



# Установка приложения на телефон



# Этапы работы над проектом



Идея продукта



Планирование



Разработка прототипа



Доработка



Тестирование



Публикация

Обсуждение: результаты тестирования



# Установка приложения на телефон

Мобильные телефоны на операционной системы Android позволяют устанавливать приложения на телефон или планшет в формате файлов с расширением **.apk**.



# Установка на телефон

Чтобы превратить файл .py с кодом на Kivy в приложение в формате .apk существует специальная библиотека — `buildozer`.

Но эта библиотека не адаптирована к работе на операционной системе Windows. Если компьютера с Linux-подобной операционной системой нет под рукой — возникает проблема.



# Установка на телефон

Для решения такого рода проблем созданы **виртуальные машины**.

**Виртуальная машина** — программная система, имитирующая функционал операционной системы внутри другой операционной системы.

Устанавливать такую «виртуальную машину» на личный компьютер и управлять ею — сложный и трудоёмкий процесс.

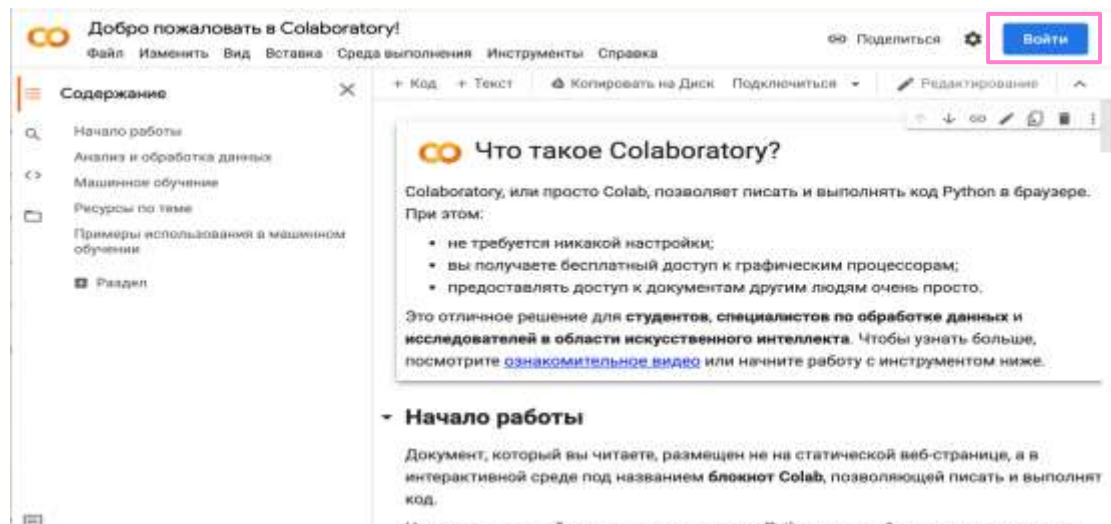
Для реализации нашей идеи существует более лёгкий и удобный онлайн-инструмент для запуска кода из браузера на Linux-подобной системе — **Google Colab**.



# Google Colab

Заходим на сайт: <https://colab.research.google.com/>

Нажимаем на кнопку войти и входим через свой аккаунт Google.

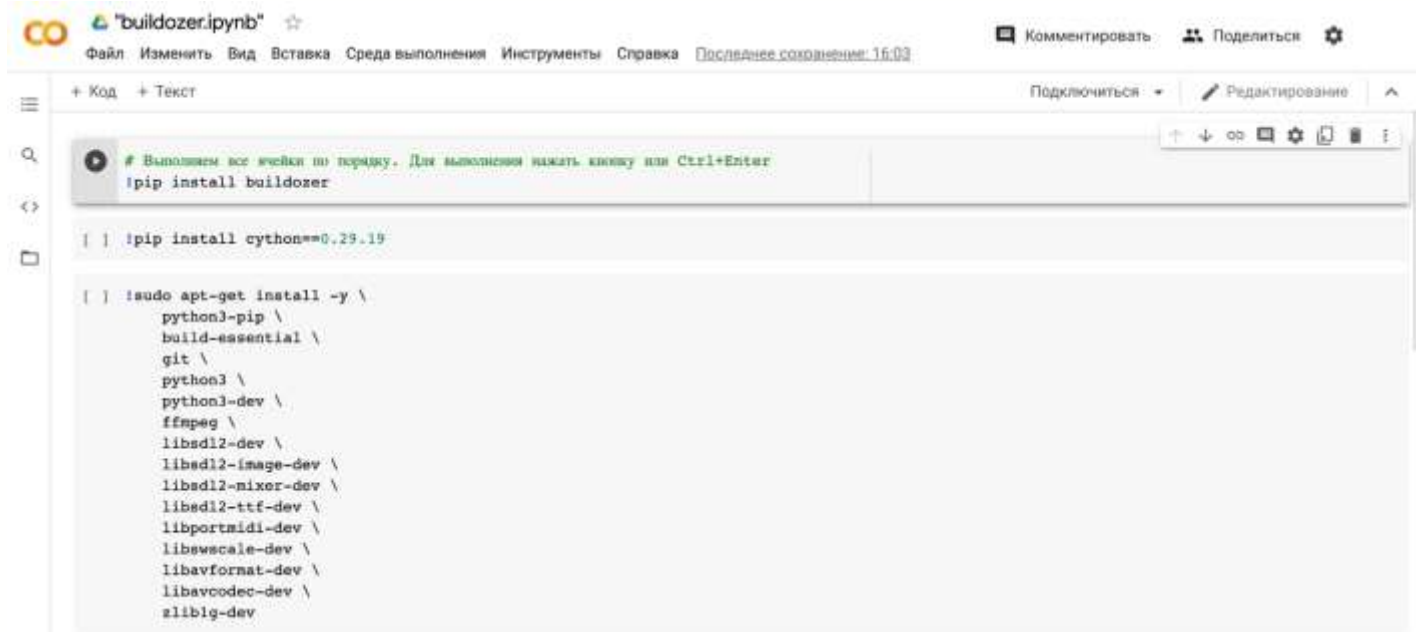


Установка на телефон



# Google Colab

Переходим по [ссылке](#)



The screenshot shows a Google Colab notebook interface. At the top, the title is "buildozer.ipynb" with a star icon. Below the title is a menu bar with options: "Файл", "Изменить", "Вид", "Вставка", "Среда выполнения", "Инструменты", "Справка", and "Последнее сохранение: 16:03". To the right of the menu bar are buttons for "Комментировать", "Поделиться", and a settings gear icon. Below the menu bar is a toolbar with "+ Код" and "+ Текст" buttons. The main area of the notebook contains three code cells. The first cell has a comment in Russian: "# Выполняем все ячейки по порядку. Для выполнения нажать кнопку или Ctrl+Enter" followed by the command "pip install buildozer". The second cell contains the command "pip install cython==0.29.19". The third cell contains a list of system packages to be installed using "sudo apt-get install -y \", including python3-pip, build-essential, git, python3, python3-dev, ffmpeg, libSDL2-dev, libSDL2-image-dev, libSDL2-mixer-dev, libSDL2-ttf-dev, libportmidi-dev, libswscale-dev, libavformat-dev, libavcodec-dev, and zlib1g-dev.

```
# Выполняем все ячейки по порядку. Для выполнения нажать кнопку или Ctrl+Enter
pip install buildozer

[ ] !pip install cython==0.29.19

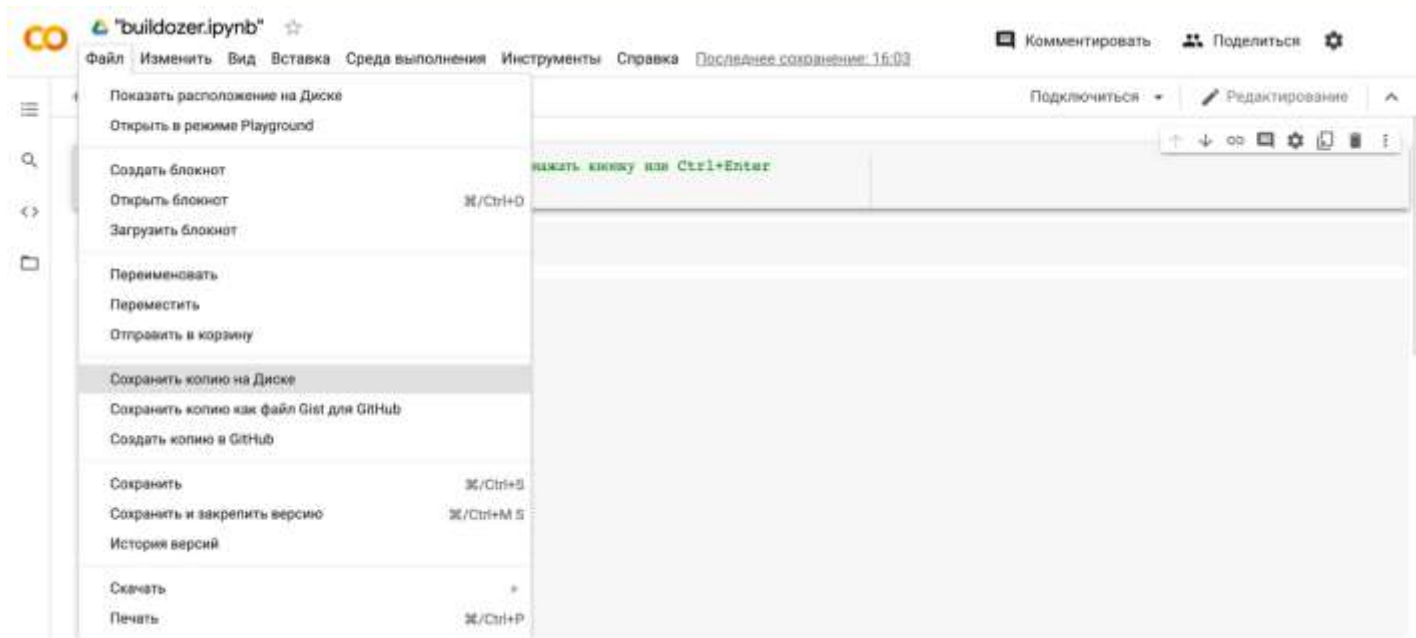
[ ] !sudo apt-get install -y \
    python3-pip \
    build-essential \
    git \
    python3 \
    python3-dev \
    ffmpeg \
    libSDL2-dev \
    libSDL2-image-dev \
    libSDL2-mixer-dev \
    libSDL2-ttf-dev \
    libportmidi-dev \
    libswscale-dev \
    libavformat-dev \
    libavcodec-dev \
    zlib1g-dev
```

Установка на телефон



# Google Colab

Копируем файл себе на диск: **Файл** **Сохранить копию на диске**.



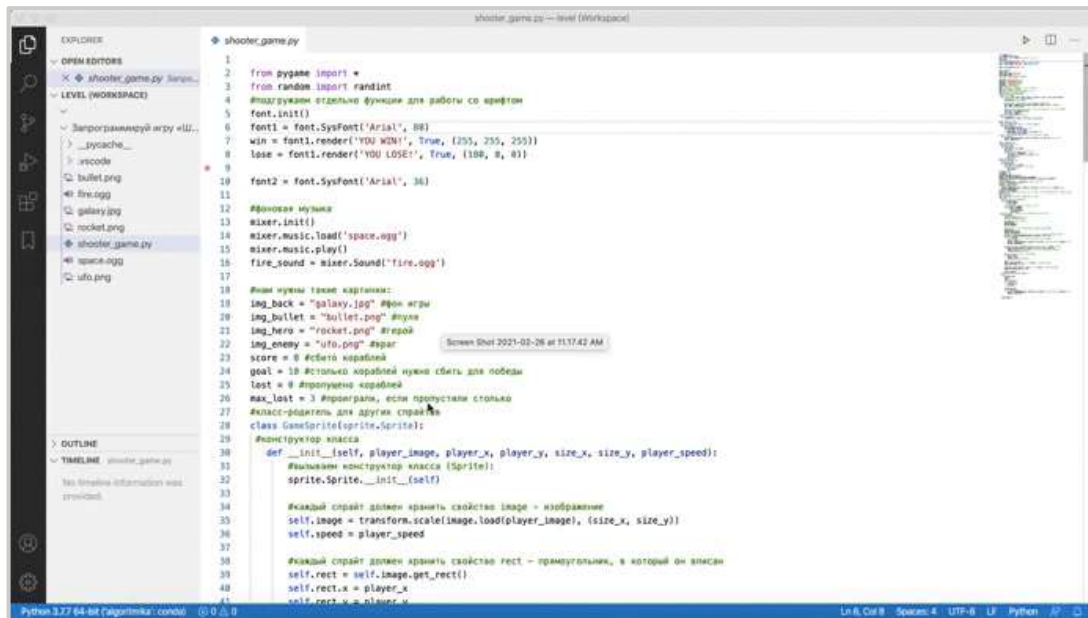
Установка на телефон





# Google Colab

1. Кликните правой кнопкой мыши по любому файлу в папке проекта и выберите **«Показать в Проводнике»/Reveal in Finder**.



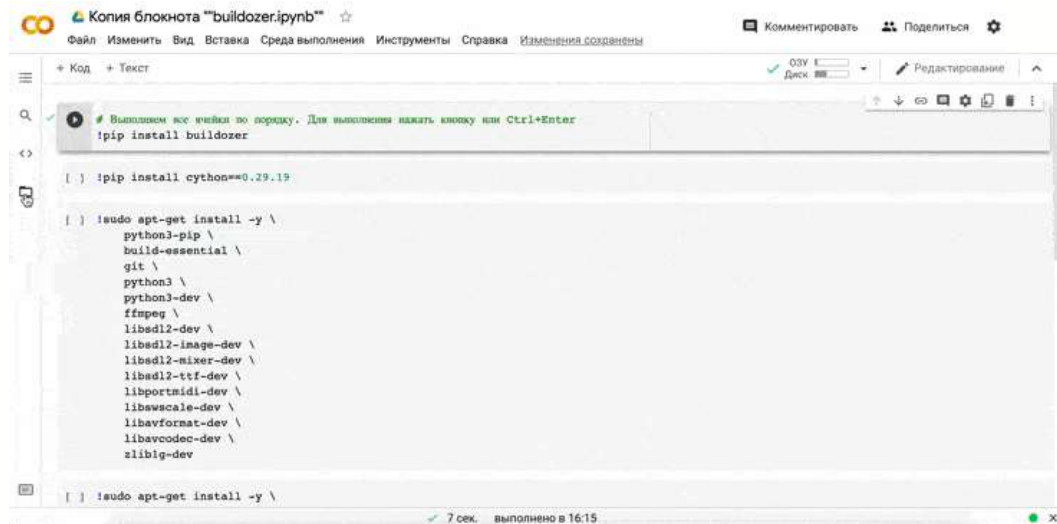
```
1
2 from pygame import *
3 from random import randint
4 #Инициализируем отдельные функции для работы со шрифтом
5 font.init()
6 font1 = Font.SysFont('Arial', 88)
7 win = font1.render('YOU WON!', True, (255, 255, 255))
8 lose = font1.render('YOU LOSE!', True, (188, 8, 0))
9
10 font2 = Font.SysFont('Arial', 36)
11
12 #Инициализируем музыку
13 mixer.init()
14 mixer.music.load('space.ogg')
15 mixer.music.play()
16 fire_sound = mixer.Sound('fire.ogg')
17
18 #Имя и путь к файлу картинок:
19 img_back = "galaxy.jpg" #Фон игры
20 img_bullet = "bullet.png" #пуля
21 img_hero = "rocket.png" #космолет
22 img_enemy = "ufo.png" #враг
23 score = 0 #счет очков
24 goal = 10 #столько кораблей нужно сбить для победы
25 lost = 0 #пропущено кораблей
26 max_lost = 3 #программа, если пропустила столько
27 #класс-родитель для других спрайтов
28 class GameSprite(sprite.Sprite):
29     #конструктор класса
30     def __init__(self, player_image, player_x, player_y, size_x, size_y, player_speed):
31         #вызываем конструктор класса (Sprite):
32         sprite.Sprite.__init__(self)
33
34         #Каждый спрайт должен хранить свойство image - изображение
35         self.image = transform.scale(image.load(player_image), (size_x, size_y))
36         self.speed = player_speed
37
38         #Каждый спрайт должен хранить свойство rect - прямоугольник, в который он вписан
39         self.rect = self.image.get_rect()
40         self.rect.x = player_x
41         self.rect.y = player_y
```

Установка на телефон



# Google Colab

2. Скопируйте файлы проекта и разместите их в новой удобно расположенной папке компьютера.
3. Добавьте файлы в Google Colab.



The screenshot shows a Google Colab notebook interface. At the top, the title is "Копия блокнота 'buildozer.ipynb'". Below the title bar, there are tabs for "Файл", "Изменить Вид", "Вставка", "Среда выполнения", "Инструменты", "Справка", and "Изменения сохранены". The main area contains a code cell with the following text:

```
!pip install buildozer

[ ] !pip install cython==0.29.19

[ ] !sudo apt-get install -y \
    python3-pip \
    build-essential \
    git \
    python3 \
    python3-dev \
    ffmpeg \
    libSDL2-dev \
    libSDL2-image-dev \
    libSDL2-mixer-dev \
    libSDL2-ttf-dev \
    libportmidi-dev \
    libswscale-dev \
    libavformat-dev \
    libavcodec-dev \
    zlib1g-dev

[ ] !sudo apt-get install -y \
```

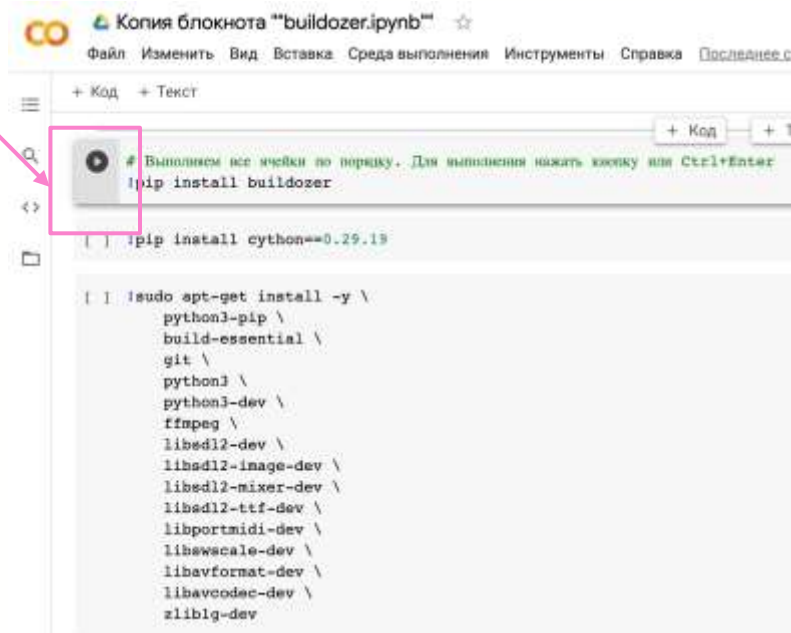
At the bottom of the code cell, a status bar indicates "7 сек. выполнено в 16:15".

Установка на телефон



# Google Colab

4. Запускайте по очереди блоки кода.



```
# Выполняем все ячейки по порядку. Для выполнения нажать кнопку или Ctrl+Enter
!pip install buildozer

[ ] !pip install cython==0.29.19

[ ] !sudo apt-get install -y \
    python3-pip \
    build-essential \
    git \
    python3 \
    python3-dev \
    ffmpeg \
    libSDL2-dev \
    libSDL2-image-dev \
    libSDL2-mixer-dev \
    libSDL2-ttf-dev \
    libportmidi-dev \
    libawescale-dev \
    libavformat-dev \
    libavcodec-dev \
    zlib1g-dev
```

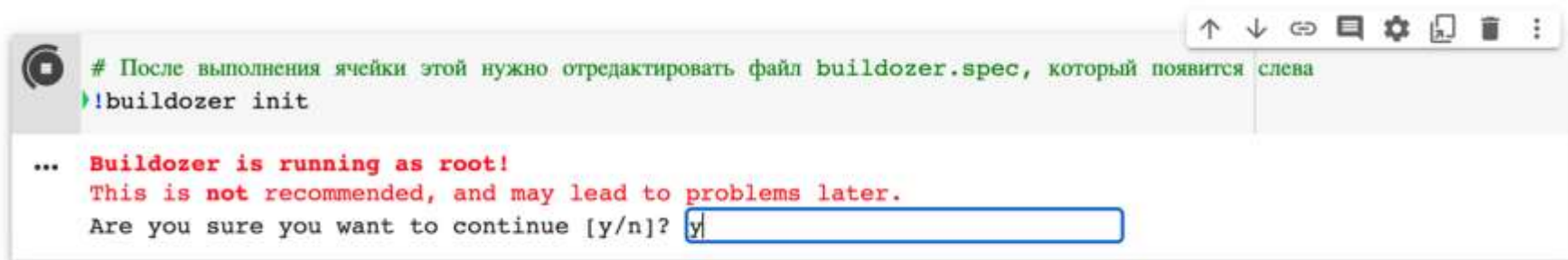
Установка на телефон



# Google Colab

На выполнении предпоследней ячейки кода в командной строке появится вопрос: «Вы уверены, что хотите продолжить?».

Необходимо ввести английскую букву **y** в командную строку и нажать Enter.



The screenshot shows a terminal window with a toolbar at the top containing icons for navigation, search, settings, and other functions. The terminal content is as follows:

```
# После выполнения ячейки этой нужно отредактировать файл buildozer.spec, который появится слева
!buildozer init

... Buildozer is running as root!
This is not recommended, and may lead to problems later.
Are you sure you want to continue [y/n]? y
```



# Google Colab

После этого в строке меню справа появится файл **buildozer.spec**.

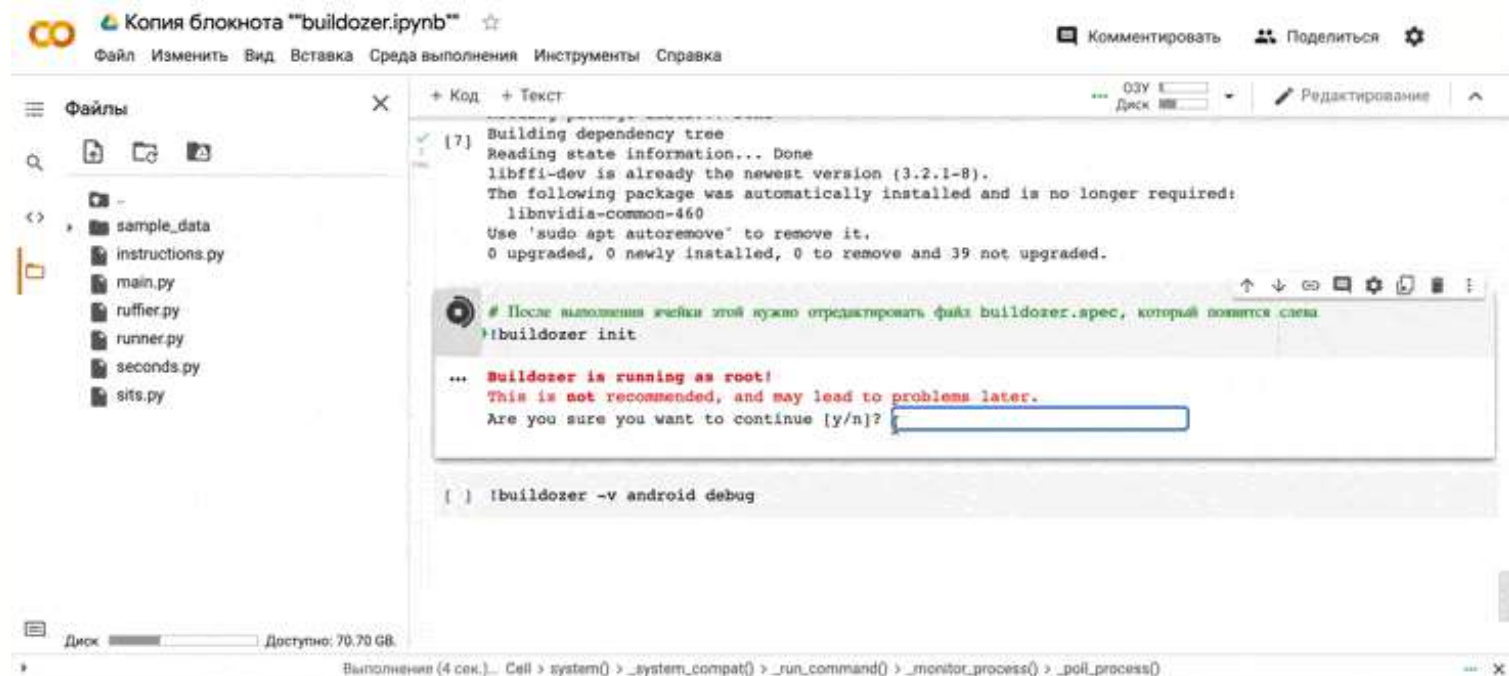
Если файл не появляется, но ячейка закончила свою работу, нажмите правой кнопкой мыши на область с файлами и выберите пункт «Обновить».

Откройте файл **buildozer.spec** и измените строки 4 и 7. Имя приложению выберите самостоятельно.

```
# (str) Title of your application  
title = Ruffier Test  
  
# (str) Package name  
package.name = ruffiertest
```



# Google Colab



Копия блокнота "buildozer.ipynb"

Файл Изменить Вид Вставка Среда выполнения Инструменты Справка

Комментировать Поделиться

ОЗУ Диск Редактирование

Файлы

- 
- sample\_data
- instructions.py
- main.py
- ruffier.py
- runner.py
- seconds.py
- sits.py

```
[7] Building dependency tree
Reading state information... Done
libffi-dev is already the newest version (3.2.1-8).
The following package was automatically installed and is no longer required:
  libnvidia-common-460
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 39 not upgraded.

# После выполнения ячейки этой нужно отредактировать файл buildozer.spec, который появится слева
$buildozer init

... Buildozer is running as root!
This is not recommended, and may lead to problems later.
Are you sure you want to continue [y/n]? 

[ ] $buildozer -v android debug
```

Диск Доступно: 70.70 GB

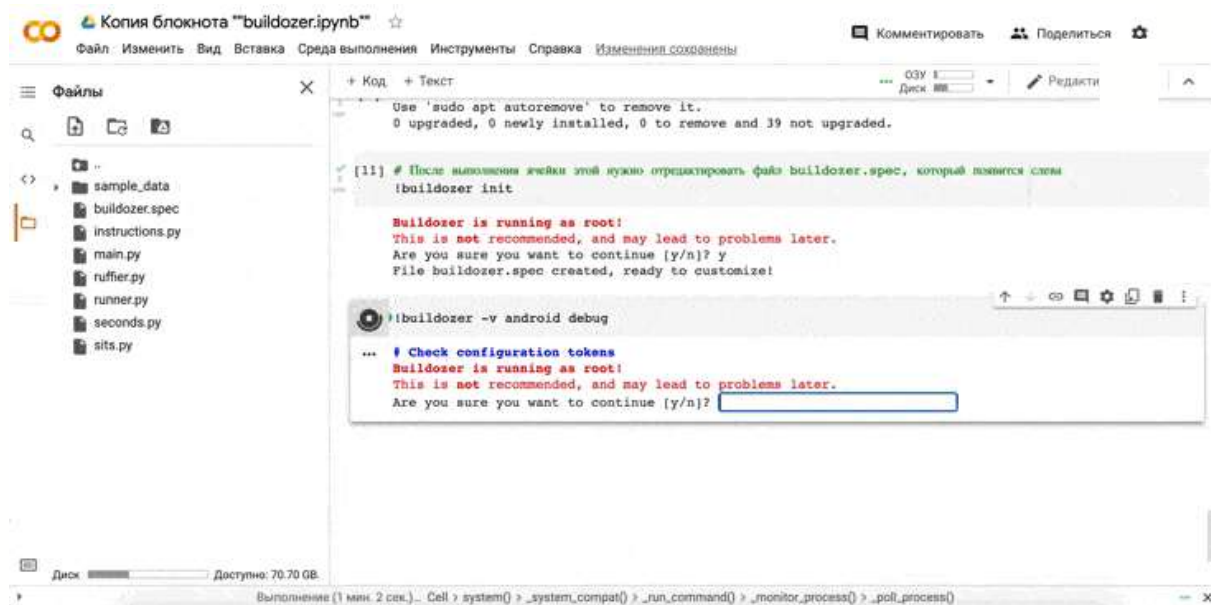
Выполнение (4 сек.)... Cell > system() > \_system\_compat() > \_run\_command() > \_monitor\_process() > \_poll\_process()

Установка на телефон



# Google Colab

Далее запускайте последнюю ячейку кода, на вопросы-подтверждения необходимо ввести английскую букву **y** в командную строку и нажать Enter.



```
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 39 not upgraded.

[11] # После выполнения ячейки этой нужно отредактировать файл buildozer.spec, который появится слева
!buildozer init

Buildozer is running as root!
This is not recommended, and may lead to problems later.
Are you sure you want to continue [y/n]? y
File buildozer.spec created, ready to customize!

!buildozer -v android debug

... # Check configuration tokens
Buildozer is running as root!
This is not recommended, and may lead to problems later.
Are you sure you want to continue [y/n]? 
```

Установка на телефон



# Google Colab

Выполнение последней ячейки кода может занять в первый раз порядка 20 минут. Далее при внесении правок в файлы и повторной сборке приложения этот процесс будет происходить значительно быстрее.

После этого в списке файлов слева появится папка **bin**, внутри которой будет находиться файл **.apk**.





# Google Colab

```
[DEBUG]: Deprecated Gradle features were used in this build, making it incompatible with Gradle
[DEBUG]: Use '--warning-mode all' to show the individual deprecation warnings.
[DEBUG]: See https://docs.gradle.org/6.4.1/userguide/command_line_interface.html#sec:command_line
[DEBUG]:
[DEBUG]: BUILD SUCCESSFUL in 43s
[DEBUG]: 27 actionable tasks: 27 executed
[DEBUG]:
[DEBUG]: <-----> 0% WAITING> IDLE> IDLE
[INFO]: <- directory context /content/.buildozer/android/platform/python-for-android
[INFO]: Of the existing distributions, the following meet the given requirements:
[INFO]:   ruffiertest: min API 21, includes recipes (hostpython3, libffi, openssl, sdl2_image, s
[INFO]: ruffiertest has compatible recipes, using this one
[INFO]: # Copying android package to current directory
[INFO]: # Android package filename not found in build output. Guessing...
[INFO]: # Found android package file: /content/.buildozer/android/platform/build-armeabi-v7a/dists/
[INFO]: # Add version number to android package
[INFO]: # Android package renamed to ruffiertest__armeabi-v7a-debug-0.1-.apk
[DEBUG]: -> running cp /content/.buildozer/android/platform/build-armeabi-v7a/dists/ruffiertest_arm
WARNING: Received a --sdk argument, but this argument is deprecated and does nothing.
No setup.py/pyproject.toml used, copying full private data into .apk.
Applying Java source code patches...
Applying patch: src/patches/SDLActivity.java.patch
# Android packaging done!
# APK ruffiertest-0.1-armeabi-v7a-debug.apk available in the bin directory
```

Установка на телефон



# Установка на телефон

Этот файл необходимо загрузить на телефон любым удобным способом. Например, отправить через мессенджер или подключить телефон к компьютеру с помощью провода.

Далее на телефоне открываем файл и устанавливаем его.

Телефон несколько раз предупредит об опасности скачивания и установки неизвестных приложений. Поскольку вы точно уверены в том, что ваше приложение безопасно, разрешайте скачивание и установку приложения.

Дождитесь окончания установки и начинайте тестировать!

