

Пояснительная записка MatematikBot

Используемые библиотеки :

```
import telegram
import logging
import sqlite3
```

Функции:

1. Команда **/start**:

Приветствие пользователя + инструкция к использованию

```
async def start(update, context) → None:
    """Отправляет сообщение когда получена команда /start
    да"""
    try:
        reply_keyboard = [["Учебник 📖", "Практика ✎"],
                           ["Дополнительно ⚙️"]]
        markup = ReplyKeyboardMarkup(reply_keyboard,
                                     one_time_keyboard=False, resize_keyboard=True)

        user = update.effective_user
        chat_id = update.message.chat_id
        photo_path = "db/start.gif"

        await update.message.reply_html(
            rf"Привет {user.mention_html()}! Меня зовут
            Никита. Я помогу тебе подготовиться к ЕГЭ по профильной
            математике💯", )

        await context.bot.send_photo(chat_id=chat_id,
                                     photo=open(photo_path, 'rb'))

        await update.message.reply_text(
            "У меня есть учебник с актуальной теорией и
```

тесты. Для навигации используй кнопки или меню в нижней части экрана."

```
reply_markup=markup, )
```

```
except Exception as ex:  
    print(f"ERROR {ex}")
```

2. Команда **/tutorial**:

Панель инлайн-кнопок

```
async def tutorial(update, context) → None:  
    """Отправляет сообщение кнопок когда получена команда  
    /tutorial"""
```

```
    keyboard = [  
        [  
            InlineKeyboardButton("1",  
callback_data='tutorial 1'),  
            InlineKeyboardButton("2",  
callback_data='tutorial 2'),  
            InlineKeyboardButton("3",  
callback_data='tutorial 3')  
        ],  
  
        [  
            InlineKeyboardButton("4",  
callback_data='tutorial 4'),  
            InlineKeyboardButton("5",  
callback_data='tutorial 5'),  
            InlineKeyboardButton("6",  
callback_data='tutorial 6')  
        ],  
  
        [  
            InlineKeyboardButton("7",  
callback_data='tutorial 7'),  
            InlineKeyboardButton("8",
```

```

callback_data='tutorial 8'),
            InlineKeyboardButton("9",
callback_data='tutorial 9')
        ],

        [
            InlineKeyboardButton("10",
callback_data='tutorial 10'),
            InlineKeyboardButton("11",
callback_data='tutorial 11'),
            InlineKeyboardButton("12",
callback_data='tutorial 12')
        ],
        [
            InlineKeyboardButton("Не помню, что в каком
задании 😞", callback_data='tutorial info')
        ]
    ]

    reply_markup = InlineKeyboardMarkup(keyboard)
    await update.message.reply_text('Выберите номер
задания ЕГЭ по математике:', reply_markup=reply_markup)

```

3. Команда **/practice**:

Панель инлайн-кнопок

```

async def practice(update, context) → None:
    """Отправляет сообщение кнопок когда получена команда
/practice"""

    keyboard = [
        [
            InlineKeyboardButton("Задание 1",
callback_data='practice 1'),
            InlineKeyboardButton("Задание 3",
callback_data='practice 3')
        ],
    ]

```

```

        [
            InlineKeyboardButton("Задание 4",
callback_data='practice 4'),
            InlineKeyboardButton("Задание 5",
callback_data='practice 5'),
            InlineKeyboardButton("Задание 6",
callback_data='practice 6')
        ],

        [
            InlineKeyboardButton("Задание 7",
callback_data='practice 7'),
            InlineKeyboardButton("Задание 8",
callback_data='practice 8'),
            InlineKeyboardButton("Задание 9",
callback_data='practice 9')
        ],

        [
            InlineKeyboardButton("Задание 10",
callback_data='practice 10'),
            InlineKeyboardButton("Задание 11",
callback_data='practice 11'),
            InlineKeyboardButton("Задание 12",
callback_data='practice 12')
        ]
    ]

    reply_markup = InlineKeyboardMarkup(keyboard)
    await update.message.reply_text("Выбери задание для
нарешивания 📌", reply_markup=reply_markup)

```

4. Функция обработки инлайн-кнопок

```

async def button(update, context) → None:
    """Функция обработки инлайн-кнопок"""

    global TASK

```

```

query = update.callback_query
chat_id = query.message.chat_id
query.answer()
task_type, number_task = query.data.split()

if number_task != "info":
    sqlite_connection =
sqlite3.connect('sqlite_python.db')
    cursor = sqlite_connection.cursor()
    sql_fetch_blob_query = f"""SELECT * from
data_theory WHERE id={number_task}"""
    cursor.execute(sql_fetch_blob_query)
    record = cursor.fetchall()

if task_type == "tutorial":
    if number_task == "info":
        text = """
1. Планиметрия
2. Векторы
3. Стереометрия
4. Простая Теория Вероятности
5. Сложная Теория Вероятности
6. Уравнение
7. Значение выражения
8. Анализ графика
9. Работа с формулами
10. Текстовая задача
11. Функции
12 Производная
        """
        await
query.message.reply_text(text)
    else:
        tutorial_photo_path = record[0][2]
        url = record[0][4]
        text = f"Теория по заданию №{number_task} ↓"

        await query.message.reply_text(text)
        await
context.bot.send_document(chat_id=chat_id,

```

```

document=open(tutorial_photo_path, "rb"))
        await query.message.reply_text(f"Ссылка на
видео: {url}")

    elif task_type == "practice":
        practice_file_path = record[0][1]

        text = f"Практика по заданию №{number_task} ↓"
        TASK = number_task

        keyboard = [
            [InlineKeyboardButton("Ответы:",
callback_data="answer 1")]
        ]
        reply_markup = InlineKeyboardMarkup(keyboard)
        await query.message.reply_text(text)
        await context.bot.send_document(chat_id=chat_id,
document=open(practice_file_path, "rb"),

reply_markup=reply_markup)
    elif task_type == "answer":
        answer_file_path = record[0][3]
        text = f"Ответы к заданию №{TASK}"
        await query.message.reply_text(text)
        await context.bot.send_document(chat_id=chat_id,
document=open(answer_file_path, "rb"))

```

5. Функция /main:

```

def main() → None:
    application =
Application.builder().token(BOT_TOKEN).read_timeout(30).b
uild()

    # Команды
    application.add_handler(CommandHandler("start",
start))

```

```
    application.add_handler(CommandHandler("tutorial",
tutorial))
    application.add_handler(CommandHandler("practice",
practice))
```

```
application.add_handler(CommandHandler("additionally",
additionally))
```

```
# Инлайн-кнопки
```

```
application.add_handler(CallbackQueryHandler(button))
```

```
# Сообщения
```

```
application.add_handler(MessageHandler(filters.Regex("Уче
бник 📖"), tutorial))
```

```
application.add_handler(MessageHandler(filters.Regex("Пра
ктика 📝"), practice))
```

```
application.add_handler(MessageHandler(filters.Regex("Доп
олнительно ⚙️"), additionally))
    application.add_handler(MessageHandler(filters.TEXT,
handler_response))
```

```
# Запуск бота
```

```
application.run_polling()
```