**Kivy**

**Быстрый старт**

<https://kivy.org/doc/stable/gettingstarted/>

Комментарий

1. Перевод Getting Started <https://kivy.org/doc/stable/gettingstarted/>

2. Перевод - адаптивный

3. все ссылки – машинный Переводб кроме раздела **«**Содержание», там прямые на разделы сайта

**Содержание**

1. Введение ([Introduction](https://kivy.org/doc/stable/gettingstarted/intro.html))

2. Инсталляция ([Installation](https://kivy.org/doc/stable/gettingstarted/installation.html))

[A first App](https://kivy.org/doc/stable/gettingstarted/first_app.html)

[Properties](https://kivy.org/doc/stable/gettingstarted/properties.html)

[Kv Design Language](https://kivy.org/doc/stable/gettingstarted/rules.html)

[Events](https://kivy.org/doc/stable/gettingstarted/events.html)

[Non-widget stuff](https://kivy.org/doc/stable/gettingstarted/framework.html)

[Layouts](https://kivy.org/doc/stable/gettingstarted/layouts.html)

[Drawing](https://kivy.org/doc/stable/gettingstarted/drawing.html)

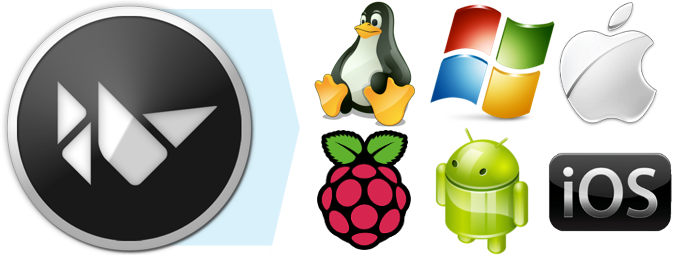
[Packaging](https://kivy.org/doc/stable/gettingstarted/packaging.html)

[Examples](https://kivy.org/doc/stable/gettingstarted/examples.html)

[Diving in](https://kivy.org/doc/stable/gettingstarted/diving.html)

**1. Введение (**[**Introduction**](https://kivy.org/doc/stable/gettingstarted/intro.html)**)**

**Kivy** – кроссплатформенный фреймворк с открытым исходным кодом для разработки приложений, использующих сенсорные пользовательские интерфейсы, в том числе и такие, как multi-touch.

[](https://kivy.org/doc/stable/_images/gs-introduction.png)

Это руководство - обзор, который познакомит с основными концепциями и необходимыми инструментами фреймворка **Kivy**. Он даёт достаточно информации для начала работы, а также ссылки на примеры разработок и соответствующие статьи с более подробной информацией.

Один и тот же код, написанный на языке программирования **Python** будет запускаться на различных платформах. Приложения, созданные на **Python & Kivy** используют:

* Настольные компьютеры: **Windows, Linux, OS X**.
* Смартфоны, планшеты и пр. гаджеты под управлением **Android, iOS и Raspberry**.
* Cенсорные профессиональные и бытовые приборы, поддерживающие **TUIO** (Tangible User Interface Objects - устройства с «осязаемым» пользовательским интерфейсом).

Kivy может использовать большинство вводов, протоколов и устройств, включая мультисенсорный симулятор мыши, WM\_Touch, WM\_Pen, Mac OS X Trackpad и Magic Mouse, Mtdev, Linux Kernel HID, TUIO.

Графический движок построен на OpenGL ES 2 с использованием конвейера.

Инструментарий включает в себя более 20 виджетов, все они легко расширяются. Многие части написаны на языке программирования **C** с использованием **Cython**.

Kivy распространяется под лицензией MIT полностью бесплатен для коммерческих разработoк.

Фреймворк стабилен, поддерживается широким сообществом, имеет хорошо документированный API и Руководство по программированию.

**2. Инсталляция (**[**Installation**](https://kivy.org/doc/stable/gettingstarted/installation.html)**)**

Для начала необходимо скачать последнюю версию Kivy для нужной среды разработки:

<http://kivy.org/#download>

После загрузки следовать инструкции по установке самого фреймворка и необходимых ему зависимостей для конкретной платформы:

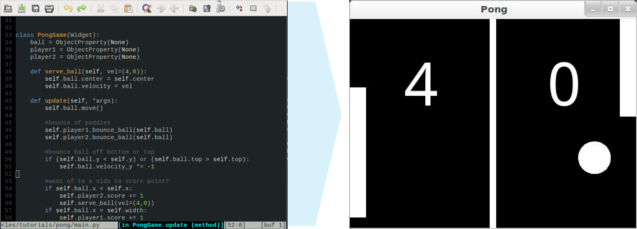
[Installation for Developement](https://kivy.org/doc/stable/installation/installation-devel.html#installation-devel)

Исходный код с последними дополнениями к фреймворку можно получить на github:

<https://github.com/kivy/kivy>

**3. Первое приложение**

[Руководство по созданию «Игры в пинг-понг»](https://kivy.org/doc/stable/tutorials/pong.html) знакомит с основными шаблонами проектирования и процессом разработки приложений.

[](https://kivy.org/doc/stable/_images/gs-tutorial.png)

Следуя инструкциям, можно создать простое приложение и запустить его в среде разработки.

[Руководство по созданию «Игры в пинг-понг»](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/tutorials/pong.html) (перевод в Документации) - самая важная статья. Она закладывает основу для концепций, о которых будет сказано позже.

**4. Свойства классов в Kivy**

Kivy вводит новый способ объявления свойств в классе. Если в классическом варианте пишется:

class MyClass(object):

def \_\_init\_\_(self):

super(MyClass, self).\_\_init\_\_()

self.numeric\_var = 1

То, при использовании свойств Kivy, можно написать:

class MyClass(EventDispatcher):

numeric\_var = NumericProperty(1)

Эти свойства реализуют [шаблон проектирования Observer](https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D0%B1%D0%BB%D1%8E%D0%B4%D0%B0%D1%82%D0%B5%D0%BB%D1%8C_(%D1%88%D0%B0%D0%B1%D0%BB%D0%BE%D0%BD_%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)) . Они помогают:

* Легко манипулировать виджетами, определенными в [языке](https://kivy.org/doc/stable/guide/lang.html) [Kv](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/lang.html);
* Автоматически наблюдать любые изменения в функциях/коде;
* Проверять и подтверждать значения;
* Оптимизировать управление памятью.

Чтобы использовать эти свойства, **нужно объявить их на уровне класса** . То есть непосредственно в классе, а не в каком-либо методе класса. Свойство - это атрибут класса, который автоматически создает атрибуты экземпляра. Каждое свойство по умолчанию предоставляет событие on\_ < propertyname >, которое вызывается при изменении состояния/значения свойства.

Kivy предоставляет следующие свойства:

[NumericProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.NumericProperty),

[StringProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.StringProperty),

[ListProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.ListProperty),

[ObjectProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.ObjectProperty),

[BooleanProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.BooleanProperty),

[BoundedNumericProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.BoundedNumericProperty),

[OptionProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.OptionProperty),

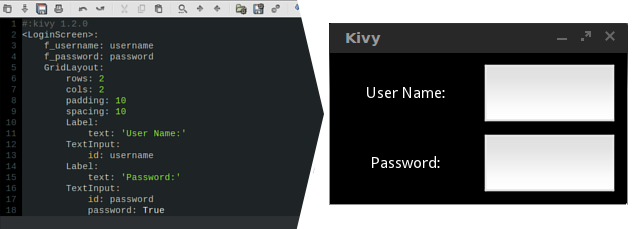
[ReferenceListProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.ReferenceListProperty),

[AliasProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.AliasProperty),

[DictProperty](https://kivy.org/doc/stable/api-kivy.properties.html#kivy.properties.DictProperty),

**5. Язык графического дизайна Kv Design Language**

В фрэймворк Kivy встроен язык проектированияKv Design Language, специально предназначенный для простого и масштабируемого графического дизайна. Это отделяет дизайн интерфейса от логики приложения, придерживаясь [принципа разделения интересов](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=http://en.wikipedia.org/wiki/Separation_of_concerns).

[](https://kivy.org/doc/stable/_images/gs-lang.png)

В приведенном выше коде:

< LoginScreen >: # каждый класс в приложении может быть представлен как правило в файле kv

GridLayout : # так вы добавляете свой виджет/макет к родителю

rows : 2 # так устанавливается каждое свойство виджета/макета

Вот так просто создать графический интерфейс на языке Kv. Документация - [Kv language](https://kivy.org/doc/stable/guide/lang.html).

**6. События**

Kivy в основном [событийно-ориентирован](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D0%B1%D1%8B%D1%82%D0%B8%D0%B9%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), то есть поток программы определяется событиями.

**События по времени**

../_images/gs-events-clock.png

Объект [Clock](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.clock.html) позволяет запланировать вызов функции в будущем как одноразовое событие с schedule\_once ( ) или как повторяющееся событие schedule\_interval ( ).

Можно создавать инициируемые события с помощью create\_trigger ( ). Триггеры имеют то преимущество, что вызываются только один раз, даже если запланированы несколько триггеров для одного и того же обратного вызова (callback).

**События ввода**

../_images/gs-events-input.png

Все события, связанные с [щелчком](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.input.motionevent.html%23kivy.input.motionevent.MotionEvent#kivy.input.motionevent.MotionEvent) мыши, касанием и колесом прокрутки, являются частью [MotionEvent](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.input.motionevent.html%23kivy.input.motionevent.MotionEvent#kivy.input.motionevent.MotionEvent) , расширенной [постобработкой ввода](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.input.postproc.html) и отправленной через событие [on\_motion](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.core.window.html%23kivy.core.window.WindowBase.on_motion#kivy.core.window.WindowBase.on_motion) в классе [Window](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.core.window.html%23kivy.core.window.WindowBase#kivy.core.window.WindowBase) . Затем это событие генерирует [on\_touch\_down ()](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget.on_touch_down#kivy.uix.widget.Widget.on_touch_down) , [on\_touch\_move ()](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget.on_touch_move#kivy.uix.widget.Widget.on_touch_move) и [on\_touch\_up ()](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget.on_touch_up#kivy.uix.widget.Widget.on_touch_up) события в [виджете](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget#kivy.uix.widget.Widget) .

Для более подробного объяснения - [Управление вводом](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.input.html).

**События класса**

../_images/gs-events-class.png

Базовый класс [EventDispatcher](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.event.html%23kivy.event.EventDispatcher#kivy.event.EventDispatcher) в [Widget](https://kivy.org/doc/stable/api-kivy.uix.widget.html#kivy.uix.widget.Widget), использует возможности [Свойств](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.properties.html) (Properties) для отправки изменений. Это означает, что когда виджет меняет свою позицию или размер, соответствующее событие запускается автоматически.

Кроме того, есть возможность создавать свои собственные события с помощью [register\_event\_type ( )](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.event.html%23kivy.event.EventDispatcher.register_event_type#kivy.event.EventDispatcher.register_event_type), о *чем свидетельствуют* события *on\_press* и *on\_release* в виджете [Button](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.button.html%23kivy.uix.button.Button#kivy.uix.button.Button).

Следует также отметить, что если переопределить событие, вы становитесь ответственным за реализацию всего его поведения, ранее обработанного базовым классом. Самый простой способ сделать это - вызвать *super ( )* :

def on\_touch\_down(self, touch):

if super(OurClassName, self).on\_touch\_down(touch):

return True

if not self.collide\_point(touch.x, touch.y):

return False

print('you touched me!')

return True

Больше о Событиях - в документации по [Events and Properties](https://kivy.org/doc/stable/guide/events.html).

**7. Дополнительные службы (NonWidget stuff)**

|  |  |
| --- | --- |
| [Анимация](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.animation.html%23kivy.animation.Animation#kivy.animation.Animation) используется для изменения свойств виджета (размера/позы/центра и т.д.) к целевому значению в пределах заданного интервала времени. Различные функции [перехода](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.animation.html%23kivy.animation.AnimationTransition#kivy.animation.AnimationTransition) можно использовать для анимации виджетов и создания плавного поведения пользовательского интерфейса. | animation_img |
| [Atlas](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.atlas.html%23kivy.atlas.Atlas#kivy.atlas.Atlas) - класс для управления картами текстур, т.е. для упаковки нескольких текстур в одно изображение. Это позволяет уменьшить количество загружаемых изображений и тем самым ускорить запуск приложения. | atlas_img |
| [Clock](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.clock.html%23kivy.clock.Clock#kivy.clock.Clock) предоставляет удобный способ планирования заданий на определённые интервалы по времени. Он предпочтительнее, чем *sleep( )*, который блокирует цикл событий kivy. Эти интервалы могут быть установлены [до](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.clock.html%23schedule-before-frame#schedule-before-frame) или [после](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.clock.html%23schedule-before-frame#schedule-before-frame) инструкций рисования OpenGL. Часы также предоставляют возможность создавать [инициируемые события](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.clock.html%23triggered-events#triggered-events) (triggered events), которые группируются и вызываются только один раз перед следующим фрэймом. | * schedule\_once () * schedule\_interval () * unschedule () * create\_trigger () |
| [UrlRequest](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.network.urlrequest.html%23kivy.network.urlrequest.UrlRequest#kivy.network.urlrequest.UrlRequest) полезен для асинхронных запросов, которые не блокируют цикл обработки событий. Можно использовать его для управления ходом запросов URL через обратные вызовы. |  |

**8. Макеты (Layouts)**

Макеты - это контейнеры, используемые для организации виджетов определенным образом.

[AnchorLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.anchorlayout.html%23module-kivy.uix.anchorlayout#module-kivy.uix.anchorlayout) :

Виджеты могут быть привязаны к верху, низу, левому краю, правому краю или центру.

[BoxLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.boxlayout.html%23module-kivy.uix.boxlayout#module-kivy.uix.boxlayout) :

Виджеты располагаются последовательно, в «вертикальной» или «горизонтальной» ориентации.

[FloatLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.floatlayout.html%23module-kivy.uix.floatlayout#module-kivy.uix.floatlayout) :

Виджеты по существу не ограничены.

[RelativeLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.relativelayout.html%23module-kivy.uix.relativelayout#module-kivy.uix.relativelayout) :

Дочерние виджеты расположены относительно макета.

[GridLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.gridlayout.html%23module-kivy.uix.gridlayout#module-kivy.uix.gridlayout) :

Виджеты расположены в сетке, определенной *строками* (*rows)* и *рядами (сols)*.

[PageLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.pagelayout.html%23module-kivy.uix.pagelayout#module-kivy.uix.pagelayout) :

Используется для создания простых многостраничных макетов, позволяющих переключаться с одной страницы на другую, используя *границы (borders)*.

[ScatterLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.scatterlayout.html%23module-kivy.uix.scatterlayout#module-kivy.uix.scatterlayout) :

Виджеты расположены аналогично RelativeLayout , но их можно вращать и масштабировать.

[StackLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.stacklayout.html%23module-kivy.uix.stacklayout#module-kivy.uix.stacklayout) :

Виджеты укладываются в виде *lr-tb* (left to right then top to bottom) - *слева направо затем сверху вниз*, или, ноборот, *lr-tb* порядке.

Когда виджет добавляется в макет, для определения его размера и положения в зависимости от типа макета используются следующие свойства:

**size\_hint** : определяет размер виджета как часть размера родителей. Значения ограничены диапазоном 0.0 – 1.0, т. е. 0.01 = 1/100 родительского размера (1%) и 1.0 = 100%.

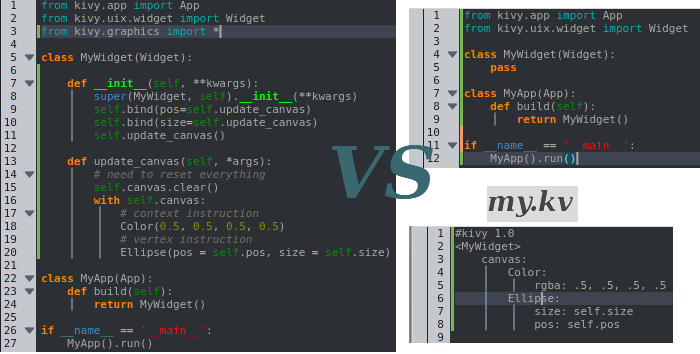
**pos\_hint** : используется для размещения виджета относительно родителя.

**size\_hint** и **pos\_hint** используются для расчета размера и положения виджета, только если значение не установлено в None. В противном случае, макет не будет размещать/изменять размер виджета и можно указать значения (x, y, ширина, высота) непосредственно в экранных координатах.

**9. Рисование (Drawing)**

У каждого виджета есть холст, то есть место для рисования. Холст представляет собой группу инструкций рисования, которые выполняются всякий раз, когда происходит изменение в графическом представлении виджета.

Вы можете добавить два типа инструкций на холст: *контекстные* инструкции и *вершинные* инструкции. Вы можете добавить инструкции либо из кода Python, либо из файла kv (предпочтительный способ). Если вы добавляете их через файл kv , преимущество заключается в том, что они автоматически обновляются, когда любое свойство зависит от изменений. В Python это нужно делать самостоятельно.



В обоих случаях холст *MyWidget* перерисовывается всякий раз, когда изменяется позиция или размер виджета.

Можно использовать группы [canvas.before](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23kivy.graphics.Canvas.before#kivy.graphics.Canvas.before) или [canvas.after](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23kivy.graphics.Canvas.after#kivy.graphics.Canvas.after) для разделения инструкций в зависимости от того, когда вы хотите, чтобы они выполнялись.

Как обрабатывается графика Kivy - [здесь](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23module-kivy.graphics#module-kivy.graphics).

**10. Компиляция Упаковка Сборка** [**¶**](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/gettingstarted/packaging.html%23packaging#packaging)

* [Создание в пакет для Windows ,](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-windows.html) 
  + [Требования](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-windows.html%23packaging-windows-requirements#packaging-windows-requirements)
  + [PyInstaller по умолчанию крюк](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-windows.html%23create-the-spec-file#create-the-spec-file)
* [Создание пакетов для OS X](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-osx.html) 
  + [Использование Buildozer](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-osx.html%23osx-kivy-sdk-packager#osx-kivy-sdk-packager)
  + [Использование PyInstaller и Homebrew](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-osx.html%23osx-pyinstaller#osx-pyinstaller)
* [Создать в пакет для Android](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-android.html) 
  + [Упаковка с Python для Android](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-android.html%23packaging-your-application-into-apk#packaging-your-application-into-apk)
  + [Упаковка вашего приложения для Kivy Launcher](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-android.html%23packaging-your-application-for-kivy-launcher#packaging-your-application-for-kivy-launcher)
* [Создание в пакет для IOS](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-ios.html) 
  + [Обобщение на распределение](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-ios.html%23compile-the-distribution#compile-the-distribution)
  + [Создание в Xcode проект](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-ios.html%23create-an-xcode-project#create-an-xcode-project)
  + [Настройка в Xcode проект](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-ios.html%23customize#customize)
  + [Известные проблемы](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-ios.html%23known-issues#known-issues)
  + [Вопросы-Ответы](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/packaging-ios.html%23ios-packaging-faq#ios-packaging-faq)

**Packaging[¶](https://kivy.org/doc/stable/gettingstarted/packaging.html" \l "packaging" \o "Permalink to this headline)**

* [Create a package for Windows](https://kivy.org/doc/stable/guide/packaging-windows.html)
  + [Requirements](https://kivy.org/doc/stable/guide/packaging-windows.html#packaging-windows-requirements)
  + [PyInstaller default hook](https://kivy.org/doc/stable/guide/packaging-windows.html#create-the-spec-file)
* [Creating packages for OS X](https://kivy.org/doc/stable/guide/packaging-osx.html)
  + [Using Buildozer](https://kivy.org/doc/stable/guide/packaging-osx.html#osx-kivy-sdk-packager)
  + [Using PyInstaller and Homebrew](https://kivy.org/doc/stable/guide/packaging-osx.html#osx-pyinstaller)
* [Create a package for Android](https://kivy.org/doc/stable/guide/packaging-android.html)
  + [Packaging with python-for-android](https://kivy.org/doc/stable/guide/packaging-android.html#packaging-your-application-into-apk)
  + [Packaging your application for the Kivy Launcher](https://kivy.org/doc/stable/guide/packaging-android.html#packaging-your-application-for-kivy-launcher)
* [Create a package for IOS](https://kivy.org/doc/stable/guide/packaging-ios.html)
  + [Compile the distribution](https://kivy.org/doc/stable/guide/packaging-ios.html#compile-the-distribution)
  + [Create an Xcode project](https://kivy.org/doc/stable/guide/packaging-ios.html#create-an-xcode-project)
  + [Customize the Xcode project](https://kivy.org/doc/stable/guide/packaging-ios.html#customize)
  + [Known issues](https://kivy.org/doc/stable/guide/packaging-ios.html#known-issues)
  + [FAQ](https://kivy.org/doc/stable/guide/packaging-ios.html#ios-packaging-faq)

**11. Примеры** [**¶**](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/gettingstarted/examples.html%23examples#examples)

Следуйте за примерами.

| **каталог** | **Имя файла / с** | **Пример Описание** |
| --- | --- | --- |
| * ./ примеры / анимация | * animate.py | * Анимация виджетов с использованием [Animation](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.animation.html%23kivy.animation.Animation#kivy.animation.Animation) |
| * ./ примеры / приложение | * app\_with\_build.py * app\_with\_kv.py * app\_with\_kv\_in\_template1.py | * Пример приложения с использованием [build ()](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.app.html%23kivy.app.App.build#kivy.app.App.build) * Применение от [.](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.lang.html%23module-kivy.lang#module-kivy.lang) файл [kv](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.lang.html%23module-kivy.lang#module-kivy.lang) * Приложение из [каталога kv\_directory](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.lang.html%23module-kivy.lang#module-kivy.lang) |
| * ./ examples / canvas | * bezier.py * canvas\_stress.py * mesh.py * multitexture.py * stencil\_canvas.py | * Как для нарисовать [Безье](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23kivy.graphics.Bezier#kivy.graphics.Bezier) линий * Стресс- тест [Холст](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23kivy.graphics.Canvas#kivy.graphics.Canvas) * Как использовать [Mesh](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23kivy.graphics.Mesh#kivy.graphics.Mesh) в киве * Как работать с несколькими текстурами с помощью [шейдера](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.shader.html%23kivy.graphics.shader.Shader#kivy.graphics.shader.Shader) * Как использовать [трафарет](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.stencil_instructions.html%23module-kivy.graphics.stencil_instructions#module-kivy.graphics.stencil_instructions) на холсте виджета |
| * ./ examples / demo : | * camera\_puzzle.py | * Головоломка с помощью [камеры](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.camera.html%23kivy.uix.camera.Camera#kivy.uix.camera.Camera) вывода |
| * ./ examples / demo / pictures | * main.py | * Основные моменты использования в [изображения](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.image.html%23module-kivy.uix.image#module-kivy.uix.image) и [Scatter](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.scatter.html%23module-kivy.uix.scatter#module-kivy.uix.scatter) Widgets |
| * ./ examples / demo / shadeditor | * main.py | * Как использовать [фрагментные и вершинные шейдеры](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.shader.html%23kivy.graphics.shader.Shader#kivy.graphics.shader.Shader) . |
| * ../ examples / demo / showcase | * main.py | * Витрина с [виджетами](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget#kivy.uix.widget.Widget) и [макетами,](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.layout.html%23module-kivy.uix.layout#module-kivy.uix.layout) доступными в кивах |
| * ./ examples / demo / touchtracer | * main.py | * Нарисуйте линии под каждым обнаруженным касанием * Хорошее место, чтобы понять, как [сенсорные события](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.input.motionevent.html%23module-kivy.input.motionevent#module-kivy.input.motionevent) работают в kivy |
| * ./ примеры / рамки / витая | * echo\_client\_app.py * echo\_server\_app.py | * Приложение клиента и сервера, использующее [Twisted внутри Kivy](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/other-frameworks.html) |
| * ./ примеры / жесты | * gesture\_board.py | * Чистая доска, чтобы попробовать [жесты](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.gesture.html%23module-kivy.gesture#module-kivy.gesture) |
| * ./ examples / guide / designwithkv | * main.py | * Руководство по программированию примеры того, как [проектировать с помощью kv lang](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/lang.html) |
| * ./ examples / tutorials / firstwidget | * 1\_skeleton.py * 2\_print\_touch.py * 3\_draw\_ellipse.py * 4\_draw\_line.py * 5\_random\_colors.py * 6\_button.py | * Пример руководства по программированию: [Ваш первый виджет](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/tutorials/firstwidget.html) |
| * ./ examples / guide / quickstart | * main.py | * Пример руководства по программированию: [создание приложения](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/basic.html) |
| * ./ examples / kinect | * main.py | * Как использовать [Kinect](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=http://en.wikipedia.org/wiki/kinect) для ввода |
| * ./ примеры / кв | * kvrun.py | * Загружайте файлы kv , используйте [kv lang](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.lang.html%23module-kivy.lang#module-kivy.lang) для загрузки различных [виджетов](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget#kivy.uix.widget.Widget) . |
| * ./ examples / RST\_Editor | * main.py | * Редактор RST для виджета [RstDocument](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.rst.html%23kivy.uix.rst.RstDocument#kivy.uix.rst.RstDocument) |
| * ./ примеры / настройки | * main.py | * Как пользоваться [настройками](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.settings.html%23kivy.uix.settings.Settings#kivy.uix.settings.Settings) классов. |
| * ./ примеры / шейдер | * plasma.py * shadertree.py | * Как использовать разные [шейдеры](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.shader.html%23kivy.graphics.shader.Shader#kivy.graphics.shader.Shader) . |
| * ./ examples / widgets | * accordion\_1.py * asyncimage.py * boxlayout\_pos\_hint.py * bubble\_test.py * customcollide.py * fbowidget.py * image\_mipmap.py * keyboardlistener.py * label\_mipmap.py * label\_with\_markup.py * popup\_with\_kv.py * rstexample.py * scatter.py * screenmanager.py * scrollview.py * spinner.py * tabbedpanel.py * tabbed\_panel\_showcase.py * textalign.py * textinput.py * unicode\_textinput.py * videoplayer.py | * Использование и демонстрация [Аккордеонного](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.accordion.html%23module-kivy.uix.accordion#module-kivy.uix.accordion) Виджета * Использование и демонстрация виджета [AsyncImage](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.image.html%23kivy.uix.image.AsyncImage#kivy.uix.image.AsyncImage) * Витрина pos\_hint под [BoxLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.boxlayout.html%23module-kivy.uix.boxlayout#module-kivy.uix.boxlayout) * Использование и демонстрация [Bubble](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.bubble.html%23module-kivy.uix.bubble#module-kivy.uix.bubble) Widget * Тест на [столкновение](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget.collide_point#kivy.uix.widget.Widget.collide_point) с виджетом нестандартной формы * Использование [FBO](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.fbo.html%23module-kivy.graphics.fbo#module-kivy.graphics.fbo) для ускорения графики * Как использовать виджет [изображения](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.image.html%23module-kivy.uix.image#module-kivy.uix.image) с mipmap * Прослушайте ввод с клавиатуры и выведите результат на консоль * Как использовать виджет [Label](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.label.html%23module-kivy.uix.label#module-kivy.uix.label) с * Использование виджета [Label](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.label.html%23module-kivy.uix.label#module-kivy.uix.label) с разметкой * Использование виджета [Popup](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.popup.html%23module-kivy.uix.popup#module-kivy.uix.popup) с языком kv * Использование и демонстрация виджета [RstDocument](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.rst.html%23kivy.uix.rst.RstDocument#kivy.uix.rst.RstDocument) * Использование и демонстрация [Scatter](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.scatter.html%23module-kivy.uix.scatter#module-kivy.uix.scatter) Widget * Использование и демонстрация модуля [ScreenManager](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.screenmanager.html%23module-kivy.uix.screenmanager#module-kivy.uix.screenmanager) * Использование и демонстрация [ScrollView](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.scrollview.html%23module-kivy.uix.scrollview#module-kivy.uix.scrollview) Widget * Использование и демонстрация [Spinner](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.spinner.html%23module-kivy.uix.spinner#module-kivy.uix.spinner) Widget * Использование в виде простого [TabbedPanel](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.tabbedpanel.html%23kivy.uix.tabbedpanel.TabbedPanel#kivy.uix.tabbedpanel.TabbedPanel) * Расширенная витрина из [TabbedPanel](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.tabbedpanel.html%23kivy.uix.tabbedpanel.TabbedPanel#kivy.uix.tabbedpanel.TabbedPanel) * Использование выравнивания текста в виджете [Label](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.label.html%23module-kivy.uix.label#module-kivy.uix.label) * Использование и [демонстрация](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.textinput.html%23module-kivy.uix.textinput#module-kivy.uix.textinput) виджета [TextInput](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.textinput.html%23module-kivy.uix.textinput#module-kivy.uix.textinput) * Витрина Юникод текста в [TextInput](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.textinput.html%23module-kivy.uix.textinput#module-kivy.uix.textinput) Widget * Использование и опции [VideoPlayer](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.videoplayer.html%23module-kivy.uix.videoplayer#module-kivy.uix.videoplayer) Widget |
| * ./ examples / widgets / sequenced\_images : | * main.py | * Продемонстрировать использование секвенированных изображений с использованием [GIF-](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=http://en.wikipedia.org/wiki/Graphics_Interchange_Format) файлов и изображений в ZIP-файле. |

**11. Примеры** [**¶**](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/gettingstarted/examples.html%23examples#examples)

Следуйте за примерами.

| **каталог** | **Имя файла / с** | **Пример Описание** |
| --- | --- | --- |
| * ./ примеры / анимация | * animate.py | * Анимация виджетов с использованием [Animation](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.animation.html%23kivy.animation.Animation#kivy.animation.Animation) |
| * ./ примеры / приложение | * app\_with\_build.py * app\_with\_kv.py * app\_with\_kv\_in\_template1.py | * Пример приложения с использованием [build ()](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.app.html%23kivy.app.App.build#kivy.app.App.build) * Применение от [.](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.lang.html%23module-kivy.lang#module-kivy.lang) файл [kv](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.lang.html%23module-kivy.lang#module-kivy.lang) * Приложение из [каталога kv\_directory](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.lang.html%23module-kivy.lang#module-kivy.lang) |
| * ./ examples / canvas | * bezier.py * canvas\_stress.py * mesh.py * multitexture.py * stencil\_canvas.py | * Как для нарисовать [Безье](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23kivy.graphics.Bezier#kivy.graphics.Bezier) линий * Стресс- тест [Холст](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23kivy.graphics.Canvas#kivy.graphics.Canvas) * Как использовать [Mesh](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html%23kivy.graphics.Mesh#kivy.graphics.Mesh) в киве * Как работать с несколькими текстурами с помощью [шейдера](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.shader.html%23kivy.graphics.shader.Shader#kivy.graphics.shader.Shader) * Как использовать [трафарет](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.stencil_instructions.html%23module-kivy.graphics.stencil_instructions#module-kivy.graphics.stencil_instructions) на холсте виджета |
| * ./ examples / demo : | * camera\_puzzle.py | * Головоломка с помощью [камеры](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.camera.html%23kivy.uix.camera.Camera#kivy.uix.camera.Camera) вывода |
| * ./ examples / demo / pictures | * main.py | * Основные моменты использования в [изображения](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.image.html%23module-kivy.uix.image#module-kivy.uix.image) и [Scatter](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.scatter.html%23module-kivy.uix.scatter#module-kivy.uix.scatter) Widgets |
| * ./ examples / demo / shadeditor | * main.py | * Как использовать [фрагментные и вершинные шейдеры](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.shader.html%23kivy.graphics.shader.Shader#kivy.graphics.shader.Shader) . |
| * ../ examples / demo / showcase | * main.py | * Витрина с [виджетами](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget#kivy.uix.widget.Widget) и [макетами,](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.layout.html%23module-kivy.uix.layout#module-kivy.uix.layout) доступными в кивах |
| * ./ examples / demo / touchtracer | * main.py | * Нарисуйте линии под каждым обнаруженным касанием * Хорошее место, чтобы понять, как [сенсорные события](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.input.motionevent.html%23module-kivy.input.motionevent#module-kivy.input.motionevent) работают в kivy |
| * ./ примеры / рамки / витая | * echo\_client\_app.py * echo\_server\_app.py | * Приложение клиента и сервера, использующее [Twisted внутри Kivy](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/other-frameworks.html) |
| * ./ примеры / жесты | * gesture\_board.py | * Чистая доска, чтобы попробовать [жесты](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.gesture.html%23module-kivy.gesture#module-kivy.gesture) |
| * ./ examples / guide / designwithkv | * main.py | * Руководство по программированию примеры того, как [проектировать с помощью kv lang](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/lang.html) |
| * ./ examples / tutorials / firstwidget | * 1\_skeleton.py * 2\_print\_touch.py * 3\_draw\_ellipse.py * 4\_draw\_line.py * 5\_random\_colors.py * 6\_button.py | * Пример руководства по программированию: [Ваш первый виджет](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/tutorials/firstwidget.html) |
| * ./ examples / guide / quickstart | * main.py | * Пример руководства по программированию: [создание приложения](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide/basic.html) |
| * ./ examples / kinect | * main.py | * Как использовать [Kinect](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=http://en.wikipedia.org/wiki/kinect) для ввода |
| * ./ примеры / кв | * kvrun.py | * Загружайте файлы kv , используйте [kv lang](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.lang.html%23module-kivy.lang#module-kivy.lang) для загрузки различных [виджетов](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget#kivy.uix.widget.Widget) . |
| * ./ examples / RST\_Editor | * main.py | * Редактор RST для виджета [RstDocument](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.rst.html%23kivy.uix.rst.RstDocument#kivy.uix.rst.RstDocument) |
| * ./ примеры / настройки | * main.py | * Как пользоваться [настройками](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.settings.html%23kivy.uix.settings.Settings#kivy.uix.settings.Settings) классов. |
| * ./ примеры / шейдер | * plasma.py * shadertree.py | * Как использовать разные [шейдеры](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.shader.html%23kivy.graphics.shader.Shader#kivy.graphics.shader.Shader) . |
| * ./ examples / widgets | * accordion\_1.py * asyncimage.py * boxlayout\_pos\_hint.py * bubble\_test.py * customcollide.py * fbowidget.py * image\_mipmap.py * keyboardlistener.py * label\_mipmap.py * label\_with\_markup.py * popup\_with\_kv.py * rstexample.py * scatter.py * screenmanager.py * scrollview.py * spinner.py * tabbedpanel.py * tabbed\_panel\_showcase.py * textalign.py * textinput.py * unicode\_textinput.py * videoplayer.py | * Использование и демонстрация [Аккордеонного](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.accordion.html%23module-kivy.uix.accordion#module-kivy.uix.accordion) Виджета * Использование и демонстрация виджета [AsyncImage](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.image.html%23kivy.uix.image.AsyncImage#kivy.uix.image.AsyncImage) * Витрина pos\_hint под [BoxLayout](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.boxlayout.html%23module-kivy.uix.boxlayout#module-kivy.uix.boxlayout) * Использование и демонстрация [Bubble](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.bubble.html%23module-kivy.uix.bubble#module-kivy.uix.bubble) Widget * Тест на [столкновение](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.widget.html%23kivy.uix.widget.Widget.collide_point#kivy.uix.widget.Widget.collide_point) с виджетом нестандартной формы * Использование [FBO](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.fbo.html%23module-kivy.graphics.fbo#module-kivy.graphics.fbo) для ускорения графики * Как использовать виджет [изображения](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.image.html%23module-kivy.uix.image#module-kivy.uix.image) с mipmap * Прослушайте ввод с клавиатуры и выведите результат на консоль * Как использовать виджет [Label](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.label.html%23module-kivy.uix.label#module-kivy.uix.label) с * Использование виджета [Label](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.label.html%23module-kivy.uix.label#module-kivy.uix.label) с разметкой * Использование виджета [Popup](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.popup.html%23module-kivy.uix.popup#module-kivy.uix.popup) с языком kv * Использование и демонстрация виджета [RstDocument](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.rst.html%23kivy.uix.rst.RstDocument#kivy.uix.rst.RstDocument) * Использование и демонстрация [Scatter](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.scatter.html%23module-kivy.uix.scatter#module-kivy.uix.scatter) Widget * Использование и демонстрация модуля [ScreenManager](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.screenmanager.html%23module-kivy.uix.screenmanager#module-kivy.uix.screenmanager) * Использование и демонстрация [ScrollView](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.scrollview.html%23module-kivy.uix.scrollview#module-kivy.uix.scrollview) Widget * Использование и демонстрация [Spinner](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.spinner.html%23module-kivy.uix.spinner#module-kivy.uix.spinner) Widget * Использование в виде простого [TabbedPanel](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.tabbedpanel.html%23kivy.uix.tabbedpanel.TabbedPanel#kivy.uix.tabbedpanel.TabbedPanel) * Расширенная витрина из [TabbedPanel](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.tabbedpanel.html%23kivy.uix.tabbedpanel.TabbedPanel#kivy.uix.tabbedpanel.TabbedPanel) * Использование выравнивания текста в виджете [Label](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.label.html%23module-kivy.uix.label#module-kivy.uix.label) * Использование и [демонстрация](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.textinput.html%23module-kivy.uix.textinput#module-kivy.uix.textinput) виджета [TextInput](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.textinput.html%23module-kivy.uix.textinput#module-kivy.uix.textinput) * Витрина Юникод текста в [TextInput](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.textinput.html%23module-kivy.uix.textinput#module-kivy.uix.textinput) Widget * Использование и опции [VideoPlayer](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.videoplayer.html%23module-kivy.uix.videoplayer#module-kivy.uix.videoplayer) Widget |
| * ./ examples / widgets / sequenced\_images : | * main.py | * Продемонстрировать использование секвенированных изображений с использованием [GIF-](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=http://en.wikipedia.org/wiki/Graphics_Interchange_Format) файлов и изображений в ZIP-файле. |

**12. Погружение** в Kivy

Kivy поставляется с набором [примеров](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/gettingstarted/examples.html) в каталоге kivy\_installation / examples. Вы должны попытаться изменить / улучшить / адаптировать их к вашим потребностям.

Просмотрите нашу [вики](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://github.com/kivy/kivy/wiki) для получения информации о связанных проектах, учебные пособия и фрагменты.

Понять основы [графики](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.graphics.html) .

Посмотрите на встроенные [виджеты](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.uix.html) .

Следуйте [Руководству](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide-index.html) по [программированию,](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/guide-index.html) чтобы узнать больше о kivy .

Посмотрите, как использовать разные [Модули](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.modules.html) в разделе модулей, таких как [Инспектор](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.modules.inspector.html) для проверки в реальном времени.

Узнайте, как обрабатывать пользовательское [управление вводом](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.input.html) .

Ознакомьтесь с [Kivy Framework](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/api-kivy.html%23module-kivy#module-kivy) .

Kivy имеет открытый исходный код, поэтому вы можете **внести свой вклад** . Ознакомьтесь с разделом « [Содействие](https://translate.google.com/translate?hl=ru&prev=_t&sl=en&tl=ru&u=https://kivy.org/doc/stable/contribute.html) » для получения рекомендаций.