**Московский государственный технический университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования» Отчет по домашнему заданию

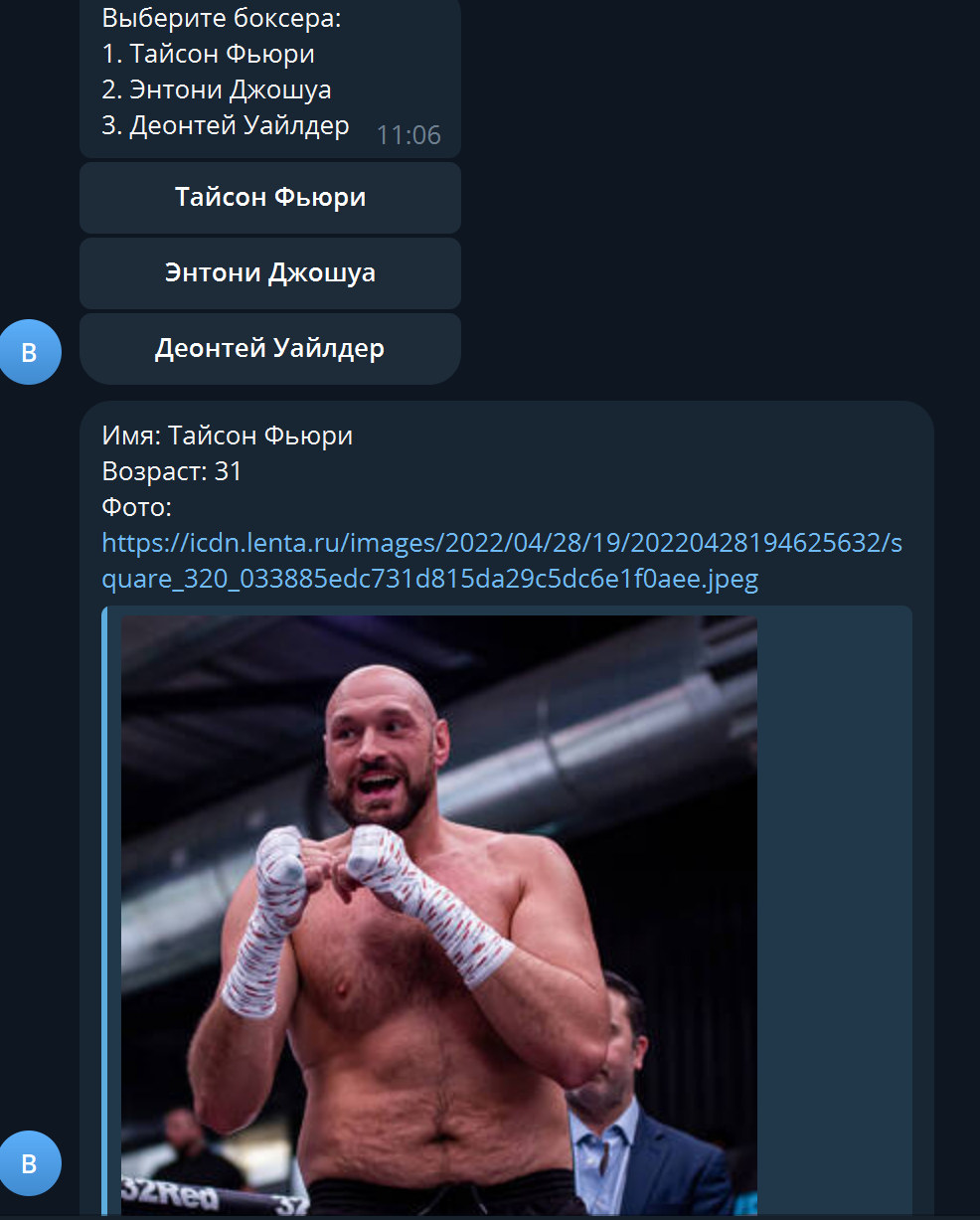
|  |  |
| --- | --- |
| Выполнил: | Проверил: |
| студент группы ИУ5-31Б | преподаватель каф. ИУ5 |
| Филенко Александр |  |
| Подпись и дата: | Подпись и дата: |

# Постановка задачи

**Задание:**

Разработать телеграмм бота по типу BoxRec, в котором можно будет просматривать информацию о лучших боксерах 3 весовых категорий.

# Результат

****

# Текст программы

import logging

from telegram import InlineKeyboardButton, InlineKeyboardMarkup

from telegram.ext import Updater, CommandHandler, CallbackQueryHandler

# Ваш токен бота

TOKEN = '6940234369:AAFE0CQjYCN3BIvX2Fq46BxfF83vW8ly35A'

# Список с информацией о боксерах в различных весовых категориях

boxers\_data = {

    'Легкий вес': [

        {'возраст': 28, 'имя': 'Майк Гарсия', 'фото': 'https://s-cdn.sportbox.ru/images/styles/upload/fp\_fotos/ae/69/91cfdea772f1f208e37b316a3ed97d5b5b045dab58135288879345.jpg'},

        {'возраст': 25, 'имя': 'Василий Ломаченко', 'фото': 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTtpzISXJ6\_18nqcu1QfI1wQLrYueXivV2aso-\_sk9Lrw&s'},

        {'возраст': 27, 'имя': 'Теофимо Лопес', 'фото': 'https://vringe.com/upload/medialibrary/faa/faa944b4ef364ed9b49230344baada01.jpg'},

    ],

    'Средний вес': [

        {'возраст': 30, 'имя': 'Канело Альварес', 'фото': 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTokqVxBHNQHEgyp8Y-qZGpkWXcodJjHv-S59VK7Cn3\_AgRzfS3oS00KhpOekXmsB6VZXU&usqp=CAU'},

        {'возраст': 29, 'имя': 'Деметриус Андраде', 'фото': 'https://vringe.com/upload/medialibrary/877/877c012ab0db6c4598f7ddabdfd3b7bb.jpg'},

        {'возраст': 28, 'имя': 'Джейми Мунгия', 'фото': 'https://img.prosports.kz/news/content//202206/162\_229e12ced4f7847c0a5d987956b38c3f.jpg'},

    ],

    'Тяжелый вес': [

        {'возраст': 31, 'имя': 'Тайсон Фьюри', 'фото': 'https://icdn.lenta.ru/images/2022/04/28/19/20220428194625632/square\_320\_033885edc731d815da29c5dc6e1f0aee.jpeg'},

        {'возраст': 32, 'имя': 'Энтони Джошуа', 'фото': 'https://vringe.com/upload/medialibrary/6c4/6c43d68eeacea4f237b07085d0a171fd.jpg'},

        {'возраст': 34, 'имя': 'Деонтей Уайлдер', 'фото': 'https://upload.wikimedia.org/wikipedia/commons/thumb/9/92/Deontay\_Wilder\_2018\_%28cropped%29.jpg/234px-Deontay\_Wilder\_2018\_%28cropped%29.jpg'},

    ],

    # Добавьте другие категории и данные по боксерам по аналогии

}

# Настройки логгирования

logging.basicConfig(format='%(asctime)s - %(name)s - %(levelname)s - %(message)s', level=logging.INFO)

logger = logging.getLogger(\_\_name\_\_)

# Словарь для отслеживания текущего выбранного боксера

selected\_boxer = {}

# Обработчик команды /start

def start(update, context):

    user = update.message.from\_user

    update.message.reply\_text(f"Привет, {user.first\_name}! Я бот с информацией о лучших боксерах. Выберите категорию:",

                              reply\_markup=generate\_categories\_keyboard())

# Обработчик нажатий на кнопки категорий

def category\_button\_click(update, context):

    query = update.callback\_query

    category = query.data.replace("category\_", "")

    boxers\_info = boxers\_data.get(category, [])

    context.user\_data['current\_category'] = category

    context.bot.send\_message(chat\_id=query.message.chat\_id, text=generate\_boxers\_list\_text(boxers\_info),

                             reply\_markup=generate\_boxers\_list\_keyboard(boxers\_info))

# Обработчик нажатий на кнопки боксеров

def boxer\_button\_click(update, context):

    query = update.callback\_query

    boxer\_index = int(query.data)

    category = context.user\_data.get('current\_category')

    boxers\_info = boxers\_data.get(category, [])

    if 0 <= boxer\_index < len(boxers\_info):

        selected\_boxer[query.from\_user.id] = boxers\_info[boxer\_index]

        message\_text = generate\_boxer\_info\_text(boxers\_info[boxer\_index])

        context.bot.send\_message(chat\_id=query.message.chat\_id, text=message\_text)

    else:

        context.bot.send\_message(chat\_id=query.message.chat\_id, text="Некорректный выбор боксера")

# Генерация клавиатуры с категориями

def generate\_categories\_keyboard():

    categories = list(boxers\_data.keys())

    keyboard = [[InlineKeyboardButton(category, callback\_data=f'category\_{category}')] for category in categories]

    return InlineKeyboardMarkup(keyboard)

# Генерация текста со списком боксеров

def generate\_boxers\_list\_text(boxers\_info):

    text = "Выберите боксера:\n"

    for i, boxer in enumerate(boxers\_info):

        text += f"{i + 1}. {boxer['имя']}\n"

    return text

# Генерация клавиатуры с боксерами

def generate\_boxers\_list\_keyboard(boxers\_info):

    keyboard = [[InlineKeyboardButton(boxer['имя'], callback\_data=str(i))] for i, boxer in enumerate(boxers\_info)]

    return InlineKeyboardMarkup(keyboard)

# Генерация текста с информацией о боксере

def generate\_boxer\_info\_text(boxer):

    text = f"Имя: {boxer['имя']}\nВозраст: {boxer['возраст']}\nФото: {boxer['фото']}\n\n"

    return text

# Основная функция бота

def main():

    updater = Updater(token=TOKEN, use\_context=True)

    dp = updater.dispatcher

    # Обработчики команд

    dp.add\_handler(CommandHandler("start", start))

    # Обработчик нажатий на кнопки категорий

    dp.add\_handler(CallbackQueryHandler(category\_button\_click, pattern='^category\_'))

    # Обработчик нажатий на кнопки боксеров

    dp.add\_handler(CallbackQueryHandler(boxer\_button\_click, pattern='^[0-9]+$'))

    # Запуск бота

    updater.start\_polling()

    updater.idle()

if \_\_name\_\_ == '\_\_main\_\_':

    main()