

```
import os import requests from flask import Flask, request from openai import OpenAI
```

Load environment variables

```
PAGE_ACCESS_TOKEN = os.environ.get("PAGE_ACCESS_TOKEN")
VERIFY_TOKEN = os.environ.get("VERIFY_TOKEN") OPENAI_KEY =
os.environ.get("OPENAI_KEY") client = OpenAI(api_key=OPENAI_KEY) app =
Flask(name) SELLER BRAIN = """ You are an AI sales assistant for a Facebook
Marketplace seller. Keep replies short, friendly, and direct (1–3 sentences). Seller
inventory categories:
```

1. Cars, trucks, motorcycles
2. ATVs, dirt bikes, jet skis, boats
3. Car parts (OEM + aftermarket)
4. Trailers, lawn mowers, power equipment
5. House appliances (fridge, washer, dryer, stove)
6. Furniture (beds, couches, dressers)
7. Tools (hand tools, power tools)

General rules:

1. ALWAYS answer “Yes, still available” if they ask about availability.
2. If they ask for price, repeat the listed price and remind it's firm unless seller says otherwise.
3. If they ask for lowest price → say “Price is firm but you can check it out in person.”
4. If they ask about condition → give a short general condition: “Good condition, works great.”
5. ALWAYS ask: “When would you like to come see it / pick it up?”
6. Never give exact address unless seller provides it; instead say: “Pickup in Parma, OH.”
7. Keep it polite, simple, and fast: no long paragraphs.

```
"""
def send_message(recipient_id, msg_text): url =
"https://graph.facebook.com/v18.0/me/messages" params = {"access_token": PAGE_ACCESS_TOKEN} payload = { "recipient": { "id": recipient_id}, "message": {"text": msg_text} } requests.post(url, params=params, json=payload)
@app.route('/', methods=['GET']) def verify(): if
request.args.get("hub.verify_token") == VERIFY_TOKEN: return
request.args.get("hub.challenge") return "Verification failed" @app.route('/'
```

```
methods=['POST']) def webhook(): data = request.get_json() if "entry" in data: for entry in data["entry"]: if "messaging" in entry: for messaging_event in entry["messaging"]: if "message" in messaging_event: sender_id = messaging_event["sender"]["id"] user_message = messaging_event["message"].get("text", "") # AI Reply response = client.chat.completions.create( model="gpt-4.1-mini", messages=[ {"role": "system", "content": ""}, {"role": "user", "content": user_message}, {"role": "assistant", "content": response}], temperature=0.5, max_tokens=100, top_p=1, frequency_penalty=0, presence_penalty=0) return Response(response=json.dumps(data), status=200, mimetype='application/json')
```