**ИСХОДНЫЙ КОД**

Телеграм-бот «OverEdit»

import telebot

from telebot import types

import sqlite3

from sqlite3 import Error

bot = telebot.TeleBot('6887025893:AAGwoLCvQDkV9xfcLm7Fej3TLbFDt\_WF7Lc')

bot.videos = []

bot.current\_video\_index = 0

bot.current\_theme = ""

bot.user\_id = 0

def create\_connection():

    try:

        conn = sqlite3.connect('over.db')

        return conn

    except Error as e:

        print(e)

        return None

def add\_user\_to\_database(conn, user\_id, first\_name, last\_name, username):

    try:

        cursor = conn.cursor()

        cursor.execute("INSERT INTO users (id, first\_name, last\_name, username) VALUES (?, ?, ?, ?)",

                       (user\_id, first\_name, last\_name, username))

        conn.commit()

    except Error as e:

        print(e)

@bot.message\_handler(commands=["start"])

def start(message):

    user\_id = message.from\_user.id

    first\_name = message.from\_user.first\_name

    last\_name = message.from\_user.last\_name or ''

    username = message.from\_user.username or ''

    conn = create\_connection()

    if conn:

        add\_user\_to\_database(conn, user\_id, first\_name, last\_name, username)

        conn.close()

    welcome\_message = f"Привет👋🏻, {first\_name} {last\_name}!\nДобро пожаловать в мир творчества и креатива с нашим Telegram-ботом OverEdit!" \

                      f"\nМы готовы помочь вам воплотить ваши идеи в жизнь, предоставляя материалы для ваших редакторских проектов." \

                      f"\nНезависимо от того, создаете вы видеоролики, фотоколлажи или другие творческие контенты, мы здесь, чтобы сделать ваш процесс редактирования проще и вдохновляющим."

    markup = types.InlineKeyboardMarkup()

    button1 = types.InlineKeyboardButton("Помощь❓", callback\_data='help')

    button2 = types.InlineKeyboardButton("Категории📋", callback\_data='categories')

    button3 = types.InlineKeyboardButton("Избранное❤", callback\_data='suggestion')

    markup.row(button1, button2, button3)

    bot.send\_message(message.chat.id, welcome\_message, reply\_markup=markup)

#------------------------------------------------------#

#               Обработка нажатия кнопок               #

#------------------------------------------------------#

@bot.callback\_query\_handler(func=lambda call: True)

def callback\_handler(call):

    if call.data == 'help':

        markup = create\_back\_keyboard()

        bot.send\_message(call.message.chat.id, "Вы выбрали Помощь❓ \n✦ Если вы хотите посмотреть видео нажмите на кнопку Категории📋"\

                         f"\n✦ Если вы хотите переключить видео нажмите на кнопку 'Вперед'"\

                         f"\n✦ Если вы хотите посмотреть предыдущее видео, то нажмите на кнопку 'Назад'"\

                         f"\n✦ Если вам понравилось данное видео и вы хотите его добавить в избранное, то нажмите на кнопку 'Добавить в избранное'"

                         f"\n✦ Чтобы посмотреть избранные видео нажмите на кнопку Избранное❤",

                         reply\_markup=markup)

        bot.answer\_callback\_query(call.id, text="", show\_alert=False)

    elif call.data == 'Back':

        start(call.message)

        bot.answer\_callback\_query(call.id, text="", show\_alert=False)

    elif call.data == 'categories':

        inline\_markup = types.InlineKeyboardMarkup()

        button\_video = types.InlineKeyboardButton("Видео🎞️", callback\_data='Video')

        button\_back = types.InlineKeyboardButton("Назад", callback\_data='Back')

        inline\_markup.row(button\_video)

        inline\_markup.row(button\_back)

        bot.send\_message(call.message.chat.id, "Выберите тип оверлея.", reply\_markup=inline\_markup)

        bot.answer\_callback\_query(callback\_query\_id=call.id, show\_alert=False)

    elif call.data == 'Video':

        inline\_markup = types.InlineKeyboardMarkup()

        button\_theme1 = types.InlineKeyboardButton("Бабочки🦋", callback\_data='Theme1')

        button\_theme2 = types.InlineKeyboardButton("Звезды✨", callback\_data='Theme2')

        button\_theme3 = types.InlineKeyboardButton("Фон🌌", callback\_data='Theme3')

        button\_back\_to\_categories = types.InlineKeyboardButton("Назад к выбору категории", callback\_data='BackToCategories')

        inline\_markup.row(button\_theme1, button\_theme2, button\_theme3)

        inline\_markup.row(button\_back\_to\_categories)

        bot.send\_message(call.message.chat.id, "Выберите тематику оверлея.", reply\_markup=inline\_markup)

        bot.answer\_callback\_query(callback\_query\_id=call.id, show\_alert=False)

    elif call.data == 'Video\_f':

        inline\_markup = types.InlineKeyboardMarkup()

        button\_theme1 = types.InlineKeyboardButton("Бабочки🦋", callback\_data='Theme1\_f')

        button\_theme2 = types.InlineKeyboardButton("Звезды✨", callback\_data='Theme2\_f')

        button\_theme3 = types.InlineKeyboardButton("Фон🌌", callback\_data='Theme3\_f')

        button\_back\_to\_categories = types.InlineKeyboardButton("Назад к выбору категории", callback\_data='BackToCategories')

        inline\_markup.row(button\_theme1, button\_theme2, button\_theme3)

        inline\_markup.row(button\_back\_to\_categories)

        bot.send\_message(call.message.chat.id, "Выберите тематику оверлея.", reply\_markup=inline\_markup)

        bot.answer\_callback\_query(callback\_query\_id=call.id, show\_alert=False)

    elif call.data == 'BackToCategories':

        inline\_markup = types.InlineKeyboardMarkup()

        button\_video = types.InlineKeyboardButton("Видео🎞️", callback\_data='Video')

        button\_back = types.InlineKeyboardButton("Назад", callback\_data='Back')

        inline\_markup.row(button\_video)

        inline\_markup.row(button\_back)

        bot.send\_message(call.message.chat.id, "Выберите тип оверлея.", reply\_markup=inline\_markup)

        bot.answer\_callback\_query(callback\_query\_id=call.id, show\_alert=False)

#---------------------------------------------------------------------------------#

#               Обработка нажатия кнопки "Назад к выбору категории"               #

#---------------------------------------------------------------------------------#

    elif call.data == 'BackToTheme':

        inline\_markup = types.InlineKeyboardMarkup()

        button\_theme1 = types.InlineKeyboardButton("Бабочки🦋", callback\_data='Theme1')

        button\_theme2 = types.InlineKeyboardButton("Звезды✨", callback\_data='Theme2')

        button\_theme3 = types.InlineKeyboardButton("Фон🌌", callback\_data='Theme3')

        button\_back\_to\_categories = types.InlineKeyboardButton("Назад к выбору категории", callback\_data='BackToCategories')

        inline\_markup.row(button\_theme1, button\_theme2, button\_theme3)

        inline\_markup.row(button\_back\_to\_categories)

        bot.send\_message(call.message.chat.id, "Выберите тематику оверлея.", reply\_markup=inline\_markup)

        bot.answer\_callback\_query(callback\_query\_id=call.id, show\_alert=False)

    elif call.data == 'NextVideo':

        if len(bot.videos) > 0:

            bot.current\_video\_index = (bot.current\_video\_index + 1) % len(bot.videos)

            markup = create\_back\_to\_theme\_keyboard()

            markup.add(types.InlineKeyboardButton("Добавить в избранное", callback\_data='AddToFavorites'))

            send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], markup,

                        bot.current\_theme, bot.user\_id, call.message.message\_id)

            bot.answer\_callback\_query(call.id, text="", show\_alert=False)

    elif call.data == 'PrevVideo':

        if len(bot.videos) > 0:

            bot.current\_video\_index = (bot.current\_video\_index - 1) % len(bot.videos)

            send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], create\_back\_to\_theme\_keyboard\_FAW(),

                        bot.current\_theme, bot.user\_id, call.message.message\_id)

            bot.answer\_callback\_query(call.id, text="", show\_alert=False)

    elif call.data == 'NextVideo\_f':

        if len(bot.videos) > 0:

            bot.current\_video\_index = (bot.current\_video\_index + 1) % len(bot.videos)

            markup = create\_back\_to\_theme\_keyboard\_FAW()

            send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], markup,

                        bot.current\_theme, bot.user\_id, call.message.message\_id)

            bot.answer\_callback\_query(call.id, text="", show\_alert=False)

    elif call.data == 'PrevVideo\_f':

        if len(bot.videos) > 0:

            bot.current\_video\_index = (bot.current\_video\_index - 1) % len(bot.videos)

            send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], create\_back\_to\_theme\_keyboard\_FAW(),

                        bot.current\_theme, bot.user\_id, call.message.message\_id)

            bot.answer\_callback\_query(call.id, text="", show\_alert=False)

#-----------------------------------------------------------------------#

#              Обработка нажатия кнопки "добавить в избранное"          #

#-----------------------------------------------------------------------#

    elif call.data == 'suggestion':

        markup = create\_favorite\_keyboard\_MENU2()

        bot.send\_message(call.message.chat.id, "Выберите тип добавленного оверлея", reply\_markup=markup)

        bot.answer\_callback\_query(callback\_query\_id=call.id, show\_alert=False)

#--------------------------------------------------------------#

#              Обработка нажатия кнопки "Бабочки"              #

#--------------------------------------------------------------#

    elif call.data == 'Theme1':

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                cursor.execute("SELECT id, file\_url FROM overlays WHERE theme = 'Butterflies' AND type = 'video'")

                videos = cursor.fetchall()

                if videos:

                    bot.videos = videos

                    bot.current\_theme = 'Butterflies'

                    markup = create\_back\_to\_theme\_keyboard()

                    markup.add(types.InlineKeyboardButton("Добавить в избранное", callback\_data='AddToFavorites'))

                    send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], markup)

                    bot.answer\_callback\_query(call.id, text="", show\_alert=False)

            finally:

                cursor.close()

                conn.close()

    elif call.data == 'Theme1\_f':

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                cursor.execute("SELECT id, file\_url FROM overlays WHERE theme = 'Butterflies' AND type = 'video' AND favorite = 1")

                videos = cursor.fetchall()

                if videos:

                    bot.videos = videos

                    bot.current\_theme = 'Butterflies'

                    markup = create\_back\_to\_theme\_keyboard\_FAW()

                    send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], markup)

                    bot.answer\_callback\_query(call.id, text="", show\_alert=False)

                else:

                    bot.send\_message(call.message.chat.id, "Нет избранных видео с темой 'Бабочки'")

            finally:

                cursor.close()

                conn.close()

#-----------------------------------------------#

#       Обработка нажатия кнопки "Звезды"       #

#-----------------------------------------------#

    elif call.data == 'Theme2':

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                cursor.execute("SELECT id, file\_url FROM overlays WHERE theme = 'Sparkles' AND type = 'video'")

                videos = cursor.fetchall()

                # Проверяем, есть ли видео

                if videos:

                    bot.videos = videos

                    bot.current\_theme = 'Sparkles'

                    markup = create\_back\_to\_theme\_keyboard()

                    markup.add(types.InlineKeyboardButton("Добавить в избранное", callback\_data='AddToFavorites'))

                    send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], markup)

                    bot.answer\_callback\_query(call.id, text="", show\_alert=False)

            finally:

                cursor.close()

                conn.close()

    elif call.data == 'Theme2\_f':

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                cursor.execute("SELECT id, file\_url FROM overlays WHERE theme = 'Sparkles' AND type = 'video' AND favorite = 1")

                videos = cursor.fetchall()

                if videos:

                    bot.videos = videos

                    bot.current\_theme = 'Sparkles'

                    markup = create\_back\_to\_theme\_keyboard\_FAW()

                    send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], markup)

                    bot.answer\_callback\_query(call.id, text="", show\_alert=False)

                else:

                    bot.send\_message(call.message.chat.id, "Нет избранных видео с темой 'Бабочки'")

            finally:

                cursor.close()

                conn.close()

#-----------------------------------------------#

#       Обработка нажатия кнопки "Фон"          #

#-----------------------------------------------#

    elif call.data == 'Theme3':

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                cursor.execute("SELECT id, file\_url FROM overlays WHERE theme = 'Background' AND type = 'video'")

                videos = cursor.fetchall()

                if videos:

                    bot.videos = videos

                    bot.current\_theme = 'Background'

                    markup = create\_back\_to\_theme\_keyboard()

                    markup.add(types.InlineKeyboardButton("Добавить в избранное", callback\_data='AddToFavorites'))

                    send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], markup)

                    bot.answer\_callback\_query(call.id, text="", show\_alert=False)

            finally:

                cursor.close()

                conn.close()

    elif call.data == 'Theme3\_f':

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                cursor.execute("SELECT id, file\_url FROM overlays WHERE theme = 'Background' AND type = 'video' AND favorite = 1")

                videos = cursor.fetchall()

                if videos:

                    bot.videos = videos

                    bot.current\_theme = 'Background'

                    markup = create\_back\_to\_theme\_keyboard\_FAW()

                    send\_video(call.message.chat.id, bot.videos[bot.current\_video\_index][1], markup)

                    bot.answer\_callback\_query(call.id, text="", show\_alert=False)

                else:

                    bot.send\_message(call.message.chat.id, "Нет избранных видео с темой 'Бабочки'")

            finally:

                cursor.close()

                conn.close()

#------------------------------------------------------#

#     Обработка нажатия кнопки "Добавить в избранное"  #

#------------------------------------------------------#

    elif call.data == 'AddToFavorites':

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                video\_url = bot.videos[bot.current\_video\_index][1]

                cursor.execute("SELECT id FROM overlays WHERE file\_url = ? AND favorite = 1", (video\_url,))

                already\_favorited = cursor.fetchone()

                if already\_favorited:

                    bot.send\_message(call.message.chat.id, "Это видео уже добавлено в избранное!")

                else:

                    cursor.execute("UPDATE overlays SET favorite = 1 WHERE file\_url = ?", (video\_url,))

                    conn.commit()

                    bot.send\_message(call.message.chat.id, "Видео добавлено в избранное!")

                    markup = create\_back\_to\_theme\_keyboard()

                    markup.add(types.InlineKeyboardButton("Удалить из избранного", callback\_data='RemoveFromFavorites'))

                    send\_video(call.message.chat.id, video\_url, markup, bot.current\_theme, bot.user\_id, call.message.message\_id)

            finally:

                cursor.close()

                conn.close()

#------------------------------------------------------#

#     Обработка нажатия кнопки "Удалить из избранного" #

#------------------------------------------------------#

    elif call.data == 'RemoveFromFavorites':

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                video\_url = bot.videos[bot.current\_video\_index][1]

                cursor.execute("SELECT id FROM overlays WHERE file\_url = ? AND favorite = 0", (video\_url,))

                already\_removed = cursor.fetchone()

                if already\_removed:

                    bot.send\_message(call.message.chat.id, "Это видео уже удалено из избранного!")

                else:

                    cursor.execute("UPDATE overlays SET favorite = 0 WHERE file\_url = ?", (video\_url,))

                    conn.commit()

                    bot.send\_message(call.message.chat.id, "Видео удалено из избранного!")

                    markup = create\_back\_to\_theme\_keyboard()

                    markup.add(types.InlineKeyboardButton("Добавить в избранное", callback\_data='AddToFavorites'))

                    send\_video(call.message.chat.id, video\_url, markup, bot.current\_theme, bot.user\_id, call.message.message\_id)

            finally:

                cursor.close()

                conn.close()

#--------------------------------#

#     Функция отправки видео     #

#--------------------------------#

def send\_video(chat\_id, video\_url, markup, theme=None, user\_id=None, message\_id=None):

    caption = f"Видео с темой {theme}" if theme else None

    if message\_id:

        bot.edit\_message\_media(chat\_id=chat\_id, message\_id=message\_id,

                               media=types.InputMediaVideo(video\_url, caption=caption), reply\_markup=markup)

    else:

        bot.send\_video(chat\_id, video\_url, caption=caption, reply\_markup=markup)

    if user\_id and theme:

        conn = create\_connection()

        if conn:

            try:

                cursor = conn.cursor()

                cursor.execute("INSERT INTO user\_overlays (user\_id, overlay\_id) VALUES (?, ?)",

                               (user\_id, bot.videos[bot.current\_video\_index][0]))

                conn.commit()

            finally:

                cursor.close()

                conn.close()

#--------------------------------------------#

#     Функция отправки видео в избранное     #

#--------------------------------------------#

def send\_favorite\_videos(chat\_id, user\_id):

    conn = create\_connection()

    if conn:

        try:

            cursor = conn.cursor()

            cursor.execute("""

                SELECT o.file\_url

                FROM user\_overlays uo

                INNER JOIN overlays o ON uo.overlay\_id = o.id

                WHERE uo.user\_id = ? AND o.favorite = 1

            """, (user\_id,))

            favorite\_videos = cursor.fetchall()

            if favorite\_videos:

                markup = create\_back\_to\_theme\_keyboard()

                markup.add(types.InlineKeyboardButton("Добавить в избранное", callback\_data='AddToFavorites'))

                for video\_url in favorite\_videos:

                    send\_video(chat\_id, video\_url[0], markup, "Избранное", user\_id)

            else:

                bot.send\_message(chat\_id, "В вашем избранном пока нет видео.")

        finally:

            cursor.close()

            conn.close()

def create\_back\_keyboard():

    markup = types.InlineKeyboardMarkup()

    button4 = types.InlineKeyboardButton("Назад", callback\_data='Back')

    markup.row(button4)

    return markup

def create\_back\_to\_categories\_keyboard():

    markup = types.InlineKeyboardMarkup()

    button\_back\_to\_categories = types.InlineKeyboardButton("Назад к выбору категории", callback\_data='BackToCategories')

    markup.row(button\_back\_to\_categories)

    return markup

def create\_favorite\_keyboard1():

    markup = types.InlineKeyboardMarkup()

    button\_theme1 = types.InlineKeyboardButton("Бабочки🦋", callback\_data='Theme1\_f')

    button\_theme2 = types.InlineKeyboardButton("Звезды✨", callback\_data='Theme2\_f')

    button\_theme3 = types.InlineKeyboardButton("Фон🌌", callback\_data='Theme3\_f')

    button\_back\_to\_categories = types.InlineKeyboardButton("Назад к выбору категории", callback\_data='BackToCategories')

    markup.row(button\_theme1)

    markup.row(button\_theme2)

    markup.row(button\_theme3)

    markup.row(button\_back\_to\_categories)

    return markup

def create\_back\_to\_theme\_keyboard():

    markup = types.InlineKeyboardMarkup()

    button\_back\_to\_theme = types.InlineKeyboardButton("Назад к выбору темы", callback\_data='BackToTheme')

    button\_prev\_video = types.InlineKeyboardButton("Назад", callback\_data='PrevVideo')

    button\_next\_video = types.InlineKeyboardButton(f"{bot.current\_video\_index + 1}/{len(bot.videos)} Вперед", callback\_data='NextVideo')

    markup.row(button\_back\_to\_theme)

    markup.row(button\_prev\_video, button\_next\_video)

    return markup

def create\_back\_to\_theme\_keyboard\_FAW():

    markup = types.InlineKeyboardMarkup()

    button\_back\_to\_theme = types.InlineKeyboardButton("Назад к выбору темы", callback\_data='BackToTheme')

    button\_prev\_video = types.InlineKeyboardButton("Назад", callback\_data='PrevVideo\_f')

    button\_next\_video = types.InlineKeyboardButton(f"{bot.current\_video\_index + 1}/{len(bot.videos)} Вперед", callback\_data='NextVideo\_f')

    markup.row(button\_back\_to\_theme)

    markup.row(button\_prev\_video, button\_next\_video)

    return markup

def create\_favorite\_keyboard\_MENU2():

    markup = types.InlineKeyboardMarkup()

    button\_video = types.InlineKeyboardButton("Видео🎞️", callback\_data='Video\_f')

    button\_back = types.InlineKeyboardButton("Назад", callback\_data='Back')

    markup.row(button\_video)

    markup.row(button\_back)

    return markup

def create\_favorite\_keyboard():

    markup = types.InlineKeyboardMarkup()

    button\_video = types.InlineKeyboardButton("Видео🎞️", callback\_data='Video')

    button\_picture = types.InlineKeyboardButton("Картинки🌆", callback\_data='Picture')

    button\_back = types.InlineKeyboardButton("Назад", callback\_data='Back')

    markup.row(button\_video, button\_picture, button\_back)

    return markup

while True:

    try:

        bot.polling(none\_stop=True)

    except Exception as e:

        print(f"Error: {e}")