import asyncio

from telegram import ReplyKeyboardMarkup, Update

from telegram.ext import Application, CommandHandler, MessageHandler, filters, ContextTypes

# Токени боти худро ворид кунед

BOT\_TOKEN = "7903594220:AAEQgpgweP\_H0XSZO7T8Il33IlrsyrdX-W0"

# ID чати Telegram-и шумо

ADMIN\_CHAT\_ID = 5607622404

# Функсияи "start"

async def start(update: Update, context: ContextTypes.DEFAULT\_TYPE):

keyboard = [

["✨ Как сделать заказ", "🚚 Доставка"],

["↔️ Обмен и Возврат", "💸 Оплата"],

["📲 ВК Сообщество", "📲 Telegram"],

["📞 Связь со мной", "📝 Отзывы"],

["⬅️ Назад"]

]

reply\_markup = ReplyKeyboardMarkup(keyboard, resize\_keyboard=True)

await update.message.reply\_text(

"Добро пожаловать! Выберите нужное меню:", reply\_markup=reply\_markup

)

# Функсия барои "Как сделать заказ"

async def order\_info(update: Update, context: ContextTypes.DEFAULT\_TYPE):

text = """✨ Как сделать заказ?

1. Присылайте нам фото товара, который хотите заказать, и желаемое количество.

2. Мы проконсультируем вас по наличию изделий.

3. Составим чек на оплату.

4. После оплаты сразу отправим заказ (или отложим до вашего приезда к нам в магазин).

5. Наши данные для оплаты:

✅ Сбербанк: +7(925)-111-50-97

✅ Имя: Амонуллох Балачонович М

✅ Тинькофф

6. После оплаты пришлите фото чека.

📸 После отправки фото чека заказ будет подтвержден, и вы получите уведомление."""

await update.message.reply\_text(text)

# Функсия барои "Доставка"

async def delivery\_info(update: Update, context: ContextTypes.DEFAULT\_TYPE):

text = """🚚 Доставка: по ВСЕЙ России и СНГ.

Отправить заказ можем любой транспортной компанией:

СДЭК, ПЭК, Деловые линии, Почта России, Байкал-сервис, Виктория и т.д.

Сроки и тарифы доставки до вашего города вы можете узнать, позвонив на горячую линию ТК или на сайте ТК!

🚌 Отправляем автобусами!

Для этого вы должны предоставить данные водителя, время отъезда и номер стоянки!

От нас грузчик 100-200₽ от веса посылки и до ТК 100-300₽ от веса!"""

await update.message.reply\_text(text)

# Функсия барои "Оплата"

async def payment\_info(update: Update, context: ContextTypes.DEFAULT\_TYPE):

text = """💸 Оплата

Оплату мы принимаем на карту тиныкоф .

Онлайн расчёт. Для оплаты менеджер выставляет счёт и отправляет вам ссылку на оплату."""

await update.message.reply\_text(text)

# Функсия барои "Обмен и Возврат"

async def exchange\_return\_info(update: Update, context: ContextTypes.DEFAULT\_TYPE):

text = """↔️ Обмен

В случае фабричного брака в течение 7 дней после выкупа.

🛑 Возврат

К сожалению, возврата нет."""

await update.message.reply\_text(text)

# Функсия барои "ВК Сообщество"

async def vk\_community(update: Update, context: ContextTypes.DEFAULT\_TYPE):

text = "📲 Наше ВК Сообщество: https://vk.com/fashionstyl01\nПодписывайтесь, чтобы быть в курсе новых товаров!"

await update.message.reply\_text(text)

# Функсия барои "Telegram"

async def telegram\_info(update: Update, context: ContextTypes.DEFAULT\_TYPE):

text = "📲 Наш Telegram канал: https://t.me/italiakorea01\nПодписывайтесь, чтобы не пропустить новинки!"

await update.message.reply\_text(text)

# Функсия барои "Связь со мной"

async def contact\_info(update: Update, context: ContextTypes.DEFAULT\_TYPE):

text = """📞 Для связи с нами, пожалуйста, используйте следующие контактные данные:

WhatsApp: +7 (925) 111-50-97

Telegram: @Sadovodru6112

ВК: https://vk.com/i606703932

"""

await update.message.reply\_text(text)

# Функсия барои "Отзывы"

async def feedback(update: Update, context: ContextTypes.DEFAULT\_TYPE):

text = "📝 Ваши отзывы очень важны для нас! Пожалуйста, оставьте ваш отзыв и поделитесь опытом."

await update.message.reply\_text(text)

# Функсия барои "Назад"

async def back(update: Update, context: ContextTypes.DEFAULT\_TYPE):

keyboard = [

["✨ Как сделать заказ", "🚚 Доставка"],

["↔️ Обмен и Возврат", "💸 Оплата"],

["📲 ВК Сообщество", "📲 Telegram"],

["📞 Связь со мной", "📝 Отзывы"]

]

reply\_markup = ReplyKeyboardMarkup(keyboard, resize\_keyboard=True)

await update.message.reply\_text(

"Вы вернулись в основное меню. Выберите нужное действие:", reply\_markup=reply\_markup

)

# Функсия барои "Сделать заказ"

async def place\_order(update: Update, context: ContextTypes.DEFAULT\_TYPE):

# Маълумоти фиристодаи корбар

user\_data = update.message.text

# Хабар ба корбар

await update.message.reply\_text("Мы приняли ваш заказ! В ближайшее время с вами свяжется наш менеджер.")

# Хабар ба Telegram (чати канала ё ID чат)

order\_info = f"Новый заказ от {update.message.from\_user.username}:\n{user\_data}"

# Фиристодани хабари фармоиш ба чати Telegram-и шумо

await context.bot.send\_message(chat\_id=ADMIN\_CHAT\_ID, text=order\_info)

# Хомӯш кардани бот пас аз 2 соат

async def shutdown\_bot():

print("Бот пас аз 2 соат хомӯш мешавад.")

await bot.close()

# Вақти хомӯш кардани бот

async def schedule\_shutdown():

await asyncio.sleep(2 \* 60 \* 60) # 2 соат дар сонияхо

await shutdown\_bot()

# Танзими боти Telegram

def main():

app = Application.builder().token(BOT\_TOKEN).build()

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

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^✨ Как сделать заказ$"), order\_info))

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^🚚 Доставка$"), delivery\_info))

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^💸 Оплата$"), payment\_info))

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^↔️ Обмен и Возврат$"), exchange\_return\_info)) # Илова кардани функсия

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^📲 ВК Сообщество$"), vk\_community)) # Илова кардани функсия

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^📲 Telegram$"), telegram\_info)) # Илова кардани функсия

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^📞 Связь со мной$"), contact\_info)) # Илова кардани функсия

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^📝 Отзывы$"), feedback)) # Илова кардани функсия

app.add\_handler(MessageHandler(filters.TEXT & filters.Regex("^⬅️ Назад$"), back)) # Илова кардани функсия

app.add\_handler(MessageHandler(filters.TEXT, place\_order)) # Функсияи нав барои фармоиш

print("Бот успешно запущен!")

app.run\_polling()

# Вақти хомӯш кардан

asyncio.run(schedule\_shutdown())

if \_\_name\_\_ == "\_\_main\_\_":

main()