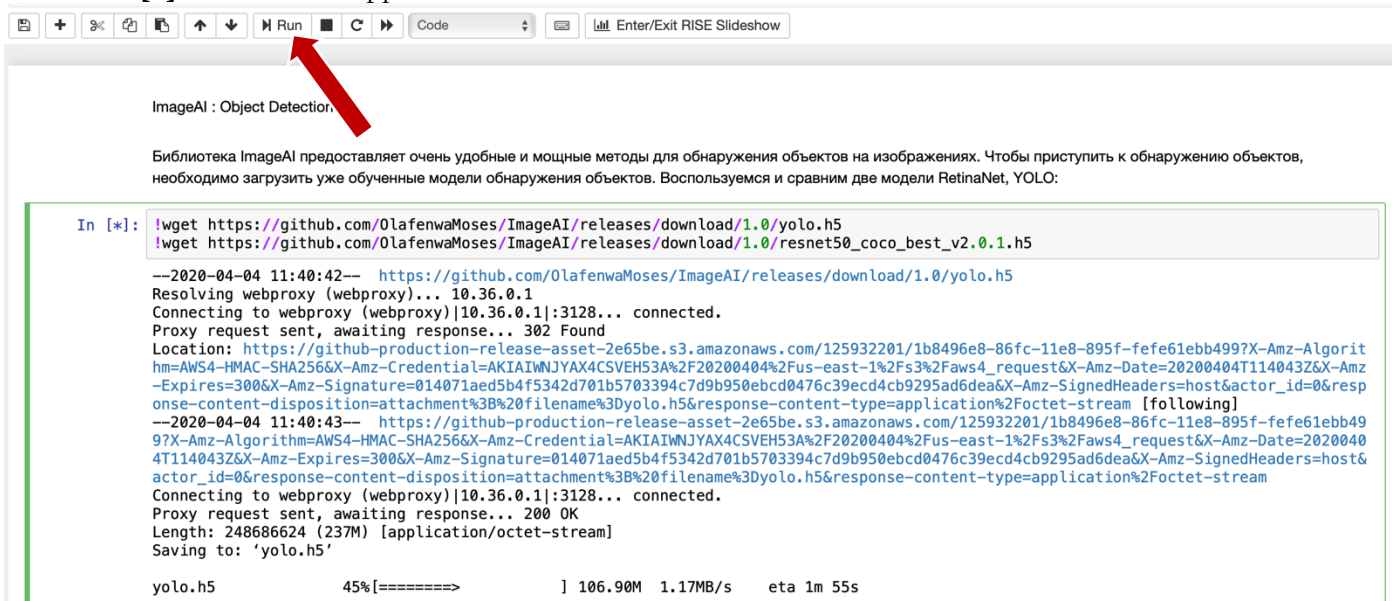


7. Аналогичным образом загрузите файл блокнота из упражнения.
8. Запустите блокнот, нажав по имени файла. Если файла нет в списке, воспользуйтесь навигацией в правом нижнем углу.



9. Далее следуйте указаниям в блокноте. Ниже показаны результаты выполнения команд.

10. Поместите курсор в первое поле **In** и нажмите **Run**, загрузка библиотек займет некоторое время, а пока можно приготовить чашечку чая. Загрузка и установка завершена, когда символ звездочки **In [*]** сменяется цифрой.



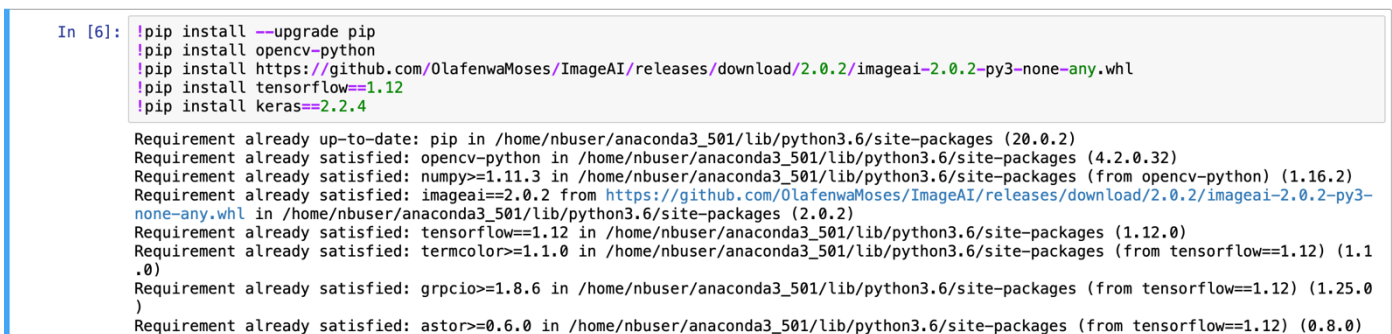
```
In [*]: !wget https://github.com/0lafenwaMoses/ImageAI/releases/download/1.0/yolo.h5
!wget https://github.com/0lafenwaMoses/ImageAI/releases/download/1.0/resnet50_coco_best_v2.0.1.h5

--2020-04-04 11:40:42-- https://github.com/0lafenwaMoses/ImageAI/releases/download/1.0/yolo.h5
Resolving webproxy (webproxy)... 10.36.0.1
Connecting to webproxy (webproxy)|10.36.0.1|:3128... connected.
Proxy request sent, awaiting response... 302 Found
Location: https://github-production-release-asset-2e65be.s3.amazonaws.com/125932201/1b8496e8-86fc-11e8-895f-fefe61ebb499?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20200404%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20200404T114043Z&X-Amz-Expires=300&X-Amz-Signature=014071aed5b4f5342d701b5703394c7d9b950ebcd0476c39ecd4cb9295ad6dea&X-Amz-SignedHeaders=host&actor_id=0&response-content-disposition=attachment%3B%20filename%3Dyolo.h5&response-content-type=application%2Foctet-stream [following]
--2020-04-04 11:40:43-- https://github-production-release-asset-2e65be.s3.amazonaws.com/125932201/1b8496e8-86fc-11e8-895f-fefe61ebb499?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20200404%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20200404T114043Z&X-Amz-Expires=300&X-Amz-Signature=014071aed5b4f5342d701b5703394c7d9b950ebcd0476c39ecd4cb9295ad6dea&X-Amz-SignedHeaders=host&actor_id=0&response-content-disposition=attachment%3B%20filename%3Dyolo.h5&response-content-type=application%2Foctet-stream
Connecting to webproxy (webproxy)|10.36.0.1|:3128... connected.
Proxy request sent, awaiting response... 200 OK
Length: 248686624 (237M) [application/octet-stream]
Saving to: 'yolo.h5'

yolo.h5          45%[=====>] 106.90M  1.17MB/s   eta 1m 55s
```

11. Поместите курсор в следующее поле **In** и нажмите **Run**, начнется установка недостающих библиотек:

Установим библиотеку компьютерного зрения — OpenCV, библиотеку ImageAI, которая позволяет выполнить обнаружение объектов на изображениях, а также библиотеку машинного обучения TensorFlow и нейросетевую библиотеку Keras конкретных версий, чтобы все заработало :)



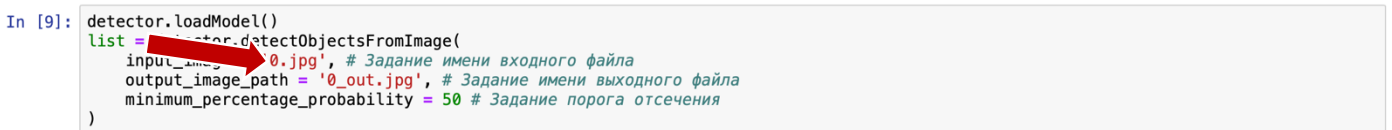
```
In [6]: !pip install --upgrade pip
!pip install opencv-python
!pip install https://github.com/0lafenwaMoses/ImageAI/releases/download/2.0.2/imageai-2.0.2-py3-none-any.whl
!pip install tensorflow==1.12
!pip install keras==2.2.4

Requirement already up-to-date: pip in /home/nbuser/anaconda3_501/lib/python3.6/site-packages (20.0.2)
Requirement already satisfied: opencv-python in /home/nbuser/anaconda3_501/lib/python3.6/site-packages (4.2.0.32)
Requirement already satisfied: numpy>=1.11.3 in /home/nbuser/anaconda3_501/lib/python3.6/site-packages (from opencv-python) (1.16.2)
Requirement already satisfied: imageai==2.0.2 from https://github.com/0lafenwaMoses/ImageAI/releases/download/2.0.2/imageai-2.0.2-py3-none-any.whl in /home/nbuser/anaconda3_501/lib/python3.6/site-packages (2.0.2)
Requirement already satisfied: tensorflow==1.12 in /home/nbuser/anaconda3_501/lib/python3.6/site-packages (1.12.0)
Requirement already satisfied: termcolor>=1.1.0 in /home/nbuser/anaconda3_501/lib/python3.6/site-packages (from tensorflow==1.12) (1.1.0)
Requirement already satisfied: grpcio>=1.8.6 in /home/nbuser/anaconda3_501/lib/python3.6/site-packages (from tensorflow==1.12) (1.25.0)
Requirement already satisfied: astor>=0.6.0 in /home/nbuser/anaconda3_501/lib/python3.6/site-packages (from tensorflow==1.12) (0.8.0)
```

12. Как только библиотеки будут установлены (строка «Successfully installed»), запустите следующие блоки **In** с кодом на Python.

13. Вам лишь необходимо изменять имя входного и выходного файла изображения, на соответствующие имена файлов, которые вы будете распознавать.

Выполняем распознавание объектов для изображения с именем 0.jpg и сохраняем результат в файл 0_out.jpg



```
In [9]: detector.loadModel()
list = detector.detectObjectsFromImage(
    input_image_path = '0.jpg', # Задание имени входного файла
    output_image_path = '0_out.jpg', # Задание имени выходного файла
    minimum_percentage_probability = 50 # Задание порога отсеивания
)
```

14. Запустите все ячейки с кодом с помощью **Run**. Через некоторое время появится распознанное изображение.

```
In [8]: detector = ObjectDetection()
detector.setModelTypeAsRetinaNet() #Для модели YOLO заменить на detector.setModelTypeAsYOLOv3()
detector.setModelPath('resnet50_coco_best_v2.0.1.h5') #Для модели YOLO заменить на detector.setModelPath("yolo.h5")
detector.loadModel()
```

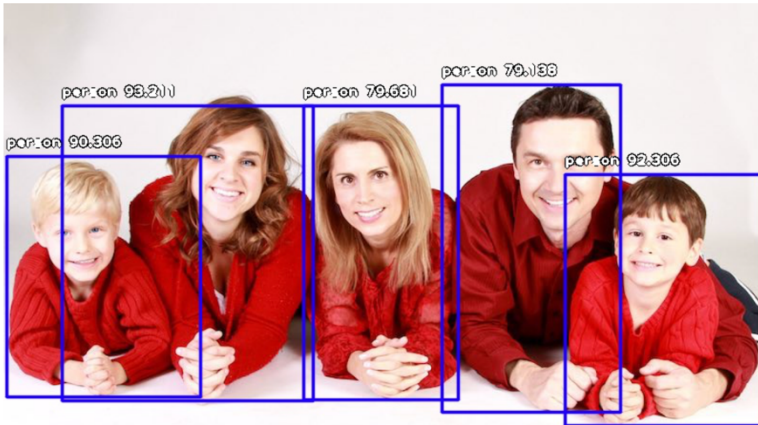
Выполняем распознавание объектов для изображения с именем 0.jpg и сохраняем результат в файл 0_out.jpg

```
In [9]: detector.loadModel()
list = detector.detectObjectsFromImage(
    input_image = '0.jpg', # Задание имени входного файла
    output_image_path = '0_out.jpg', # Задание имени выходного файла
    minimum_percentage_probability = 50 # Задание порога отсеечения
)
```

Изображение можно посмотреть в папке проекта, либо вывести на экран в блокноте:

```
In [10]: Image(filename='0_out.jpg')
```

Out[10]:



15. Для распознавания других изображений, достаточно изменить имена файлов и запускать ячейку:

```
In [9]: detector.loadModel()
list = detector.detectObjectsFromImage(
    input_image = '2.jpg', # Задание имени входного файла
    output_image_path = '2_out.jpg', # Задание имени выходного файла
    minimum_percentage_probability = 50 # Задание порога отсеечения
)
```

16. Для изменения модели следует изменить две строки кода на те, что указаны в комментариях в следующей ячейке:

```
In [8]: detector = ObjectDetection()
detector.setModelTypeAsRetinaNet() #Для модели YOLO заменить на detector.setModelTypeAsYOLOv3()
detector.setModelPath('resnet50_coco_best_v2.0.1.h5') #Для модели YOLO заменить на detector.setModelPath("yolo.h5")
detector.loadModel()
```