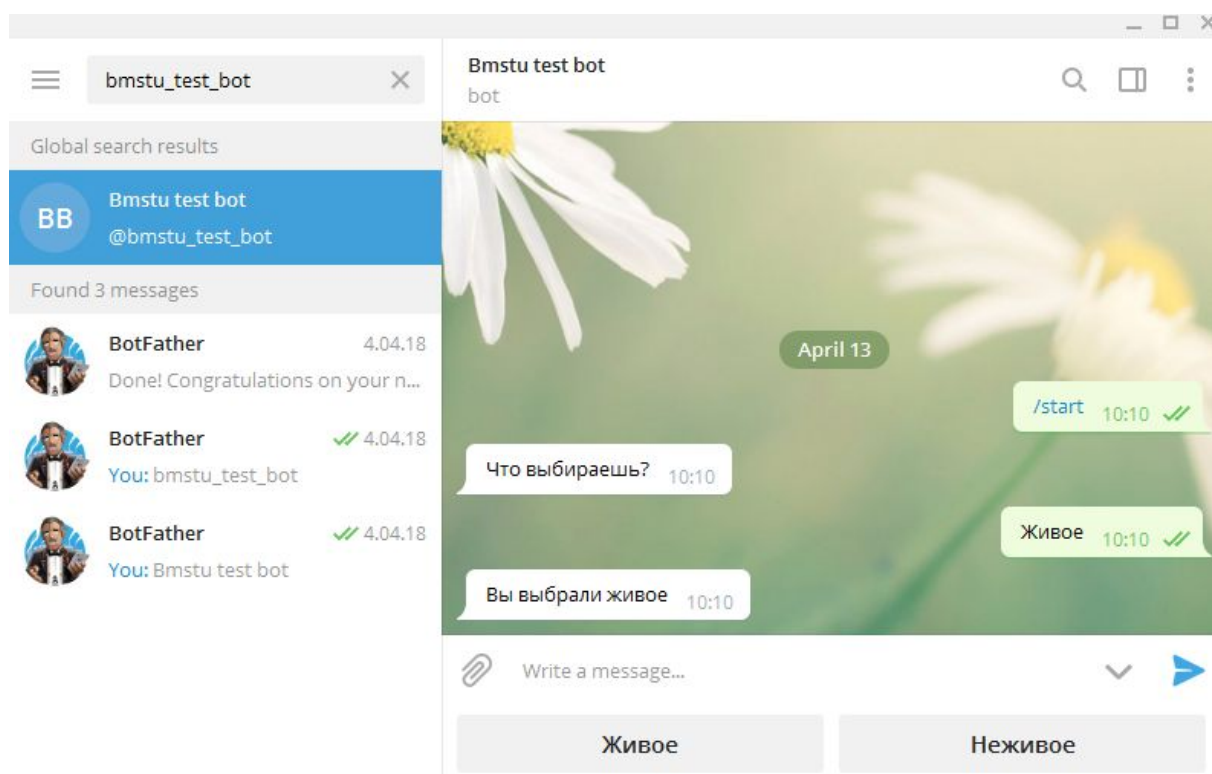


Лабораторная работа 8

Создание меню для бота для Telegram

Задание

Создать меню (клавиатуру) для вашего бота. Модифицируйте текст пунктов меню, не используйте примеры из теории ниже. На нажатие каждой клавиши должен выдаваться ответ, глубина вложенности вопросов и ответов должна быть не меньше двух, т.е. на первый ответ пользователя бот присылает новую клавиатуру с новыми вопросами. Всего различных клавиш должно быть не меньше восьми.



Дополнительно

Вывести счётчик нажатий каждой клавиши

Теория

Клавиатура у телеграм ботов является просто дублированием для ввода текста, однако она добавляет интерфейсу удобства. При нажатии любой клавиши на клавиатуре, по сути, вашему боту просто отправляется соответствующий текст.

Для разработки телеграм ботов мы будем использовать язык Python, среду разработки PyCharm и модуль pyTelegramBotAPI, автоматизирующий запросы к серверам API

телеграма. Документация к модулю доступна по адресу <https://github.com/eternnoir/pyTelegramBotAPI>

Порядок установки модулей и настройки PyCharm описан ниже.

Также для работы с прокси серверами необходимо установить модуль PySocks.

Для корректной работы с кириллицей подключим возможность работы с кодировкой utf-8:

```
# -*- coding: utf-8 -*-
```

Для начала создадим простого бота, который при запуске будет здороваться с пользователем. Для этого напишем следующий код:

```
import telebot

TOKEN = ''
bot = telebot.TeleBot(TOKEN)

@bot.message_handler(commands=['start'])

def start(m):
    msg = bot.send_message(m.chat.id, 'Привет!')

bot.polling(none_stop=True)
```

Для корректной работы воспользуемся прокси для доступа к серверам телеграм. В объявлениях необходимо добавить библиотеку apihelper, и ниже ввести данные прокси-сервера, специально созданного для ваших лабораторных:

```
from telebot import apihelper

apihelper.proxy = {
    'https': 'socks5h://bmstu:welcome_@94.103.82.136:1080'
}
```

Для добавления клавиатуры нам необходимо в шапку программы добавить строку

```
from telebot import types
```

А в функции start следует добавить следующий код:

```
keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True)
keyboard.add(*[types.KeyboardButton(name) for name in
['Живое', 'Неживое']])
msg = bot.send_message(m.chat.id, 'Что выбираешь?',
reply_markup=keyboard)
```

1. С помощью `types.ReplyKeyboardMarkup()` мы создаём объект нашей будущей клавиатуры, в скобках прописываются [нужные](#) параметры. `resize_keyboard=True` позволяет клавиатуре растягиваться на необходимую высоту вместо того, чтобы занимать всё пространство.
2. С помощью `.add()` мы добавляем кнопки в нашу клавиатуру. `types.KeyboardButton()` представляет собой объект самой кнопки, а в скобках указывается текст на ней. В коде выше я наполняю объект нашей клавиатуры *объектами* кнопок, заполняя их из списка ['Живое', 'Неживое'].
3. Последнее, что нужно сделать — привязать нашу клавиатуру к нужному сообщению, делается это при помощи параметра `reply_markup` в любом методе `send_....`

Теперь научим бота обрабатывать ответы:

```
def name (m):
    if m.text == 'Живое':
        bot.send_message(m.chat.id, 'Вы выбрали живое')
    elif m.text == 'Неживое':
        bot.send_message(m.chat.id, 'Вы выбрали неживое')
```

Для того, чтобы поймать любой текст, который нам отправляет пользователь, необходимо использовать соответствующий хендлер:

```
@bot.message_handler(content_types=["text"])
```

Клавиатуры позволяют отправлять целые сообщения по одному лишь нажатию пользователя. В кнопки можно записывать любые текстовые фразы, даже содержащие в себе Емоji. 🤔

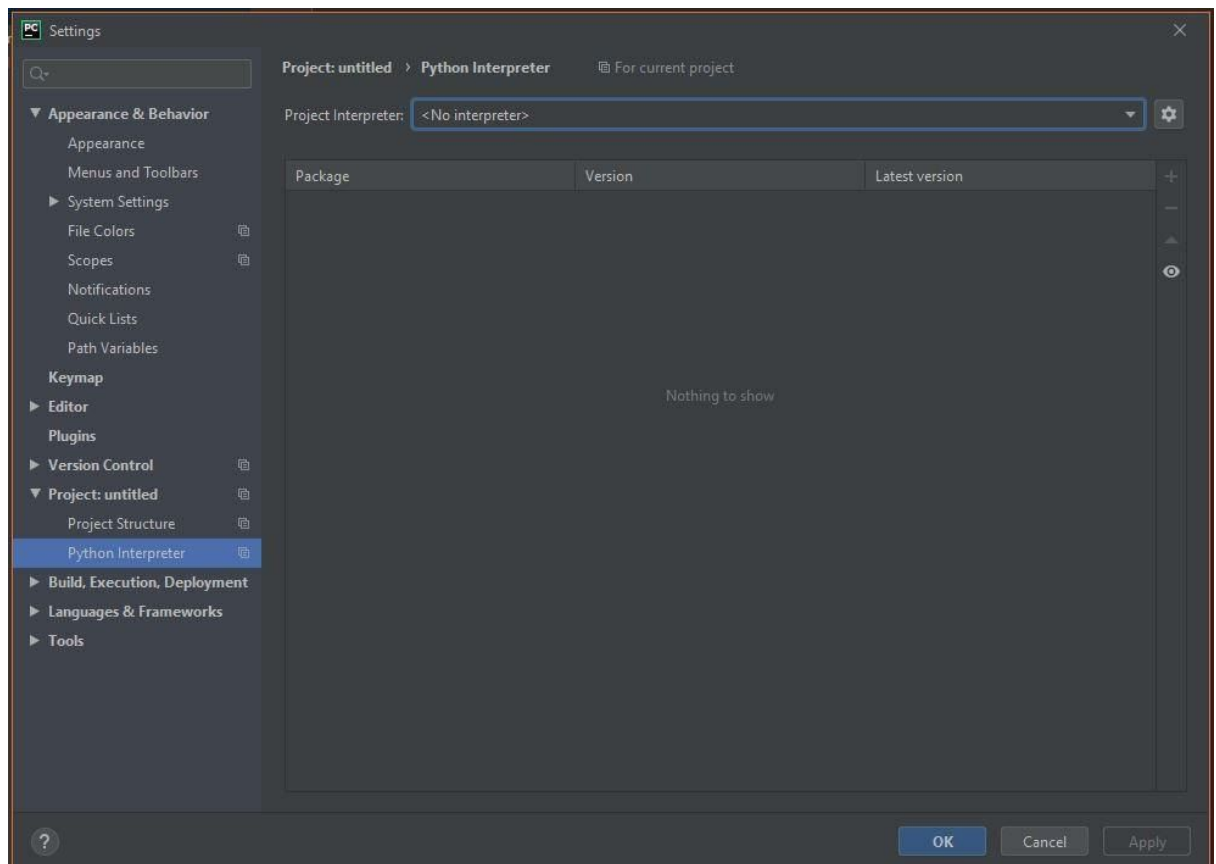
После отправки бот должен обработать полученную фразу, причем неважно, как именно.

У пользователя всегда есть остаётся возможность отправить сообщение при помощи обычной клавиатуры.

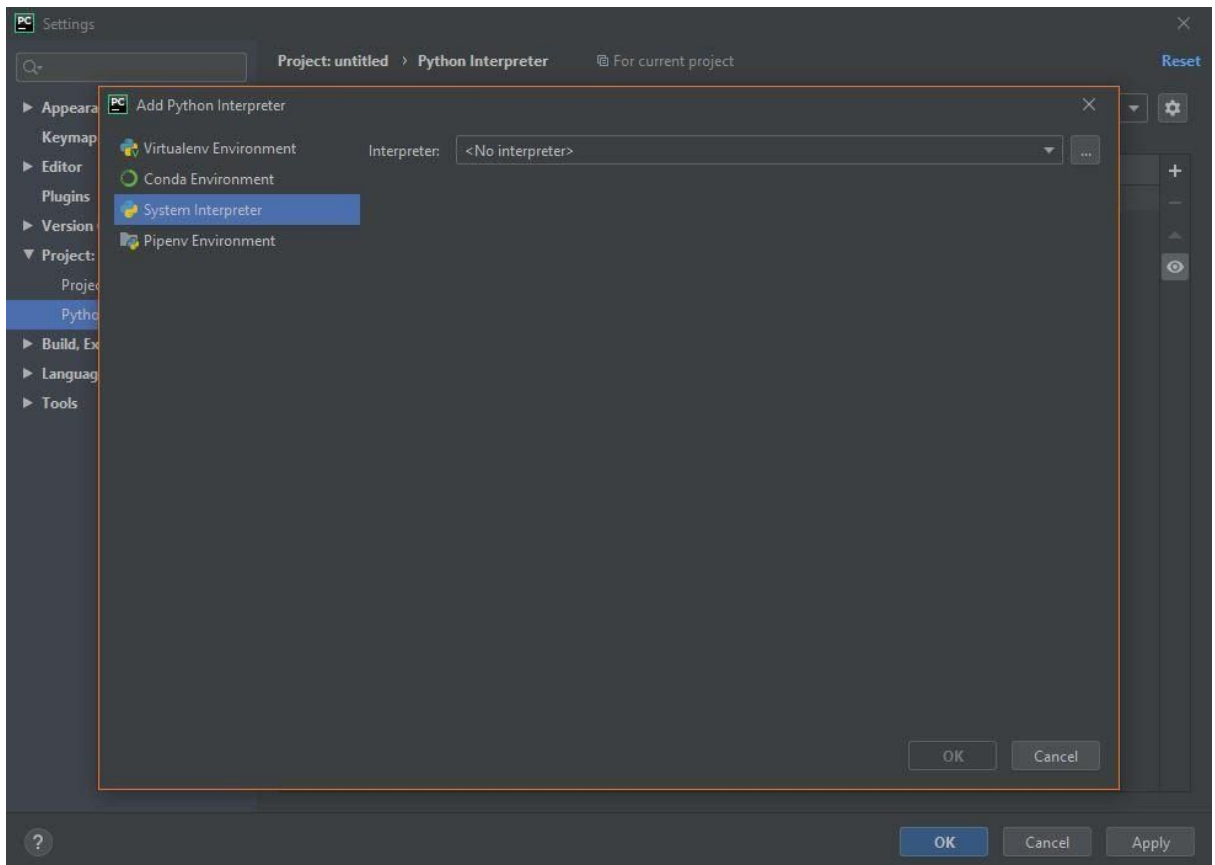
Для объекта кнопки есть два интересных параметра, принимающих логические значения — `request_contact` и `request_location`. Первый запрашивает телефонный номер пользователя, а второй его местоположение. [Узнать больше о параметрах и возможностях клавиатуры можно в официальной документации Telegram Bot API.](#)

Настройка PyCharm

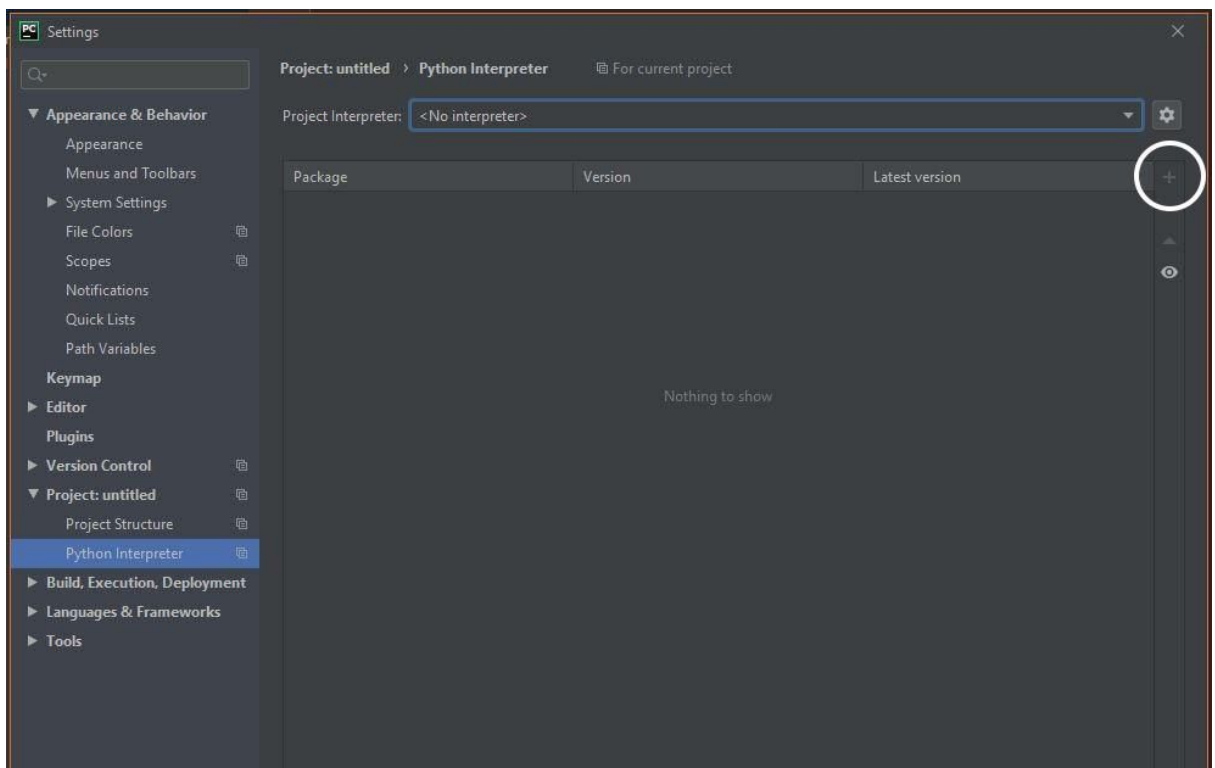
1. Устанавливаете PyCharm <https://www.jetbrains.com/ru-ru/pycharm/download>
2. Устанавливаете python <https://www.python.org/downloads>
3. Создаете новый проект File - New Project
4. При создании проекта выберите Existing interpreter и укажите путь, по которому установился питон. Под Windows, вероятнее всего, это будет `C:\Users\username\AppData\Programs\Python\python.exe`
5. В проекте создаете новый файл File - New... - Python file
6. Идёте в настройки проекта: File - Settings - Project: Untitled - Python Interpreter
7. Убедитесь, что в строчке Project Interpreter указан путь к питону. Если нет, нажмите рядом на шестеренку и укажите путь к интерпретатору питона. На скриншоте как раз путь к интерпретатору не указан



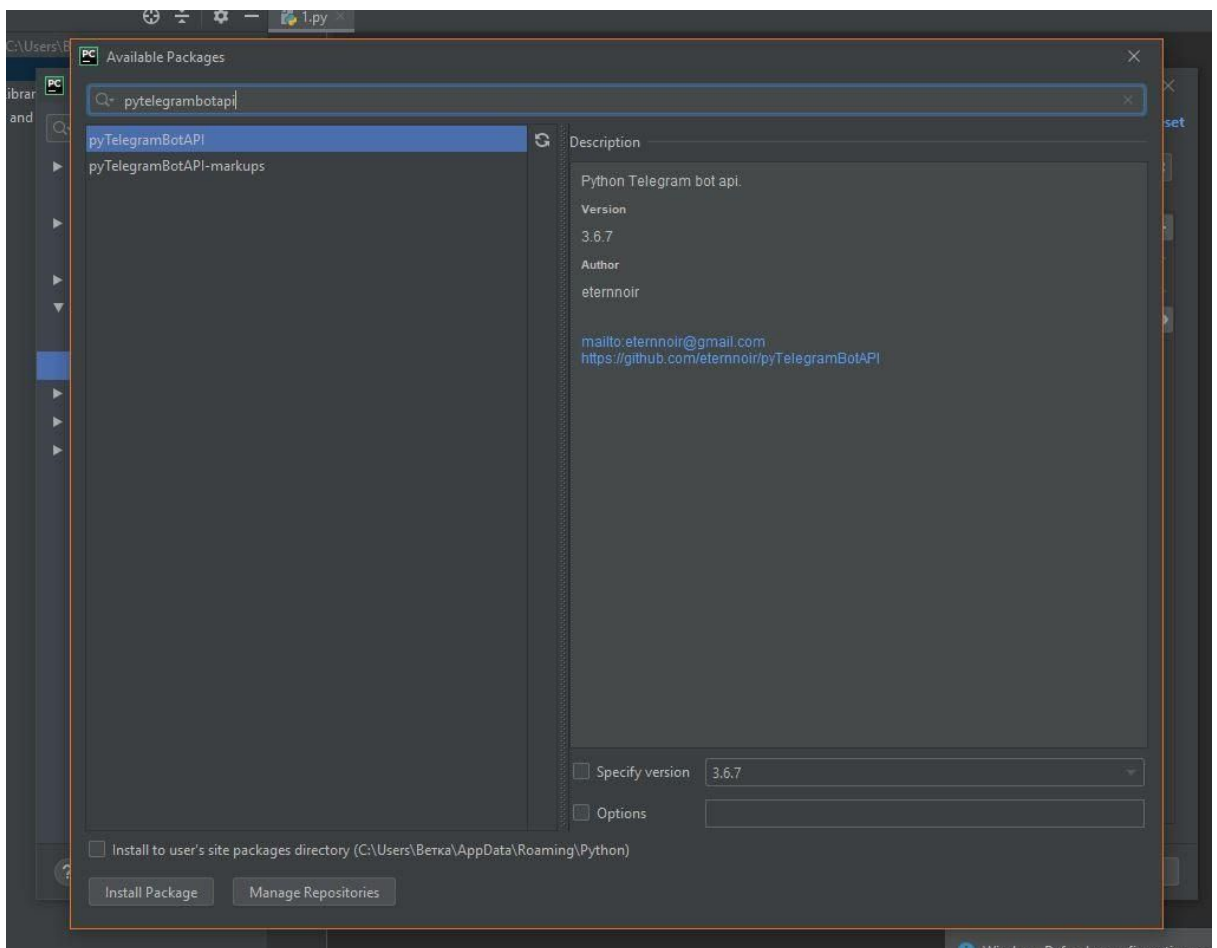
- В этом случае, нажав на шестеренку, указываем его в разделе System Interpreter



- После этого в предыдущем окне надо установить две библиотеки: `pyTelegramBotAPI` и `PySocks`. Нажмите на плюсики справа



10. Найдите нужные библиотеки и установите их



11. Готово, вы великолепны!