Группа: ИУ5-31Б

Студент: Болотин Ю.А.

**Постановка задачи**

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

**Текст программы**

from aiogram import Bot, Dispatcher, types  
from aiogram.contrib.fsm\_storage.memory import MemoryStorage  
from aiogram.contrib.middlewares.logging import LoggingMiddleware  
from aiogram.dispatcher import FSMContext  
from aiogram.dispatcher.filters.state import State, StatesGroup  
from aiogram.utils import executor  
  
API\_TOKEN = ''  
  
# Инициализация бота и диспетчера  
bot = Bot(token=API\_TOKEN)  
storage = MemoryStorage()  
dp = Dispatcher(bot, storage=storage)  
  
  
# Определение состояний  
class Form(StatesGroup):  
 State1 = State() # Начальное состояние  
 State2 = State() # Второе состояние  
 State3 = State() # Третье состояние  
  
# Обработчик команды /start  
@dp.message\_handler(commands='start')  
async def cmd\_start(message: types.Message):  
 await Form.State1.set()  
 await message.reply("Вы находитесь в первом состоянии. Напишите что-нибудь, чтобы перейти во второе состояние.")  
  
# Обработчик для первого состояния  
@dp.message\_handler(state=Form.State1)  
async def process\_state1(message: types.Message):  
 await Form.State2.set()  
 await message.reply("Теперь вы во втором состоянии. Напишите что-нибудь, чтобы перейти в третье состояние.")  
  
# Обработчик для второго состояния  
@dp.message\_handler(state=Form.State2)  
async def process\_state2(message: types.Message):  
 await Form.State3.set()  
 await message.reply("Теперь вы в третьем состоянии. Напишите что-нибудь, чтобы вернуться в первое состояние.")  
  
# Обработчик для третьего состояния  
@dp.message\_handler(state=Form.State3)  
async def process\_state3(message: types.Message):  
 await Form.State1.set()  
 await message.reply("Вы вернулись в первое состояние. Напишите что-нибудь, чтобы перейти во второе состояние.")  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 executor.start\_polling(dp, skip\_updates=True)