```
import nltk
from nltk.stem import WordNetLemmatizer
import numpy as np
import random
import string
lemmatizer = WordNetLemmatizer()
def preprocess(text):
    tokens = nltk.word_tokenize(text)
    tokens = [lemmatizer.lemmatize(word.lower()) for word in tokens]
    return tokens
def remove_punctuation(tokens):
    return [word for word in tokens if word.isalnum()
text = "Hello! How are you doing today?"
tokens = preprocess(text)
clean_tokens = remove_punctuation(tokens)
print(clean_tokens)
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.svm import LinearSVC
training_data = [
    {"class": "greeting", "patterns": ["hi", "hello", "hey", "how are you"]},
    {"class": "goodbye", "patterns": ["bye", "see you later", "goodbye"]},
    {"class": "thanks", "patterns": ["thanks", "thank you", "thanks a lot"]},
]
corpus = [pattern for intent in training_data for pattern in intent['patterns']]
labels = [intent['class'] for intent in training_data]
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(corpus).toarray()
clf = LinearSVC()
clf.fit(X, labels)
def classify_intent(text):
    tokens = preprocess(text)
```

```
    cleaned_tokens = remove_punctuation(tokens)
    tfidf = vectorizer.transform([" ".join(cleaned_tokens)]).toarray()
    predicted_class = clf.predict(tfidf)[0]
    return predicted_class
user_input = "Hi there!"
intent = classify_intent(user_input)
print("Intent:", intent)
from flask import Flask, request, jsonify

app = Flask(_name_)
@app.route('/chat', methods=['POST'])
def chat():
    user_message = request.json['message']
    intent = classify_intent(user_message)
    response = generate_response(intent)
    return jsonify({'message': response})
def generate_response(intent):
    responses = {
        "greeting": "Hello! How can I help you?",
        "goodbye": "Goodbye! Have a nice day.",
        "thanks": "You're welcome!",
        # Add more responses based on intents
    }
    return random.choice(responses[intent])

if _name_ == '_main_':
    app.run(debug=True)
```