Test Results

September 12, 2024

```python
[1]: import os
     from langchain_groq import ChatGroq
     from  dotenv import load_dotenv

     load_dotenv()
     groq_api_key = os.getenv("GROQ_API_KEY")
     os.environ["GROQ_API_KEY"] = groq_api_key
     model = ChatGroq(
         model="llama-3.1-70b-versatile"
     )
```

```python
[2]: import bs4
     from langchain_community.document_loaders import WebBaseLoader

     bs4_strainer = bs4.SoupStrainer(class_=("ok-wysiwyg-wrapper"))
     loader = WebBaseLoader(
         web_paths=("https://students.ok.ubc.ca/academic-success/advising-options/
       ↪academic-advising/frequently-asked-questions/#lab",),
         bs_kwargs={"parse_only": bs4_strainer},
     )

     docs = loader.load()
     len(docs[0].page_content)
```

    USER_AGENT environment variable not set, consider setting it to identify your
    requests.

```
[2]: 22533
```

```python
[4]: print(docs[0].page_content[:1000])
```

    Explore answers to questions related to advising services, degree planning,
    transfer credits, and records and registration.
    School of Engineering
    For questions about programs or courses offered by the School of Engineering,
    contact School of Engineering Academic Advising.
    Other Faculty resources
    For other Faculty resources, including forms, Honours program applications, and

department contacts, refer to the faculty resources at the bottom of this page

Records and registration

How do I access Workday?
Use Workday on a desktop device for the best user experience. Log in to Workday
at myworkday.ubc.ca with your Campus-wide Login (CWL).
If you don't have a Student CWL, you can create one after accepting your offer
and receiving a student number. You can set up a Student CWL through UBC IT. If
you do not have Multi-factor Authentication set up, follow the
instructions online, then log into Workday.
View our How-To tutorials here.
Back to top

Where do I find my registration date and ti

```python
from langchain_text_splitters import RecursiveCharacterTextSplitter

text_splitter = RecursiveCharacterTextSplitter(
    chunk_size = 1000,
    chunk_overlap = 200,
    add_start_index = True
)

all_splits = text_splitter.split_documents(docs)
len(all_splits)
```

[5]: 32

```python
len(all_splits[0].page_content)
```

[9]: 954

```python
all_splits[10].metadata
```

[10]: {'source': 'https://students.ok.ubc.ca/academic-success/advising-
options/academic-advising/frequently-asked-questions/#lab',
  'start_index': 7436}

```python
from langchain_openai import OpenAIEmbeddings
from langchain_pinecone import Pinecone, PineconeVectorStore
import pinecone

load_dotenv()
pinecone_api_key = os.getenv('PINECONE_API_KEY')
os.environ["PINECONE_API_KEY"] = pinecone_api_key

openai_api_key = os.getenv("OPENAI_API_KEY")
```

```python
os.environ["OPENAI_API_KEY"] = openai_api_key

embeddings = OpenAIEmbeddings(
    model="text-embedding-3-small"
)
index_name = "bolt-chatbot"
vectorstore = PineconeVectorStore.from_documents(
    documents=all_splits,
    embedding=embeddings,
    index_name=index_name
)
```

```python
[15]: retriever = vectorstore.as_retriever(search_type="similarity",␣
      ↪search_kwargs={"k": 3})
      retrieved_docs = retriever.invoke("What is a closed course?")

      len(retrieved_docs)
```

```
[15]: 3
```

```python
[16]: print(retrieved_docs[0].page_content)
```

Back to top

What is the difference between reserved and unreserved seats in a course?
Some courses might have "reserved" seats. These seats are reserved for specific
students, for example students in a specific program or year level. If you are a
member of the group that these seats are reserved for, you will be able to
register.
"Unreserved" seats are open for any students that do not meet the criteria for
the reserved seats in that course but do meet the pre-requisites.
Back to top

What is a "closed" course?
A closed course indicates that the course is full.
Back to top

```python
[21]: from langchain_core.output_parsers import StrOutputParser
      from langchain.prompts import PromptTemplate

      parser = StrOutputParser()

      template = """
      Answer the question based on the context below. You are an academic advisor for␣
       ↪university students. If you can't
      answer the question, reply I don't know.

      Context: {context}
```

```
Question: {question}
"""

prompt = PromptTemplate.from_template(template)
prompt.format(context="here is some context", question="here is a question")
```

[21]: " \nAnswer the question based on the context below. You are an academic advisor
for university students. If you can't \nanswer the question, reply I don't
know.\n\nContext: here is some context\n\nQuestion: here is a question\n"

[22]:
```python
from operator import itemgetter

chain = (
    {
        "context": itemgetter("question") | retriever,
        "question": itemgetter("question"),
    }
    | prompt
    | model
    | parser
)
```

[26]:
```python
question = input()
print(f"Question: {question}")
print(f"Answer: {chain.invoke({'question': question})}")
print()
```

Question: What grade do I need to achieve to get transfer credits?
Answer: You need to achieve a grade of four or better on approved AP exams to be
eligible for transfer credit. For IB courses, transfer credit depends on what
you completed and your minimum score, but the specific score is not mentioned in
the provided context.

[ ]: