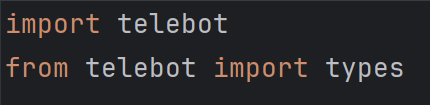
**Урок 8**

**Расположение кнопок**

На предыдущем уроке мы создали inline-кнопки, которые «встраиваются» в само сообщение. Создали мини-викторину с ботом. Познакомились с параметром **callback\_data**, классами **InlineKeyboardMarkup** и **InlineKeyboardButton**

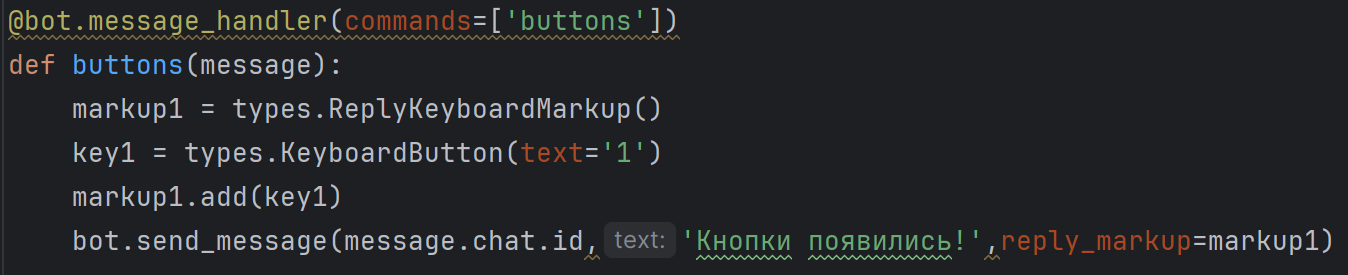
В прошлый раз мы уже пробовали менять расположение inline-кнопок, для этого мы указывали параметр **row\_width.** Он определяет количество кнопок в одном ряду. Сегодня мы разберёмся в этой теме подробнее и попробую менять разметку reply-кнопок и inline-кнопок.

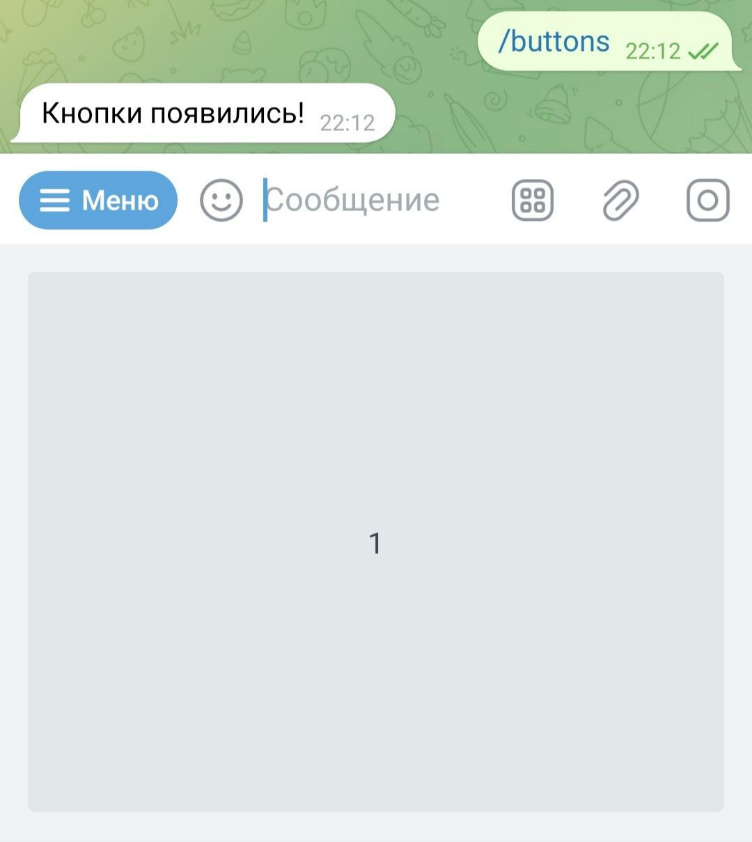
Для начала мы рассмотрим клавиатуру **Reply.** Для того, чтобы было удобнее использовать библиотеку types можно в начале кода сразу импортировать её



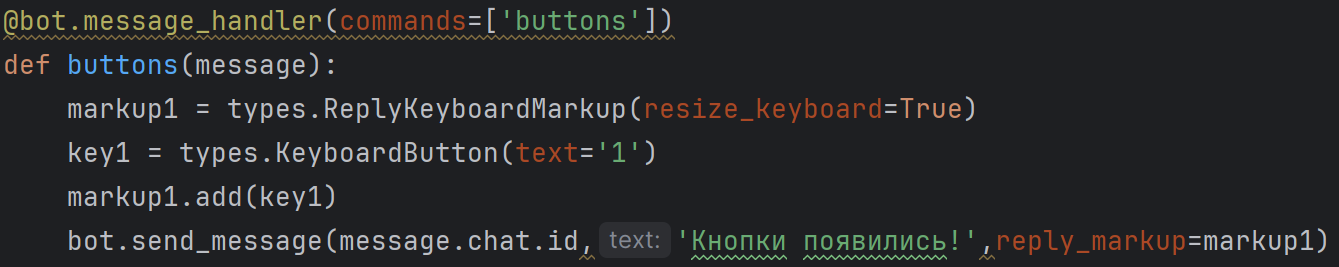
**Самостоятельная работа**

Добавьте боту новую команду **/** **buttons.** Запрограммируйте бота так, чтобы при вызове этой команды появлялась reply-кнопка.



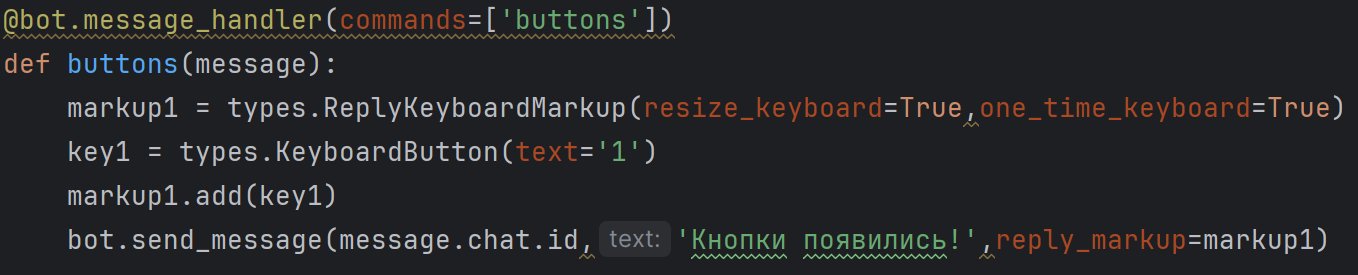


Отлично, клавиатура появилась! Но эта одна кнопка с маленьким текстом занимает очень много места. Telegram позволяет автоматически уменьшить размер кнопок, для этого необходимо передать в инициализатор класса **ReplyKeyboardMarkup** параметру **resize\_keyboard** значение **True**.





Так как у нас нет обработчика обычных сообщений, текст, отправляемый нажатием на эту кнопку, остаётся без ответа. А что ещё можно было заметить при использовании этой кнопки? Она остаётся всё там же, даже если мы после нажатия хотели бы вернуться к привычной клавиатуре. Для этого тоже есть решение — необходимо указать параметр **one\_time\_keyboard=True**



Теперь, когда мы один раз нажимаем на кнопку, она исчезает и на её месте появляется обычная клавиатура. Это особенно удобно, если вы тестируете своего бота через приложение на телефоне.

**Добавляем больше кнопок**

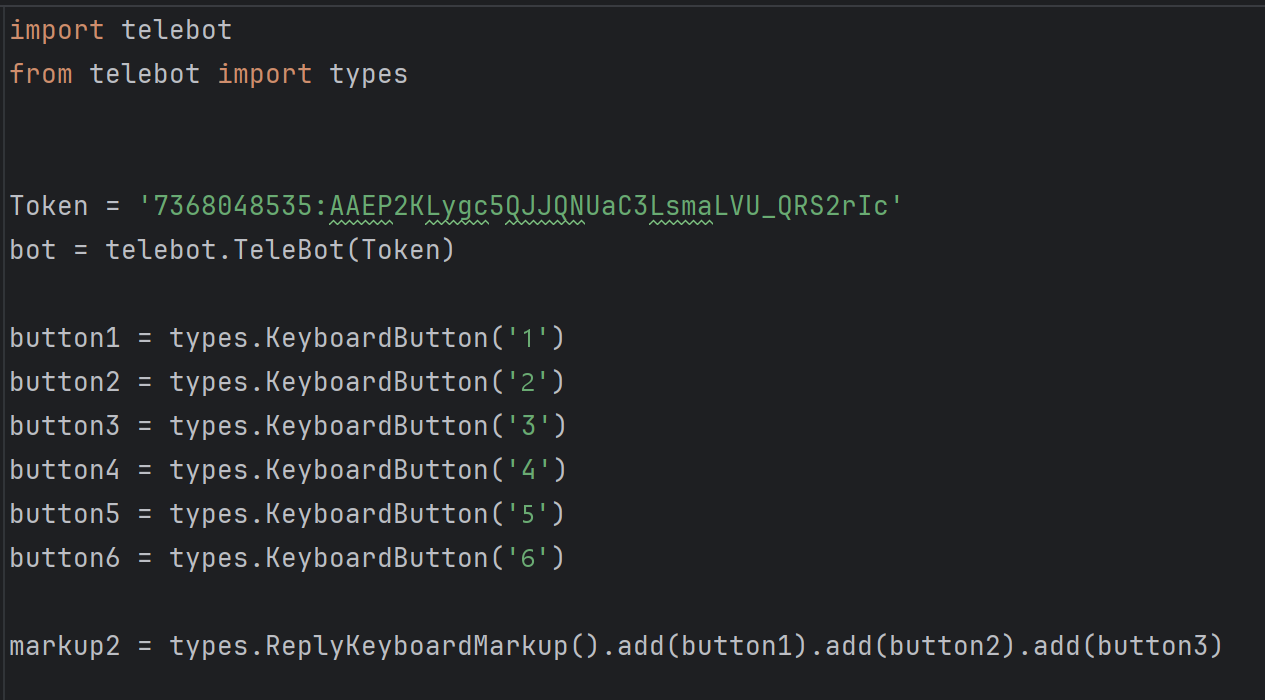
Рассмотрим подробно работу встроенных методов для создания более сложных клавиатур, а именно **row**, **insert** и **add**.

* Метод **add** принимает в себя любое количество кнопок, всегда начинает добавление с новой строки и переносит ряд при достижении значения установленной ширины.
* Метод **row** тоже принимает любое количество, но при этом не переносит кнопки на новый ряд, а добавляет всё полученное в одну строчку.
* Метод **insert** работает по схеме схожей с **add**, но только начинает добавлять к последнему ряду. И только если там уже достигнута максимальная ширина, начинает новую строку.

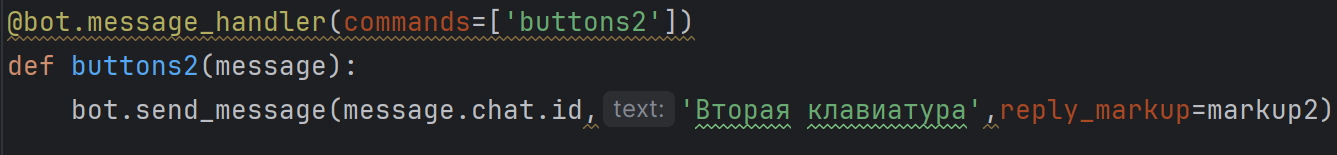
Для того, чтобы было удобнее тестировать разное расположение кнопок можно создать кнопки, которые мы сможем использовать повторно и сгенерировать несколько разных клавиатур.

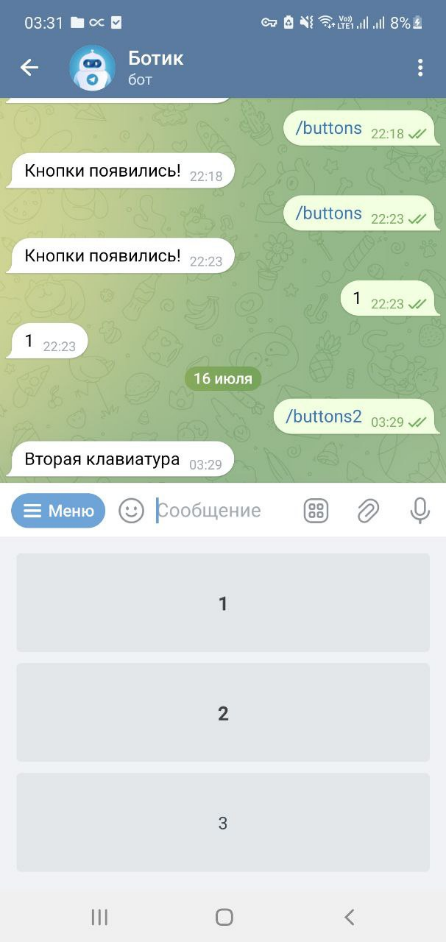
**Самостоятельная работа**

Добавьте несколько новых команд своему боту, например, /buttins2, /buttons3, buttons4. Создайте кнопки и клавиатуру Reply перед обработчиком сообщений.

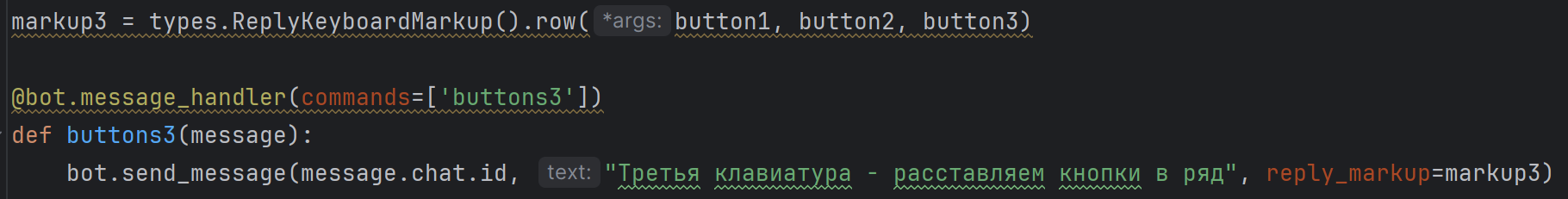


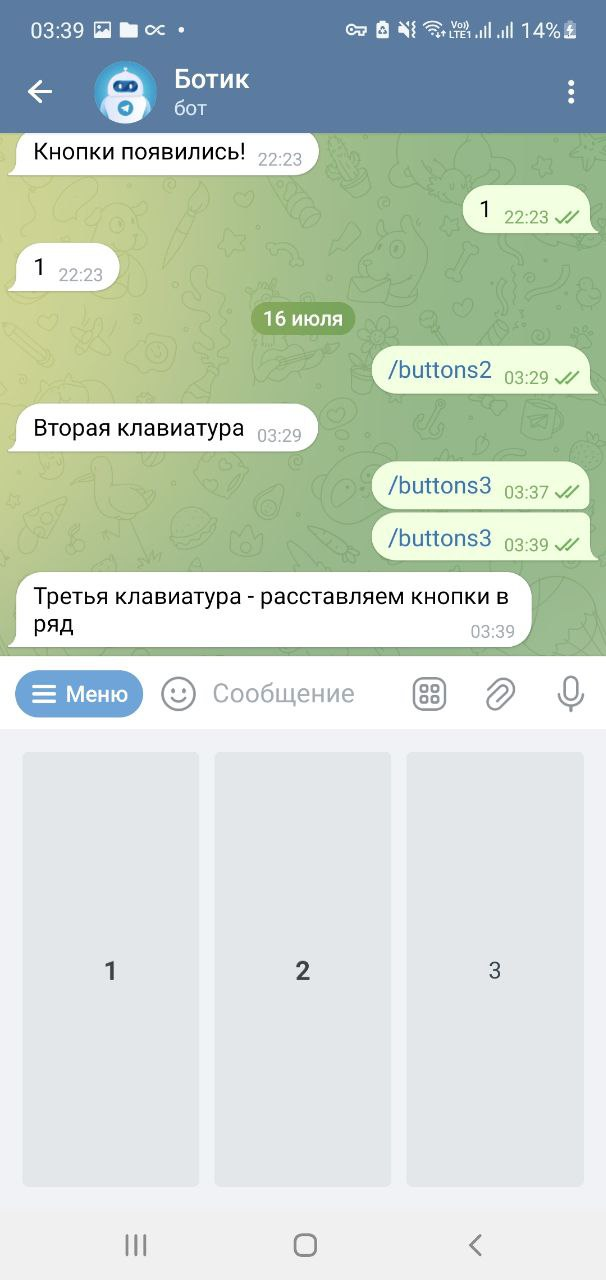
Отправляем все готовые кнопки и смотрим на результат:



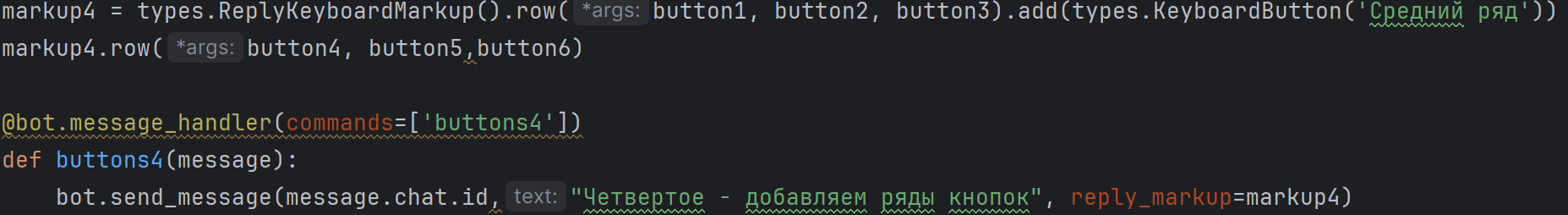


Далее попробуем расставить кнопки в один ряд:





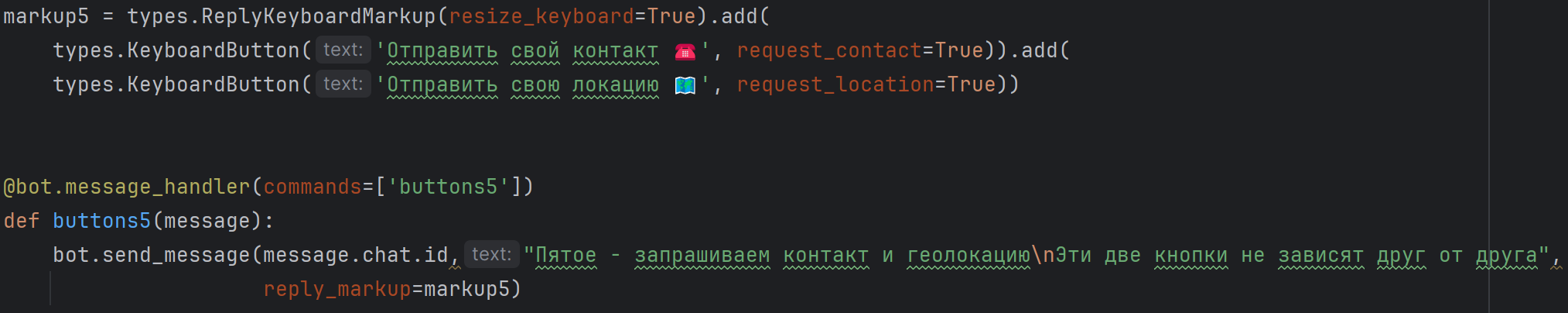
Следующий вариант:

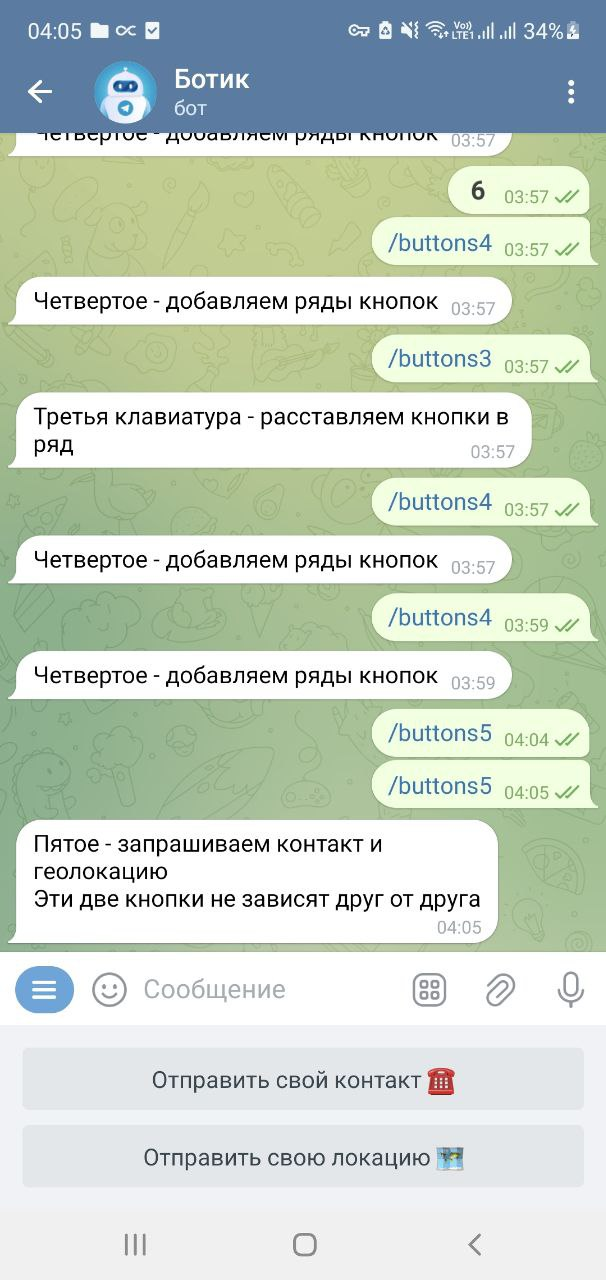




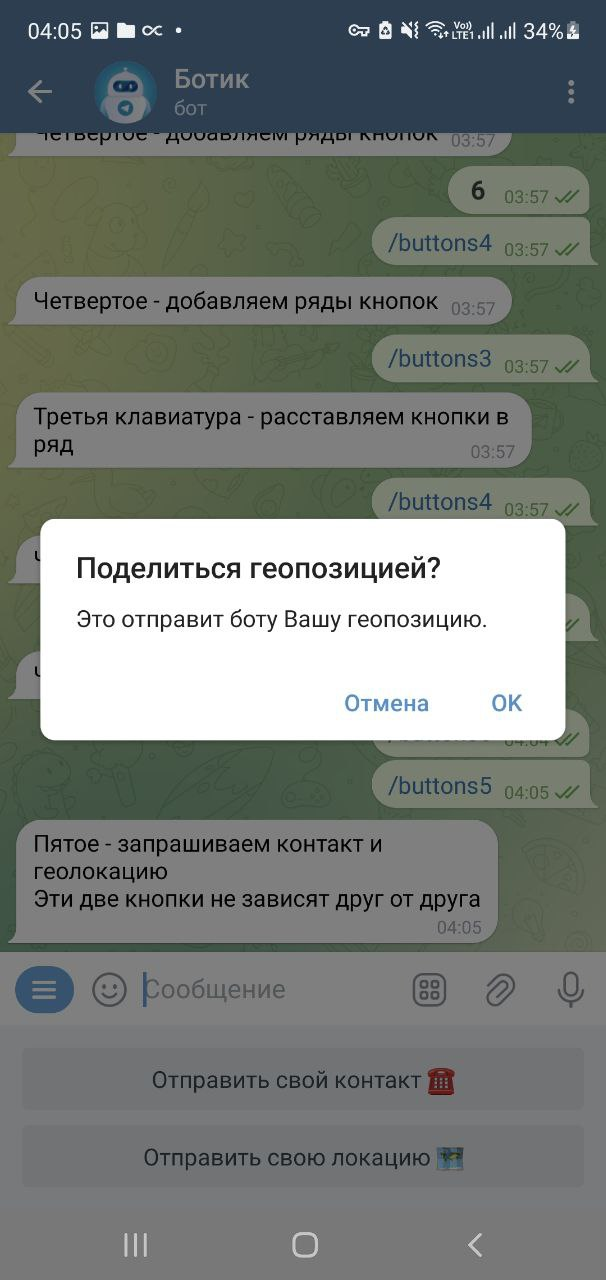
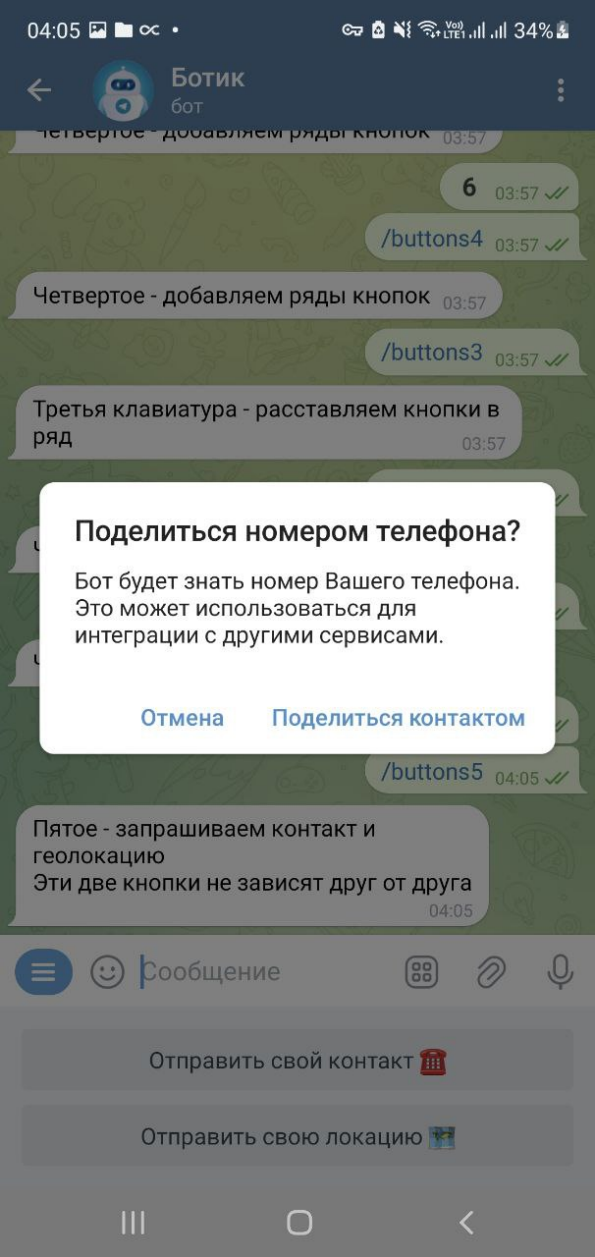
Здесь наглядно видно работу методов добавления кнопок в разметку.

**ReplyKeyboardMarkup** позволяет запросить у пользователя его контакт или локацию. Это те самые два исключения из правила, когда при нажатии кнопки будет отправлено не то, что написано на кнопке. Их можно отправлять как по одной, так и в составе более сложной клавиатуры. Для этого указываем параметры **request\_contact** и **request\_location**. Добавим обе кнопки и посмотрим, что будет:



****

Попробуем нажать на кнопки. При нажатии на каждую из этих кнопок клиент спросит, уверены ли вы, что хотите поделиться с ботом этими данными и при утвердительном ответе отправит их.

** **

**Самостоятельная работа**

Добавьте кнопки с цифрами и буквами с помощью разных методов.

**Inline клавиатуры**

Теперь перейдем к inline-клавиатурам. Они имеют больше параметров, поэтому позволяют нам делать больше разных вещей. Самое популярное использование — как кнопка, которая передаёт значение, то есть «если пользователь нажал кнопку X, сделать Y». На предыдущем занятии мы уже использовали это свойство, когда делали мини-викторину. Также мы устанавливали ширину клавиатуры. Как это можно сделать? С помощью параметра **row\_width.** (напомню, что это не распространяется на метод row)

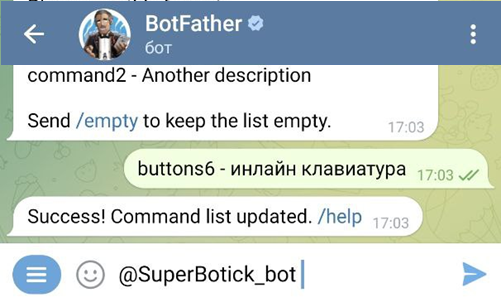
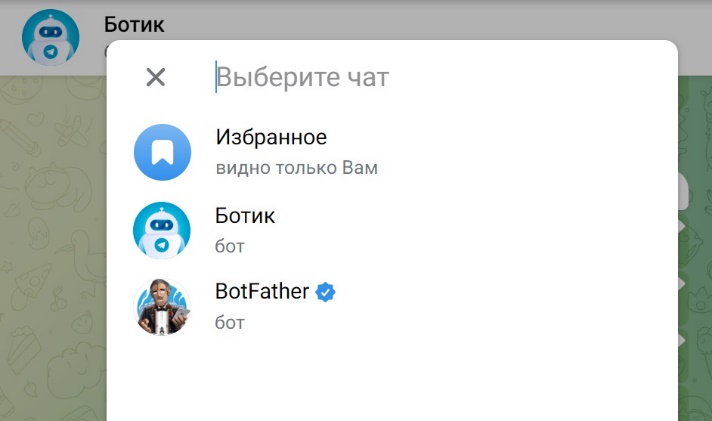
Разберёмся в том, что можно ещё сделать с помощью inline-кнопок. Ну и так как методы добавления новых элементов мы уже разобрали, добавим кнопки при этом не забудьте добавить новую команду своему боту.

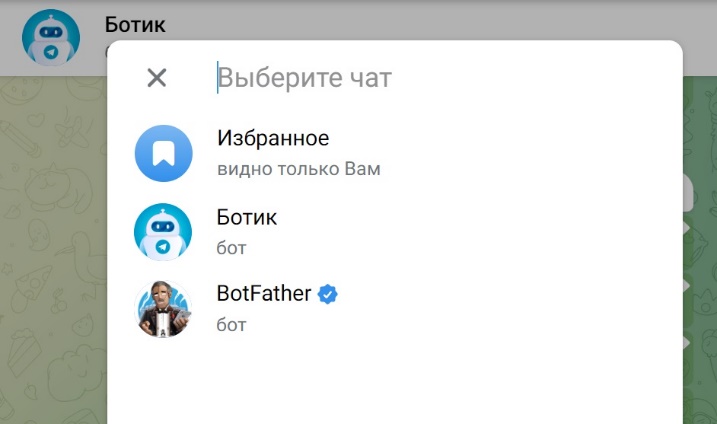
Проверим как работают кнопки и заодно разберём код.

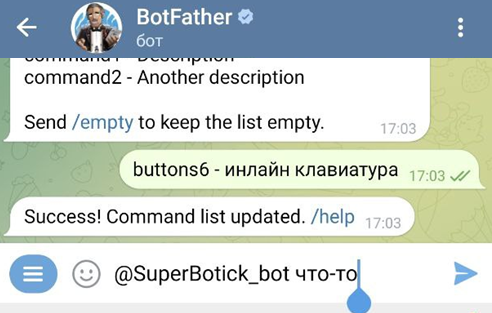


Сначала мы добавляем кнопки, у которых указываем уже не **callback\_data**, а другие параметры. То, что мы добавим в **switch\_inline\_query**, будет автоматически использовано при нажатии кнопки: пользователю предложат выбрать чат, а там вызовется inline режим этого бота (в поле ввода сообщения добавится юзернейм бота), следом через пробел будет прописано то, что мы указали. Параметр может принимать пустую строку, тогда inline режим запустится без какого-либо запроса, но если будет указан текст, то он и добавится.

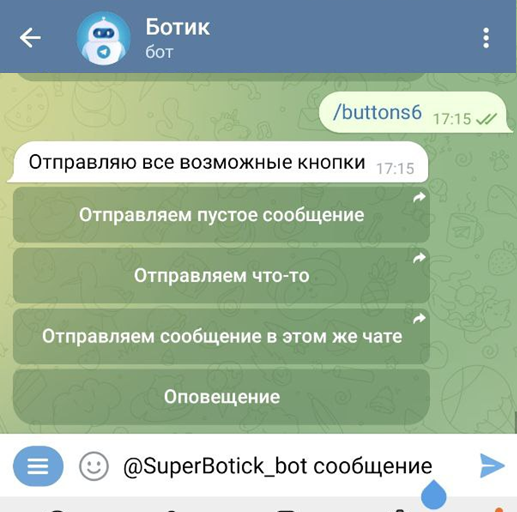
Первая кнопка отправляет пустой текст так как мы ничего не указали в кавычках:



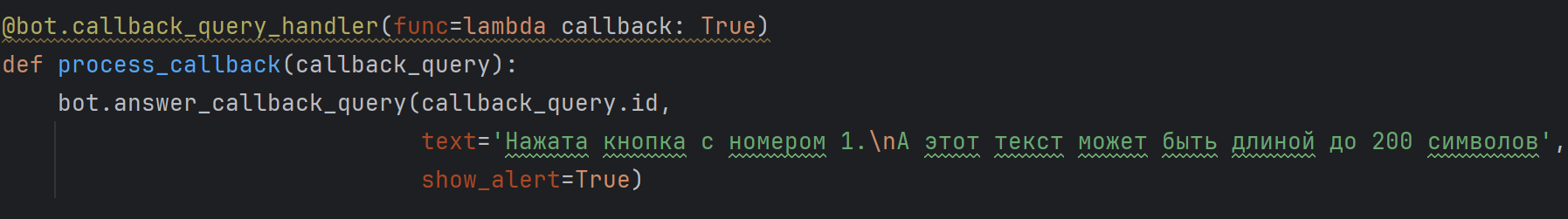
Вторая кнопка — отправляет текст «что-то»



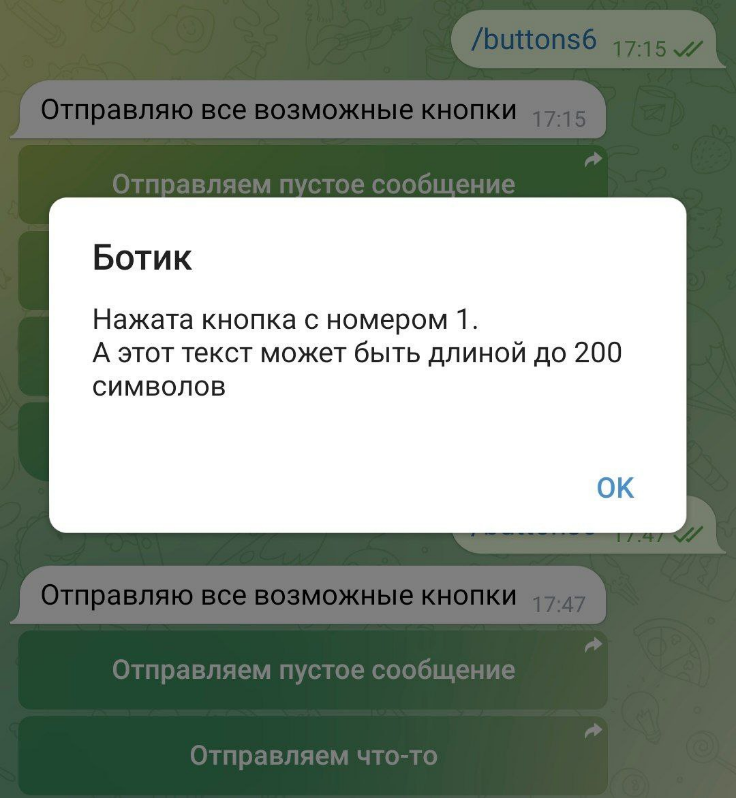
При использовании **switch\_inline\_query\_current\_chat** произойдёт ровно то же, что и в предыдущем пункте, но без выбора чата, а запустится в текущем. Нажмём на третью кнопку и проверим



Последняя кнопка пока не работает, так как мы отправили **callback\_data**, но не обработали его. Дополним программу:



Если пользователь нажмёт на кнопку, то появится окно с текстом, который мы указали. Для этого мы использовали команду **answer\_callback\_query**, в которой указали параметр show\_alert=True, таким образом мы сообщаем клиенту, что нужно показать окошко с текстом. Теперь протестируем четвёртую кнопку:



Всё получилось!

**Самостоятельная работа**

С помощью разных кнопок редактируйте программу с третьего урока «Диалог с ботом» — добавьте кнопки с фразами шаблонами, чтобы пользователь смог их использовать для общения с ботом.

**Рефлексия**

* Сегодня мы научились менять расположение inline-кнопок и reply-кнопок
* Познакомились с встроенными методами для создания более сложных клавиатур — row, insert и add
* Использовали новые функции кнопок, например, узнали, что кнопка может отправлять контакт или геолокацию

***Тайминг.***

|  |  |  |
| --- | --- | --- |
|  | Раздел урока | Время с начала урока, мин. |
|  | Повторение материала предыдущего урока. Постановка цели урока | 0 |
|  | Самостоятельная работа | 10 |
|  | Добавление кнопок в Reply-клавиатуре | 15 |
|  | Самостоятельная работа | 35 |
|  | Расположение кнопок в Inline-клавиатуре | 45 |
|  | Бескомпьютерная деятельность | 65 |
|  | Самостоятельная работа | 70 |
|  | Рефлексия | 85 |