**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №6

«Разработка бота на основе конечного автомата для Telegram с использованием языка Python»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-31Б |  | преподаватель каф.ИУ5 |
| Кондрахин Сергей Сергеевич |  | Гапанюк Юрий  Евгеньевич |
|  |  |  |

Москва, 2021 г.

**Задание**

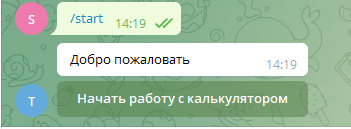
Разработайте бота для Telegram. Бот должен реализовывать конечный автомат их трех состояний.

**Текст программы**

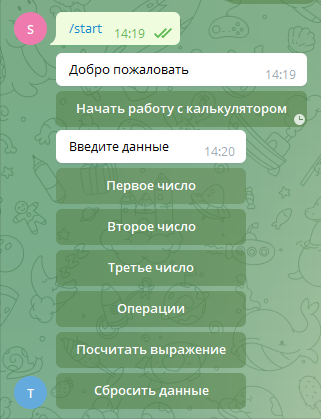
import telebot  
from telebot import types  
  
TOKEN = '\*\*\*'  
bot = telebot.TeleBot(TOKEN)  
global config  
config = ['Первое число', 'Второе число', 'Третье число', 'Операции', 'Посчитать выражение']  
global cases  
cases = ['first', 'second', 'third', 'action', 'result']  
global call  
call = ''  
global actions  
actions = ['plus', 'minus', 'multiplication', 'divide']  
global act  
act = ['+', '-', '\*', '/']  
  
@bot.message\_handler(commands='start')  
def start(message):  
 msg = 'Добро пожаловать'  
 markup = types.InlineKeyboardMarkup()  
 btn = types.InlineKeyboardButton('Начать работу с калькулятором', callback\_data='work')  
 markup.add(btn)  
 bot.send\_message(message.chat.id, msg, reply\_markup=markup)  
  
@bot.callback\_query\_handler(lambda message: message.data == 'work')  
def work(message):  
 msg = 'Введите данные'  
 markup = types.InlineKeyboardMarkup(row\_width=1)  
 for i in range(5):  
 btn = types.InlineKeyboardButton(f'{config[i]}', callback\_data=f'{cases[i]}')  
 markup.add(btn)  
 btn = types.InlineKeyboardButton('Сбросить данные', callback\_data='reset')  
 markup.add(btn)  
 bot.send\_message(message.message.chat.id, msg, reply\_markup=markup)  
  
@bot.callback\_query\_handler(lambda message: message.data == cases[0] or message.data == cases[1] or message.data == cases[2])  
def nums(message):  
 global call  
 call = cases[cases.index(message.data)]  
 msg = 'Введите число'  
 bot.send\_message(message.message.chat.id, msg)  
  
@bot.callback\_query\_handler(lambda message: message.data == cases[3])  
def action(message):  
 msg = 'Выберите операцию'  
 markup = types.InlineKeyboardMarkup(row\_width=2)  
 btn = types.InlineKeyboardButton('+', callback\_data='plus')  
 btn1 = types.InlineKeyboardButton('-', callback\_data='minus')  
 btn2 = types.InlineKeyboardButton('\*', callback\_data='multiplication')  
 btn3 = types.InlineKeyboardButton('/', callback\_data='divide')  
 markup.add(btn, btn1, btn2, btn3)  
 bot.send\_message(message.message.chat.id, msg, reply\_markup=markup)  
  
@bot.callback\_query\_handler(lambda message: message.data in actions)  
def act(message):  
 if message.data == actions[0]:  
 config[3] = actions[0]  
 elif message.data == actions[1]:  
 config[3] = actions[1]  
 elif message.data == actions[2]:  
 config[3] = actions[2]  
 else:  
 config[3] = actions[3]  
 markup = types.InlineKeyboardMarkup(row\_width=1)  
 msg = 'Введите данные'  
 for i in range(5):  
 if not config[i].isdigit() and not config[i] in actions:  
 btn = types.InlineKeyboardButton(f'{config[i]}', callback\_data=f'{cases[i]}')  
 markup.add(btn)  
 btn = types.InlineKeyboardButton('Сбросить данные', callback\_data='reset')  
 markup.add(btn)  
 bot.send\_message(message.message.chat.id, msg, reply\_markup=markup)  
  
@bot.callback\_query\_handler(lambda message: message.data == cases[4])  
def result(message):  
 if config[0].isdigit() and config[1].isdigit() and config[2].isdigit() and (config[3] in actions):  
 if config[3] == 'plus':  
 config[4] = float(config[0]) + float(config[1]) + float(config[2])  
 msg = f'{float(config[0])} + {float(config[1])} + {float(config[2])} = {config[4]}'  
 elif config[3] == 'minus':  
 config[4] = float(config[0]) - float(config[1]) - float(config[2])  
 msg = f'{float(config[0])} - {float(config[1])} - {float(config[2])} = {config[4]}'  
 elif config[3] == 'multiplication':  
 config[4] = float(config[0]) \* float(config[1]) \* float(config[2])  
 msg = f'{float(config[0])} \* {float(config[1])} \* {float(config[2])} = {config[4]}'  
 else:  
 if config[1] == '0' or config[2] == '0':  
 msg = 'Делить на ноль нельзя'  
 else:  
 config[4] = float(config[0]) / float(config[1]) / float(config[2])  
 msg = f'{float(config[0])} / {float(config[1])} / {float(config[2])} = {config[4]}'  
 markup = types.InlineKeyboardMarkup()  
 btn = types.InlineKeyboardButton('Сбросить данные', callback\_data='reset')  
 markup.add(btn)  
 bot.send\_message(message.message.chat.id, msg, reply\_markup=markup)  
 else:  
 msg = 'Недостаточно данных'  
 markup = types.InlineKeyboardMarkup()  
 for i in range(5):  
 if not config[i].isdigit() and not config[i] in actions:  
 btn = types.InlineKeyboardButton(f'{config[i]}', callback\_data=f'{cases[i]}')  
 markup.add(btn)  
 btn = types.InlineKeyboardButton('Сбросить данные', callback\_data='reset')  
 markup.add(btn)  
 bot.send\_message(message.message.chat.id, msg, reply\_markup=markup)  
  
@bot.callback\_query\_handler(lambda message: message.data == 'reset')  
def reset(message):  
 msg = 'Данные сброшены'  
 global config  
 config = ['Первое число', 'Второе число', 'Третье число', 'Операции', 'Посчитать выражение']  
 markup = types.InlineKeyboardMarkup()  
 btn = types.InlineKeyboardButton('Продолжить', callback\_data='work')  
 markup.add(btn)  
 bot.send\_message(message.message.chat.id, msg, reply\_markup=markup)  
  
@bot.message\_handler(content\_types='text')  
def text(message):  
 if (call == cases[0] or call == cases[1] or call == cases[2]) and message.text.isdigit():  
 if call == cases[0]:  
 config[0] = message.text  
 bot.send\_message(message.chat.id, f'Вы ввели первое число {float(message.text)}')  
 elif call == cases[1]:  
 config[1] = message.text  
 bot.send\_message(message.chat.id, f'Вы ввели второе число {float(message.text)}')  
 elif call == cases[2]:  
 config[2] = message.text  
 bot.send\_message(message.chat.id, f'Вы ввели третье число {float(message.text)}')  
 markup = types.InlineKeyboardMarkup(row\_width=1)  
 msg = 'Введите данные'  
 for i in range(5):  
 if not config[i].isdigit() and not config[i] in actions:  
 btn = types.InlineKeyboardButton(f'{config[i]}', callback\_data=f'{cases[i]}')  
 markup.add(btn)  
 btn = types.InlineKeyboardButton('Сбросить данные', callback\_data='reset')  
 markup.add(btn)  
 bot.send\_message(message.chat.id, msg, reply\_markup=markup)  
 if (call == cases[0] or call == cases[1] or call == cases[2]) and not message.text.isdigit():  
 msg = 'Ошибка, попробуйте еще раз'  
 bot.send\_message(message.chat.id, msg)  
  
bot.polling(none\_stop=True)

**Примеры выполнения программы**

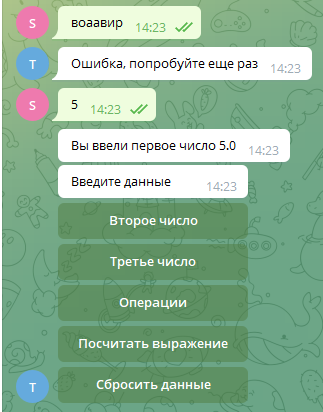
Команда /start



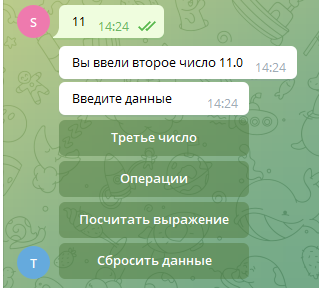
Кнопка «Начать работу с калькулятором»



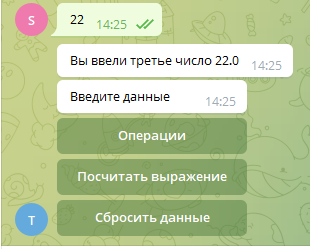
Кнопка «Первое число»



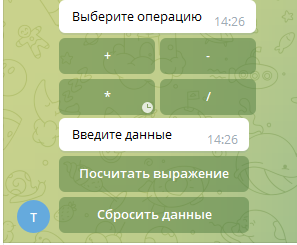
Кнопка «Второе число»



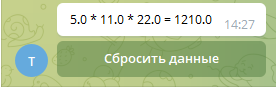
Кнопка «Третье число»



Кнопка «Операции»



Кнопка «Посчитать выражение»



Кнопка «Сбросить данные»

