|  |  |  |
| --- | --- | --- |
| Защищено:  Гапанюк Ю. Е.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |  | Демонстрация:  Гапанюк Ю. Е.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

**Отчет по Лабораторной работе №6 по курсу**

**«Парадигмы и конструкции языков программирования»**

#### "Написание телеграмм-бота на языке Python"

|  |  |
| --- | --- |
| ИСПОЛНИТЕЛЬ: | Агапова Анна Денисовна |
| студент группы  ИУ5Ц-52Б | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | (подпись) |
| Гапанюк Ю.Е. | "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

Москва, МГТУ 2024

**СОДЕРЖАНИЕ**

[Практическое задание 3](#_Toc182428920)

[Листинг программы 4](#_Toc182428921)

[Результат работы программы 5](#_Toc182428922)

## Практическое задание

Разработайте простого бота для Telegram. Бот должен использовать функциональность создания кнопок.

В качестве темы для телеграмм-бота я решила взять переводчик.

Для демонстрации работы с библиотекой googletrans я использую автоопределение языка ввода, а также даю возможность пользователю выбрать конкретный режим.

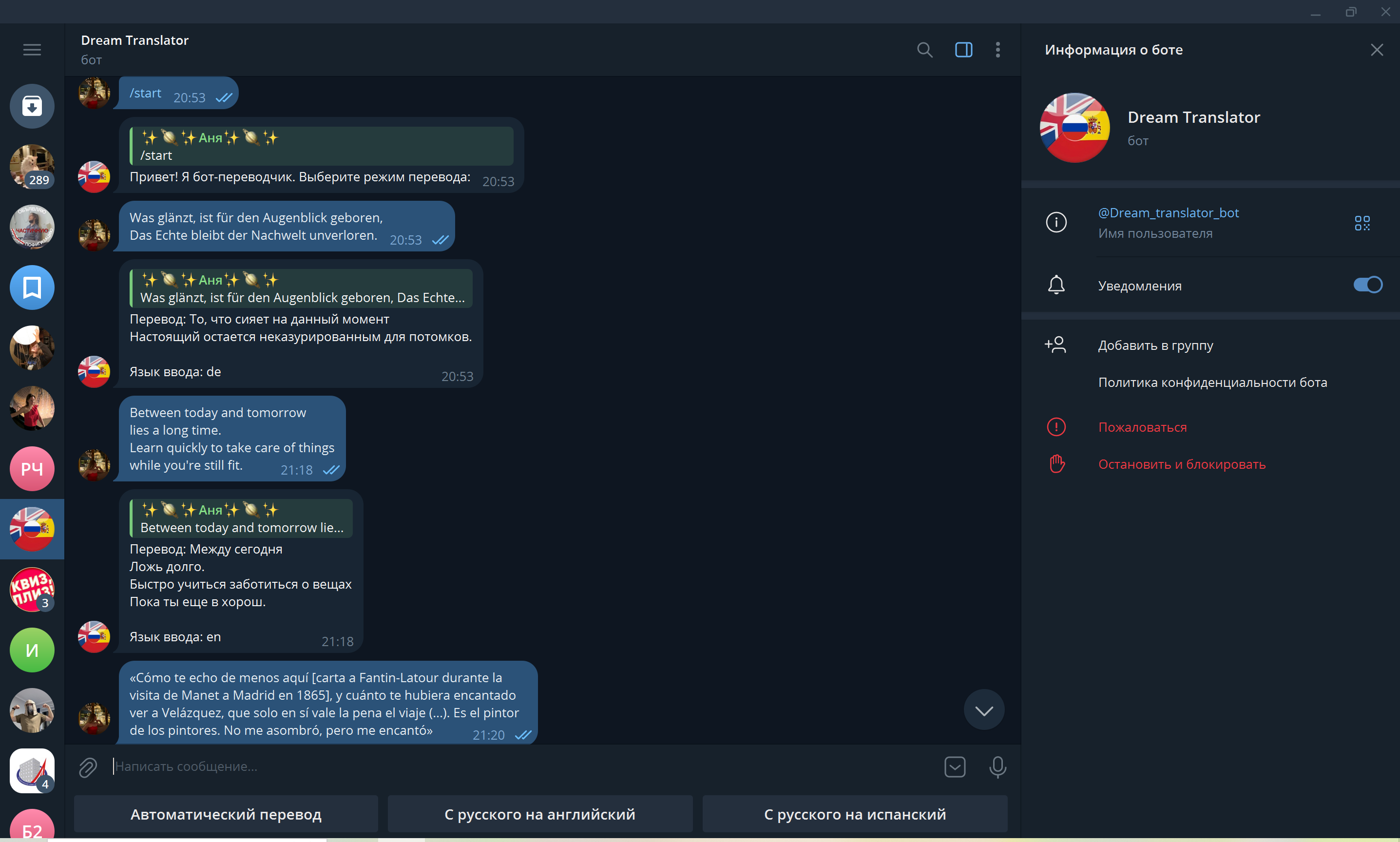
## Листинг программы

**main.py**

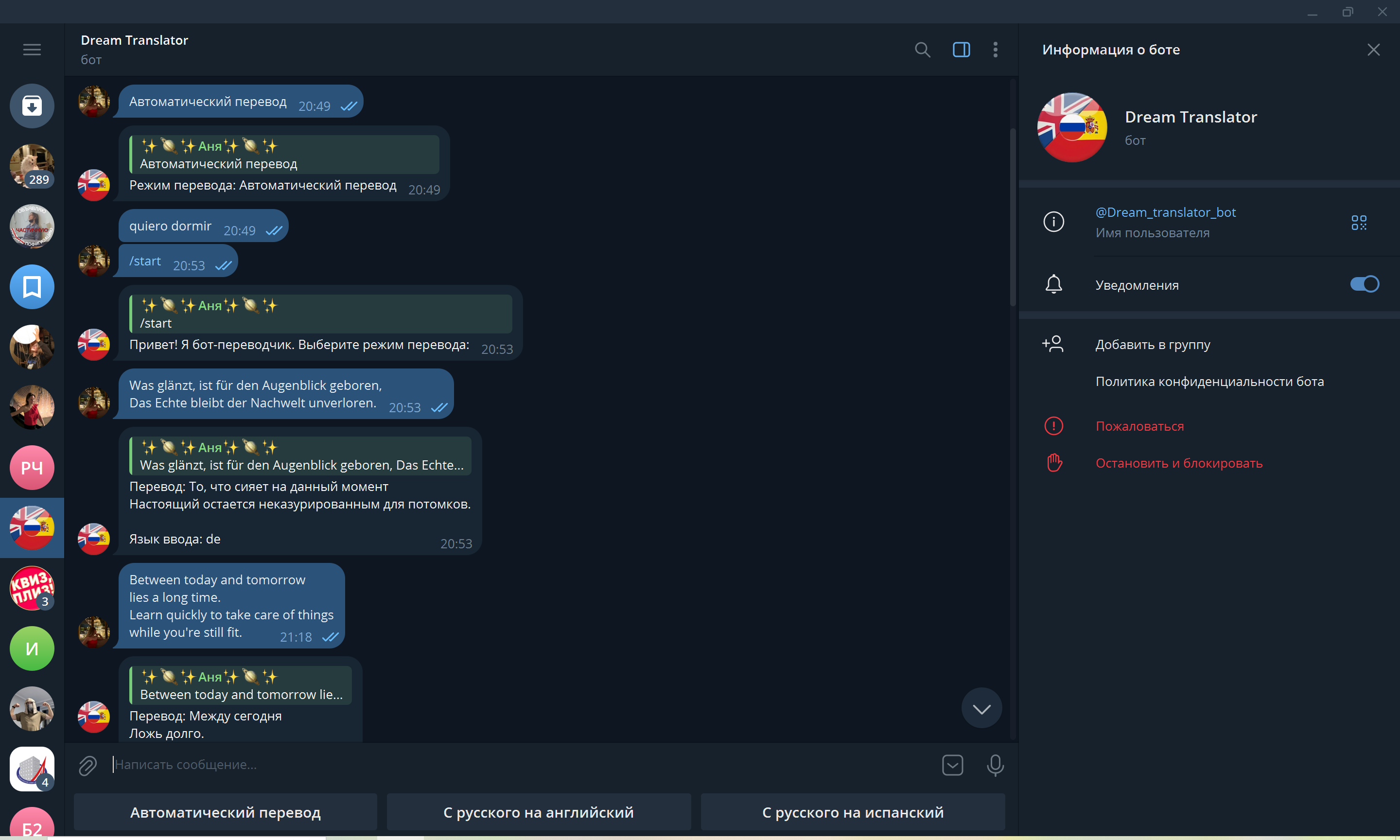
import telebot  
from googletrans import Translator  
from telebot.types import ReplyKeyboardMarkup, KeyboardButton  
  
  
bot = telebot.TeleBot('7015897860:AAGMy8s99Ps8O58NpgT2DddzcTMfvxHHc84')  
translator = Translator()  
  
  
keyboard = ReplyKeyboardMarkup(resize\_keyboard=True)  
button\_auto = KeyboardButton('Автоматический перевод')  
button\_ru\_to\_en = KeyboardButton('С русского на английский')  
button\_ru\_to\_es = KeyboardButton('С русского на испанский')  
keyboard.add(button\_auto, button\_ru\_to\_en, button\_ru\_to\_es)  
  
  
user\_mode = {}  
  
  
@bot.message\_handler(commands=['start'])  
def send\_welcome(message):  
 user\_id = message.chat.id  
 user\_mode[user\_id] = 'auto'  
 bot.reply\_to(message, "Привет! Я бот-переводчик. Выберите режим перевода:", reply\_markup=keyboard)  
  
  
@bot.message\_handler(func=lambda message: True)  
def translate\_message(message):  
 user\_id = message.chat.id  
 text = message.text  
  
 if text in ['Автоматический перевод', 'С русского на английский', 'С русского на испанский']:  
  
 if text == 'Автоматический перевод':  
 user\_mode[user\_id] = 'auto'  
 bot.reply\_to(message, "Режим перевода: Автоматический перевод")  
 elif text == 'С русского на английский':  
 user\_mode[user\_id] = 'ru\_to\_en'  
 bot.reply\_to(message, "Режим перевода: С русского на английский")  
 elif text == 'С русского на испанский':  
 user\_mode[user\_id] = 'ru\_to\_es'  
 bot.reply\_to(message, "Режим перевода: С русского на испанский")  
 return  
  
 try:  
 if user\_mode[user\_id] == 'auto':  
 detected = translator.detect(text)  
 if detected.lang is None:  
 bot.reply\_to(message, "Не удалось определить язык ввода.")  
 return  
 detected\_lang = detected.lang  
 translated = translator.translate(text, dest='ru')  
 bot.reply\_to(message, f"Перевод: {translated.text}\n\nЯзык ввода: {detected\_lang}")  
  
 elif user\_mode[user\_id] == 'ru\_to\_en':  
 translated = translator.translate(text, src='ru', dest='en')  
 bot.reply\_to(message, f"Перевод: {translated.text}")  
  
 elif user\_mode[user\_id] == 'ru\_to\_es':  
 translated = translator.translate(text, src='ru', dest='es')  
 bot.reply\_to(message, f"Перевод: {translated.text}")  
  
 except Exception as e:  
 bot.reply\_to(message, f"Произошла ошибка: {e}")  
  
  
bot.polling()

## Результат работы программы

Режим по умолчанию – **автоматический перевод**



При выборе **автоматического перевода**



При выборе перевода **с русского на английский** и **с русского на испанский**

