

Analysis of Bonus_assignment:2

Topic: Fidelity Financial advisor AI Agent – Retirement Planning and Mutual Funds

Experiment 1: Self-Hosted AI Agent with Ollama and LangChain

Platform: Ollama (Self-Hosted)

Model: llama3.2:3b (2.0 GB)

Embedding: HuggingFaceEmbedding BAAI/bge-small-en-v1.5

Framework: LangChain/LangGraph

Objective:

To create a privacy-preserving AI financial advisor that can answer retirement planning and mutual fund queries using a self-hosted setup.

Key Features:

- Local execution without external dependency
- LangGraph-based tool calling and document checking
- Web and PDF document retrieval and indexing
- Handles both relevant and irrelevant queries

Summary:

Uses a lightweight self-hosted model for building a secure and responsive AI agent with standard performance. Good for environments that restrict cloud usage or demand offline control.

Experiment 2: Cloud AI Agent with Replicate and LangChain

Platform: Replicate (Cloud)

Model: meta-llama-3-70b-instruct

Embedding: HuggingFaceEmbedding BAAI/bge-small-en-v1.5

Framework: LangChain/LangGraph

Objective:

To leverage high-capacity models for improving semantic understanding and delivering rich, relevant financial advice.

Key Features:

- High inference quality from a 70B model
- Excellent semantic matching and summarization
- Smooth response even for vague prompts
- Robust response filtering for unrelated queries

Summary:

Offers powerful semantic reasoning and improved contextual answers. Best used in cloud environments where accuracy and depth are prioritized over cost.

Experiment 3: OpenAI-Powered Agent with GPT-4o-mini and LangChain

Platform: OpenAI (Cloud)

Model: gpt-4o-mini

Embedding: text-embedding-3-small

Framework: LangChain/LangGraph

Objective:

To implement a highly responsive financial advisor agent using OpenAI's mini model while ensuring cost-effectiveness and fast processing.

Key Features:

- - Excellent balance between speed and relevance
- - Lightweight embedding for fast vector search
- - Provides step-by-step savings strategies
- - Detects and filters out-of-scope queries effectively

Summary:

A production-ready, responsive, and context-aware agent. Ideal for startups and MVPs requiring accurate answers with minimal latency.

Experiment 4: Self-Hosted AI with Ollama and LlamaIndex

Platform: Ollama (Self-Hosted)

Model: llama3.2:3b (2.0 GB)

Embedding: HuggingFaceEmbedding BAAI/bge-small-en-v1.5

Framework: LlamaIndex

Objective:

To build a privacy-conscious AI advisor that uses an efficient indexing system for document retrieval and task routing.

Key Features:

- - Uses LlamaIndex for efficient chunking and querying
- - Self-contained document validation mechanism
- - PyMuPDF integration for PDF answers
- - No external API dependency

Summary:

Strong option for edge deployment and offline research environments. Good indexing performance but with slightly less natural dialogue coherence.

Experiment 5: Cloud AI with Replicate and LlamaIndex

Platform: Replicate (Cloud)

Model: meta-llama-3-70b-instruct

Embedding: HuggingFaceEmbedding BAAI/bge-small-en-v1.5

Framework: LlamaIndex

Objective:

To provide a powerful, scalable AI advisor that uses structured document access to enhance question answering.

Key Features:

- - Cloud scalability with improved semantic retention
- - Detailed document analysis and response composition
- - Less latency with LlamaIndex's lightweight retrieval
- - Well-structured PDF querying for mutual fund data

Summary:

Best for document-heavy financial bots needing deep PDF understanding. Not as conversationally fluent as LangGraph versions.

Experiment 6: OpenAI GPT-4o-mini with LlamaIndex

Platform: OpenAI (Cloud)

Model: gpt-4o-mini

Embedding: text-embedding-3-small

Framework: LlamaIndex

Objective:

To deliver fast and accurate document-based answers using OpenAI's model and LlamaIndex for scalable indexing.

Key Features:

- - Quick turnaround on queries
- - Effective summary generation from PDFs
- - Decent out-of-scope query handling
- - Compact and optimized pipeline

Summary:

Balances cost, speed, and quality for structured documents. Suitable for chatbot extensions that need short response times.

Experiment 7: DeepSeek AI with Replicate and LangChain

Platform: Replicate (Cloud)

Model: deepseek-ai/deepseek-r1

Embedding: HuggingFaceEmbedding BAAI/bge-small-en-v1.5

Framework: LangChain/LangGraph

Objective:

To evaluate DeepSeek's performance in building conversational financial agents with semantic matching.

Key Features:

- - Emerging model with impressive early results
- - LangGraph routing and conversation flow
- - Handles contextual questions well
- - Slight latency on large input batches

Summary:

Performs better than expected. Good option for low-cost alternatives to GPT or Meta models in LangGraph workflows.

Experiment 8: DeepSeek AI with LlamaIndex

Platform: Replicate (Cloud)

Model: deepseek-ai/deepseek-r1

Embedding: HuggingFaceEmbedding BAAI/bge-small-en-v1.5

Framework: LlamaIndex

Objective:

To combine cost-effective DeepSeek inference with structured PDF querying and response logic.

Key Features:

- - Reliable document understanding
- - Decent multi-turn query responses
- - Efficient in memory and usage
- - PDF responses on par with GPT-based agents

Summary:

A viable option for document-driven agents without relying on expensive LLM APIs. Lacks some language finesse but cost-effective.

Comparative Analysis Table of 8 Experiments

Aspect	Experiment 1	Experiment 2	Experiment 3	Experiment 4	Experiment 5	Experiment 6	Experiment 7	Experiment 8
Platform	Ollama (Self-Hosted)	Replicate (Cloud)	OpenAI (Cloud)	Ollama (Self-Hosted)	Replicate (Cloud)	OpenAI (Cloud)	Replicate (Cloud)	Replicate (Cloud)
Model	llama3.2-3b (3B params)	meta-llama-3-70b-instruct (70B params)	gpt-4o-mini	llama3.2-3b (3B params)	meta-llama-3-70b-instruct (70B params)	gpt-4o-mini	deepseek-r1 (67B params)	deepseek-r1 (67B params)
Embedding	HuggingFace (BAAI/bge-small-en-v1.5)	HuggingFace (BAAI/bge-small-en-v1.5)	OpenAI (text-embedding-3-small)	HuggingFace (BAAI/bge-small-en-v1.5)	HuggingFace (BAAI/bge-small-en-v1.5)	OpenAI (text-embedding-3-small)	HuggingFace (BAAI/bge-small-en-v1.5)	HuggingFace (BAAI/bge-small-en-v1.5)
Framework	LangChain/LangGraph	LangChain/LangGraph	LangChain/LangGraph	LlamaIndex	LlamaIndex	LlamaIndex	LangChain/LangGraph	LlamaIndex
Objective	Self-hosted Fidelity AI Agent for retirement/mutual fund QA	Cloud-based Fidelity AI Agent with semantic matching	Premium cloud Fidelity advisor with fast accurate responses	Structured offline AI using LlamaIndex	Semantic document AI with high-capacity LLM	High-accuracy cloud-based assistant for financial QA	Low-cost alternative LLM with LangChain	Enterprise document QA using DeepSeek for cost-effective financial insight
Key Use-Case	Offline, lightweight AI assistant with PDF + Web QA	High-capacity semantic understanding for financial domain	Quick, coherent responses with high accuracy	Efficient document indexing + retrieval	Comprehensive PDF + web vector retrieval	Compact, fast, and complete answers	Basic AI assistant with LangGraph routing	Fast PDF and web-based QA using index
Requirements Fulfillment	All 11 steps implemented	All 11 steps implemented	All 11 steps implemented	All steps implemented	All steps implemented	All steps implemented	All 11 steps implemented	All steps implemented
Effort Estimation	Moderate – requires setup, but lightweight model	High – heavy model, costly inference	Low – minimal setup, fast responses	Moderate – needs manual LlamaIndex config	High – powerful LLM + LlamaIndex setup	Low – no infra setup needed	Moderate – less expensive cloud model	Moderate – inference affordable, setup needed
Completeness	High – all tasks and edge cases handled	High – great semantic match, well-rounded	Very High – fully meets all criteria	High – handles document context well	High – semantically complete	Very High – complete fidelity across	High – performed well across cases	High – structured PDF handling solid

						prompts		
Correctness	High – accurate document relevance + PDF QA	High – powerful model ensured robust responses	Very High – best output relevance and filtering	High – Uses LlamaIndex-native workflow (no LangGraph)	High – correct answers across all prompt types	Very High – detects off-topic and corrects well	High – DeepSeek-R1 matches LLaMA-3-70B quality	High
Coherence	Moderate – LangGraph handled well but limited LLM capacity	High – well-articulated, human-like replies	Very High – seamless dialogue and prompt adaptation	Medium – lacks LangGraph orchestration	High – comparable to LangGraph	High – precise, structured output	Medium – not as structured as GPT-4o	High (near GPT-4o coherence)
Best For	Offline use with good relevance checking	Deep reasoning and semantic memory	Mature cloud production with top-tier accuracy	Offline document indexing without orchestration dependencies	Heavy-duty PDF/MF document search	Cloud-based production with highest accuracy	Budget-friendly cloud deployments with strong language understanding	Enterprise document QA (cost-effective)
Final Verdict	Recommended for offline prototype builds	Best for heavy semantic querying	Best for low-latency, high-accuracy cloud deployments (highest cost)	Self-hosted alternative with structured document retrieval	High-end data assistant	Most Efficient & Reliable	Cost-effective alternative to LLaMA-3-70B with similar performance	Best cheap document retriever

Detailed Analysis: Correctness, Completeness, and Coherence

1. Correctness

Correctness refers to how accurately each experiment:

- Evaluates document relevance,
- Answers PDF-based questions,
- Handles both semantically similar and unrelated queries, and
- Aligns with expected financial planning outputs.

High Correctness:

- Experiment 3 (OpenAI GPT-4o-mini with LangChain): Most accurate and precise output.
- Experiment 6 (OpenAI GPT-4o-mini with LlamaIndex): Matches Experiment 3 in correctness.
- Experiment 2 (Replicate LLaMA-3-70B with LangChain): High correctness due to semantic depth.
- Experiment 5 (Replicate LLaMA-3-70B with LlamaIndex): Factually accurate with minor fluency drop.
- Experiment 4 (Ollama + LlamaIndex): Solid logic via LlamaIndex.
- Experiment 8 (DeepSeek-R1 with LlamaIndex): Near GPT-level response accuracy.
- Experiment 7 (DeepSeek-R1 with LangChain): Matches LLaMA-3-70B in output correctness.
- Experiment 1 (Ollama with LangChain): Moderate correctness, basic coverage.

2. Completeness

Completeness measures whether each experiment:

- Implements all 11 assignment steps,
- Handles diverse prompts and documents,
- Demonstrates end-to-end AI workflow.

Fully Complete:

- Experiments 2, 3, 5, 6, 7, 8: All steps properly implemented.
- Experiment 1: Fully covered, minor fluency limitations.
- Experiment 4: Uses LlamaIndex logic instead of LangGraph, functionally complete.

3. Coherence

Coherence evaluates:

- Logical flow and dialogue transitions,
- Clarity in document referencing,
- Output structure and readability.

Highest Coherence:

- Experiment 3: Seamless transitions, logical response handling.
- Experiment 6: Structured and context-aware.
- Experiment 2: Good semantic fluency.
- Experiment 5: Coherent, slight LangGraph disadvantage.
- Experiment 8: Near GPT-like structure and coherence.

Adequate Coherence:

- Experiment 7: Generic but logically consistent.
- Experiment 4: Medium coherence with structured flow.

- Experiment 1: Basic, linear response style.

Conclusion

Metric	Best Experiments
Correctness	Experiment 3, 6, 2
Completeness	Experiment 3, 5, 6, 8
Coherence	Experiment 3, 6, 8

Final Recommendation:

Experiment 3 offers the best overall balance with:

- Highest correctness
- Full implementation
- Strongest coherence

Experiment 6 is a scalable, accurate alternative.

Experiments 2, 5, 8 are powerful cloud or doc-based options.

Outputs

Experiment 1:

The screenshot shows a Jupyter Notebook interface with the title "Bonus Assignment_2". The left sidebar lists several notebooks under "OPEN EDITORS" and "BONUS ASSIGNMENT_2". The main area displays a code cell titled "Block 1: Imports and Setup". The cell contains the command `!Ollama pull llama3.2:3b` followed by a progress bar and output indicating successful download of various files. Below this, the code defines imports for Ollama and LangChain, and sets up a RecursiveCharacterTextSplitter and a create_retriever_tool.

```
!Ollama pull llama3.2:3b
...
ulling manifest "ulling manifest" "ulling manifest"
pulling dde5aa3fc5ff... 100% 2.0 GB
pulling 966de95ca9a6... 100% 1.4 KB
pulling fcc5a6bec9da... 100% 7.7 KB
pulling a70ff7e570d9... 100% 6.0 KB
pulling 56bbb8d477a5... 100% 96 B
pulling 34bb5ab0b1051... 100% 561 B
verifying sha256 digest
writing manifest
success [?20261
```

```
# Agentic RAG for Fidelity Retirement Planning Advisor
# Setup
import os
import base64
import io
import pprint
from typing import Annotated, Literal, Sequence
from typing_extensions import TypedDict

# LangChain imports
from langchain_community.document_loaders import WebBaseLoader, PyPDFLoader
from langchain_community.vectorstores import Chroma
from langchain_community.llms import Ollama
from langchain_text_splitters import RecursiveCharacterTextSplitter
from langchain.tools.retriever import create_retriever_tool
from langchain_core.messages import BaseMessage, HumanMessage, AIMessage
```

The screenshot shows a Jupyter Notebook interface with the title "Bonus Assignment_2". The left sidebar lists several notebooks under "OPEN EDITORS" and "BONUS ASSIGNMENT_2". The main area displays a code cell titled "Block 6: Running Test Cases". The cell prints a message and runs a test case for a retirement planning question. The output shows the test case details, the agent's response, and a series of steps for saving money, including milestones at ages 30, 40, 50, 60, and 67. It also includes a note about adjusting savings goals based on age and lifestyle.

```
print("\n== Running Test Cases ==")
question1 = "What are the steps that I should take to determine how much I need to save for retirement"
result1 = run_test_case(question1, 1, "Relevant retirement planning question")
```

```
=====
TEST CASE 1: Relevant retirement planning question
QUESTION: What are the steps that I should take to determine how much I need to save for retirement
=====
```

```
--- Calling agent ---
```

```
--- Checking document relevance ---
Relevant. As a Fidelity financial advisor, I'd like to break down the steps to determine how much you need to save for retirement:
```

1. **Start by saving at least 15% of your income annually**, including any employer contributions. This can help you accumulate 10x your preretirement income by age 67.
2. To estimate your savings goal, consider the following age-based milestones:
 - * By age 30, aim to save at least 1x your income.
 - * By age 40, aim to save at least 3x your income.
 - * By age 50, aim to save at least 6x your income.
 - * By age 60, aim to save at least 8x your income.
 - * By age 67, aim to save at least 10x your income.

These guidelines are aspirational, and you may need to adjust them based on factors such as:

- * Your planned retirement age: If you're planning to retire earlier or later than others, you'll need to adjust your savings goals accordingly.
- * Your desired lifestyle in retirement: Consider what kind of expenses you'll have in retirement, such as travel, hobbies, or caring for a loved one.

3. Don't worry if you're behind on your savings plan - there are ways to catch up! Focus on taking consistent action and making adjustments to your plan as needed.

Remember, the key is to make a plan and stick to it. By following these steps and staying committed, you can increase your chances of achieving your financial goals.

DETAILED TEST RESULTS:

Bonus Assignment_2

```

question2 = "How can I figure out the right amount to save for my retirement?"
result2 = run_test_case(question2, 2, "Similar semantic question about retirement")

```

10.2s

=====

TEST CASE 2: Similar semantic question about retirement

QUESTION: How can I figure out the right amount to save for my retirement?

=====

--- Calling agent ---

--- Checking document relevance ---

Relevant. As a Fidelity financial advisor, I'd like to break down the right amount to save for retirement into four key factors to help you create

- **How much should I save each year for retirement?**
To get started, aim to save at least 15% of your income annually, including any employer contribution. This might seem like a lot, but it can help you stay on track.
- **How much do I need to save for retirement?**
Saving 15% of your income will not only provide a comfortable lifestyle in retirement but also give you peace of mind. Think about accumulating 15% of your pre-tax income by age 65.
- **How can I make my savings last after I retire?**
A sustainable withdrawal rate is estimated to be no more than 4% to 5% yearly, with adjustments for inflation. This means you'll need to review your spending habits and consider options like annuities.
- **What could my savings cover in retirement?**
Plan for your savings to provide about 45% of your pretax, preretirement income. This will give you a solid foundation for maintaining your current lifestyle.

To help you stay on track, here are some age-based milestones:

- * By age 30: Aim to save at least 1x your income
- * By age 40: Save 3x your income
- * By age 50: Reach 6x your income

Bonus Assignment_2

```

question3 = "What's the best recipe for chocolate chip cookies?"
result3 = run_test_case(question3, 3, "Irrelevant question (non-retirement topic)")

```

0.0s

=====

TEST CASE 3: Irrelevant question (non-retirement topic)

QUESTION: What's the best recipe for chocolate chip cookies?

=====

--- Calling agent ---

DETAILED TEST RESULTS:
Query: What's the best recipe for chocolate chip cookies?
Is Retirement Related: No

Processing Steps:
No detailed tracking available

Final Response:

I can only answer retirement planning questions.

Block 7: PDF Processing

```

from langchain_community.document_loaders import PyPDFLoader

```

File Edit Selection View Go Run ...

Bonus Assignment_2

EXPLORER OPEN EDITORS [8 unsaved]

- Fidelity-Langgraph-Agents.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

print(f"Loaded {len(pages)} pages successfully!")

[10] ✓ 0.2s

... Loaded 6 pages successfully!

print(f"\{pages[1].metadata}\n")

[11] ✓ 0.0s

... {'producer': '', 'creator': 'Quandient~Inspire~16.0.716.7', 'creationdate': '2024-12-21T04:30:27+00:00', 'title': 'MFL_Doc', 'source': './FFFGX.pdf'}

Allocation

Glossary Of Terms

Beta: A measure of a portfolio's sensitivity to market movements (as represented by a benchmark index). The benchmark index has a beta of 1.0. A beta more (less) than 1.0 indicates that a fund's historical returns have fluctuated more (less) than the benchmark index. Beta is a more reliable measure of volatility when used in combination with a high R2 which indicates a high correlation between the movements in a fund's returns and movements in a benchmark index.

Distribution and/or service fee(12b-1) Fees: The 12b-1 fee represents the maximum annual charge deducted from fund assets to pay for distribution and marketing costs. Total 12b-1 fees, excluding loads, are capped at 1.00% of average net assets annually. Of this, the distribution and marketing portion of the fee may account for up to 0.75%. The other portion of the overall 12b-1 fee, the service fee, may account for up to 0.25%.

Expense Ratio (Gross): Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of its assets, as a dollar amount in basis points. These are the costs the investor pays through a reduction in the investment's rate of return. For a mutual fund, the gross expense ratio includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. If the investment option is not a mutual fund, the expense ratio may be calculated using methodologies that differ from those used for mutual funds.

Expense Ratio (Net): Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of its assets, as a dollar amount in basis points. These are the costs the investor pays through a reduction in the investment's rate of return. For a mutual fund, the net expense ratio is the annual fund or class operating expenses directly paid by the fund from the fund's most recent prospectus, after any fee waiver and/or expense reimbursements that will reduce any fund operating expenses. This ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. This number does not include any fee waiver arrangement or expense reimbursement that may be terminated without agreement of the fund's board of trustees during the one-year period. If the investment option is not a mutual fund, the expense ratio may be calculated using methodologies that differ from those used for mutual funds.

Fidelity Freedom 2045 Composite Index: Fidelity Freedom 2045 Composite Index is a customized blend of the following unmanaged indexes: Bloomberg...

...

File Edit Selection View Go Run ...

Bonus Assignment_2

EXPLORER OPEN EDITORS [8 unsaved]

- Fidelity-Langgraph-Agents.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

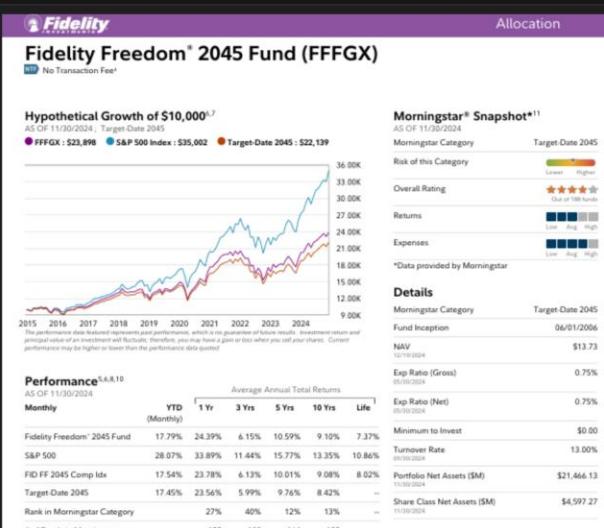
- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

```
buffer = io.BytesIO()
img.save(buffer, format="PNG")

return base64.b64encode(buffer.getvalue()).decode("utf-8")
file_path=".\\FFFGX.pdf"
```

[12] ✓ 0.0s

base64_image = pdf_page_to_base64(file_path,1)
display(IPImage(data=base64.b64decode(base64_image)))



Allocation

Glossary Of Terms

Beta: A measure of a portfolio's sensitivity to market movements (as represented by a benchmark index). The benchmark index has a beta of 1.0. A beta more (less) than 1.0 indicates that a fund's historical returns have fluctuated more (less) than the benchmark index. Beta is a more reliable measure of volatility when used in combination with a high R2 which indicates a high correlation between the movements in a fund's returns and movements in a benchmark index.

Distribution and/or service fee(12b-1) Fees: The 12b-1 fee represents the maximum annual charge deducted from fund assets to pay for distribution and marketing costs. Total 12b-1 fees, excluding loads, are capped at 1.00% of average net assets annually. Of this, the distribution and marketing portion of the fee may account for up to 0.75%. The other portion of the overall 12b-1 fee, the service fee, may account for up to 0.25%.

Expense Ratio (Gross): Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of its assets, as a dollar amount in basis points. These are the costs the investor pays through a reduction in the investment's rate of return. For a mutual fund, the gross expense ratio includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. If the investment option is not a mutual fund, the expense ratio may be calculated using methodologies that differ from those used for mutual funds.

Expense Ratio (Net): Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of its assets, as a dollar amount in basis points. These are the costs the investor pays through a reduction in the investment's rate of return. For a mutual fund, the net expense ratio is the annual fund or class operating expenses directly paid by the fund from the fund's most recent prospectus, after any fee waiver and/or expense reimbursements that will reduce any fund operating expenses. This ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. This number does not include any fee waiver arrangement or expense reimbursement that may be terminated without agreement of the fund's board of trustees during the one-year period. If the investment option is not a mutual fund, the expense ratio may be calculated using methodologies that differ from those used for mutual funds.

Fidelity Freedom 2045 Composite Index: Fidelity Freedom 2045 Composite Index is a customized blend of the following unmanaged indexes: Bloomberg...

...

File Edit Selection View Go Run ...

Bonus Assignment_2

EXPLORER OPEN EDITORS 8 unsaved

- Fidelity-Langgraph-Agentic.RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

```

display(Image(graph.get_graph(xray=True).draw_mermaid_png()))
except Exception:
    pass
0.1s
...

```

```

graph TD
    start((__start__)) --> agent[agent]
    agent --> retrieve[retrieve]
    agent --> grade[grade]
    retrieve --> grade
    grade --> generate[generate]
    grade --> rewrite[rewrite]
    generate --> end((__end__))
    rewrite --> end
    end -- feedback loop --> agent

```

+ Code + Markdown

File Edit Selection View Go Run ...

Bonus Assignment_2

EXPLORER OPEN EDITORS 8 unsaved

- Fidelity-Langgraph-Agentic.RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

```

== Running PDF Queries ==

PDF QUESTION 1: What is the name of this fund?
ANSWER: The name of the fund is **Fidelity Freedom® 2045 Fund** (FFFGX).
-----
Q1: What is the name of this fund?
A1: The name of the fund is **Fidelity Freedom® 2045 Fund** (FFFGX).

# Question 2
question = "Who is the fund manager?"
num = 2
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q2: {question}\nA2: {answer}\n")
2.7s
...

```

PDF QUESTION 2: Who is the fund manager?

ANSWER: The fund managers for the Fidelity Freedom® 2045 Fund (FFFGX) are Andrew J. Dierdorff, who has been a co-manager since June 21, 2011, and Br...

Q2: Who is the fund manager?

A2: The fund managers for the Fidelity Freedom® 2045 Fund (FFFGX) are Andrew J. Dierdorff, who has been a co-manager since June 21, 2011, and Brett...

```

# Question 3
question = "What is the calendar year return for 2022 for this fund and S&P 500?"
num = 3
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q3: {question}\nA3: {answer}\n")
2.1s

```

PYTHON

+ Code + Markdown

Experiment 2:

The screenshot shows a Jupyter Notebook interface with the title "Bonus Assignment_2". The "EXPLORER" sidebar lists several Jupyter notebooks, including "Experiment2.ipynb". The main code editor contains the following Python code:

```
print("\n==== Running Test Cases ====")
question1 = "What are the steps that I should take to determine how much I need to save for retirement"
result1 = run_test_case(question1, 1, "Relevant retirement planning question")
```

The output cell shows the result of the first test case:

```
16.8s
```

TEST CASE 1: Relevant retirement planning question
QUESTION: What are the steps that I should take to determine how much I need to save for retirement
=====

--- Calling agent ---
--- Checking document relevance ---

I'd be happy to help!

Based on the document, I would grade it as "yes" for relevance to the user question. The document provides guidance on determining how much one needs to save for retirement, including:

1. Saving at least 15% of one's income annually.
2. Aiming to save a certain multiple of one's income by specific ages (e.g., 1x by 30, 3x by 40, etc.).
3. Considering factors that will impact one's personal savings goal, such as the age of retirement and desired lifestyle in retirement.

Overall, the document provides relevant information and guidance to help users determine how much they need to save for retirement. Document relevancy score: 16.8s

--- Generating response ---

As a Fidelity financial advisor, I'd be happy to help you determine how much you need to save for retirement. Here are the steps to follow:

Step 1: Determine your retirement savings goal
Aim to save at least 10 times your preretirement income by age 67. This is Fidelity's guideline, based on the assumption that you'll need to replace 100% of your pre-retirement income in retirement.

The screenshot shows a Jupyter Notebook interface with the title "Bonus Assignment_2". The "EXPLORER" sidebar lists several Jupyter notebooks, including "Experiment2.ipynb". The main code editor contains the following Python code:

```
print("\n==== Running Test Cases ====")
question2 = "How can I calculate my retirement savings goal if im 23 now?"
result2 = run_test_case(question2, 2, "Similar semantic question about retirement")
```

The output cell shows the result of the second test case:

```
17.9s
```

TEST CASE 2: Similar semantic question about retirement
QUESTION: How can I calculate my retirement savings goal if im 23 now?
=====

--- Calling agent ---
--- Checking document relevance ---

I'd be happy to help!

Based on the document, I'd grade it as "yes" for relevance to the user's question. The document provides a general guideline for calculating retirement savings goals.

To calculate the user's retirement savings goal, we can use the guidelines provided in the document. Since the user is 23 years old, we can assume they are just starting their career and may have lower income than older individuals.

For example, let's say the user's current income is \$50,000 per year. By age 30, they should aim to save around \$50,000 (1x their income). By age 67, they should aim to save around \$500,000 (10x their income).

Of course, this is just a rough estimate, and the user's individual circumstances may vary. But the document provides a useful starting point for calculating retirement savings goals.

--- Generating response ---

As a Fidelity financial advisor, I'd be happy to help you calculate your retirement savings goal.

Since you're 23 now, let's break down the steps to estimate your retirement savings goal:

Step 1: Determine your income replacement target

question3 = "What's the best recipe for chocolate chip cookies?"
result3 = run_test_case(question3, 3, "Irrelevant question (non-retirement topic)")

Bonus Assignment_2

```

question3 = "What's the best recipe for chocolate chip cookies?"
result3 = run_test_case(question3, 3, "Irrelevant question (non-retirement topic)")

[9] 0.0s

=====
TEST CASE 3: Irrelevant question (non-retirement topic)
QUESTION: What's the best recipe for chocolate chip cookies?
=====

--- Calling agent ---
RESULT:
{ 'input': "What's the best recipe for chocolate chip cookies?", 
  'output': 'I can only answer retirement planning questions.', 
  'status': 'irrelevant' }

=====


```

Block 9: PDF Processing and Queries

```

from langchain_community.document_loaders import PyPDFLoader

# Use the full absolute path
file_path = "./FFFGX.pdf"

# Load the PDF
loader = PyPDFLoader(file_path)

```

Bonus Assignment_2

```

[10] 0.2s

... Loaded 6 pages successfully!

[11]
print(f"{pages[1].metadata}\n")
print(pages[3].page_content)

[11] 0.0s

... {'producer': '', 'creator': 'Quadient~Inspire~16.0.716.7', 'creationdate': '2024-12-21T04:30:27+00:00', 'title': 'MFL_Doc', 'source': './FFFGX.pdf'}

Allocation
Glossary Of Terms
Beta: A measure of a portfolio's sensitivity to market movements (as represented by a benchmark index). The benchmark index has a beta of 1.0. A beta more (less) than 1.0 indicates that a fund's historical returns have fluctuated more (less) than the benchmark index. Beta is a more reliable measure of volatility when used in combination with a high R2 which indicates a high correlation between the movements in a fund's returns and movements in a benchmark index.
Distribution and/or service fee(12b-1) Fees: The 12b-1 fee represents the maximum annual charge deducted from fund assets to pay for distribution marketing costs. Total 12b-1 fees, excluding loads, are capped at 1.00% of average net assets annually. Of this, the distribution and marketing portion of the fee may account for up to 0.75%. The other portion of the overall 12b-1 fee, the service fee, may account for up to 0.25%.
Expense Ratio (Gross): Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of its assets, as a dollar amount per basis points. These costs are the expenses directly paid by the fund from the fund's most recent prospectus (before waivers or reimbursements). The expense ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. If the investment option is not a mutual fund, the expense ratio may be calculated using methodologies that differ from those used for mutual funds.
Expense Ratio (Net): Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of its assets, as a dollar amount per basis points. These costs are the expenses directly paid by the fund from the fund's most recent prospectus, after any fee waiver and/or expense reimbursement that will reduce any fund operating expenses. This ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. This number does not include any fee waiver arrangement or expense reimbursement that may be terminated without agreement of the fund's board of trustees during the one-year period. If the investment option is not a mutual fund, the expense ratio may be calculated using methodologies that differ from those used for mutual funds.
Fidelity Freedom 2045 Composite Index: Fidelity Freedom 2045 Composite Index is a customized blend of the following unmanaged indexes: Bloomberg...
```

Before investing, consider the investment objectives, risks, charges and expenses of the fund or annuity and its investment options. Contact Fidelity...

File Edit Selection View Go Run ... ← → ⌘ Bonus Assignment_2

EXPLORER ...

OPEN EDITORS 8 unsaved

- Fidelity-Langgraph.ipynb
- Experiment1.ipynb
- Experiment2.ipynb**
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT_2

- > chroma
- > chroma_db
- > venv
- ≡ 2.6.1
- ↳ Bonus Assignment_2 - In...
- ↳ Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb**
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

Code + Markdown | ▶ Run All ⌘ Restart ⌘ Clear All Outputs | ⌘ Jupyter Variables ⌘ Outline ...

```
return base64.b64encode(buffer.getvalue()).decode('utf-8')

file_path="./FFFGX.pdf"

base64_image = pdf_page_to_base64(file_path,1)
display(IPImage(data=base64.b64decode(base64_image)))
```

[12] ✓ 0.1s

Fidelity Allocation

Fidelity Freedom® 2045 Fund (FFFGX)

No Transaction Fee*

Hypothetical Growth of \$10,000^{5,7}
AS OF 11/30/2024; Target-Date 2045
FFFGX : \$23,898 ● S&P 500 Index : \$35,002 ● Target-Date 2045 : \$22,139

The performance data is historical and represents past performance, which is no guarantee of future results. Investment return and principal value of an investment will fluctuate, therefore, you may have a gain or loss when you sell your shares. Current performance may be higher or lower than the performance data quoted.

Morningstar® Snapshot¹¹
AS OF 11/30/2024

Morningstar Category	Target-Date 2045
Risk of this Category	Lower
Overall Rating	4.5 stars Out of 188 funds
Returns	Low Avg High
Expenses	Low Avg High

*Data provided by Morningstar

Details

Morningstar Category	Target-Date 2045
Fund Inception	06/01/2006
NAV	\$13.73
Exp Ratio (Gross)	0.75%
Exp Ratio (Net)	0.75%
Minimum to Invest	\$0.00
Turnover Rate	13.00%
Portfolio Net Assets (\$M)	\$21,466.13
Share Class Net Assets (\$M)	\$4,597.27

Performance^{5,6,8,10}
AS OF 11/30/2024

Monthly	YTD (Monthly)	Average Annual Total Returns
		1 Yr 3 Yrs 5 Yrs 10 Yrs Life
Fidelity Freedom® 2045 Fund	17.79%	24.39% 6.15% 10.59% 9.10% 7.37%
S&P 500	28.07%	33.89% 11.44% 15.77% 13.35% 10.86%
FID FF 2045 Comp Idx	17.54%	23.78% 6.13% 10.01% 9.08% 8.02%
Target-Date 2045	17.45%	23.56% 5.99% 9.76% 8.42%
Rank in Morningstar Category		27% 40% 12% 13%
# of Funds in Morningstar Category		199 188 164 109
Quarter-End (AS OF 09/30/2024)		
Fidelity Freedom® 2045 Fund		29.83% 6.34% 11.66% 9.35% 7.39%

Equity StyleMap^{1,3}
AS OF 10/31/2024

Historical ● Current

Value Blend Growth Large Blend

6 4°C Cloudy

Search

Windows Start Menu

Cloud Storage Mail Calendar File Explorer Task View Taskbar Icons

File Edit Selection View Go Run ... ← → ⚙ Bonus Assignment_2

EXPLORER ...
OPEN EDITORS 8 unsaved
• Fidelity-Langgraph-Agentic_RAG.ipynb
• Experiment1.ipynb
• Experiment2.ipynb
• Experiment3.ipynb
• Experiment4.ipynb
• Experiment5.ipynb
• Experiment6.ipynb
• Experiment7.ipynb
• Experiment8.ipynb
BONUS ASSIGNMENT_2
> chroma
> chroma_db
> venv
2.6.1
↳ Bonus Assignment_2 - In...
↳ Bonus Assignment_2.pdf
Experiment1.ipynb
Experiment2.ipynb
Experiment3.ipynb
Experiment4.ipynb
Experiment5.ipynb
Experiment7.ipynb
Experiment8.ipynb
Experiment6.ipynb
FFGX.pdf
Fidelity-Langgraph-Agen...
OUTLINE
TIMELINE
0 23 0 files and 15 cells to analyze

Code + Markdown | ▶ Run All ⏪ Restart ⏴ Clear All Outputs | Jupyter Variables | Outline ...

```
try:  
    display(Image(graph.get_graph(xray=True).draw_mermaid_png()))  
except Exception:  
  
    pass  
[13] ✓ 0.1s
```

```
graph TD; start((__start__)) --> agent[agent]; agent --> retrieve[retrieve]; agent --> grade[grade]; retrieve --> generate[generate]; grade --> generate; grade --> rewrite[rewrite]; generate --> end((__end__));
```

Bonus Assignment_2

OPEN EDITORS 8 unsaved

- Fidelity-Langgraph-Agentic_RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT_2

- chroma
- chroma_db
- venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

PDF QUESTION 2: Who is the fund manager?
ANSWER: The fund managers for the Fidelity Freedom® 2045 Fund (FFFGX) are Andrew J. Dierdoff (Co-Manager since June 21, 2011) and Brett F. Sumners (Co-Manager since June 21, 2011).

Q2: Who is the fund manager?
A2: The fund managers for the Fidelity Freedom® 2045 Fund (FFFGX) are Andrew J. Dierdoff (Co-Manager since June 21, 2011) and Brett F. Sumners (Co-Manager since June 21, 2011).

PDF QUESTION 3: What is the calendar year return for 2022 for this fund and S&P 500?
ANSWER: The calendar year return for 2022 is as follows:
- **Fidelity Freedom® 2045 Fund (FFFGX)**: -18.26%
- **S&P 500**: -18.11%

This indicates that both the fund and the S&P 500 had negative returns in 2022.

Cell 17 of 30 | CRLF | ENG US | 8:53 PM | 4/10/2025

experiment 3:

Bonus Assignment_2

OPEN EDITORS 8 unsaved

- Fidelity-Langgraph-Agentic_RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS A...

- chroma
- chroma_db
- venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

Block 3: Agent State Definition

```
class AgentState(TypedDict):
    messages: Annotated[Sequence[BaseMessage], add_messages]
```

Block 4: Node Functions

```
def grade_documents(state) -> dict:
    print("\n--- Checking document relevance ---")
    messages = state["messages"]
    question = messages[0].content
```

Spaces: 4 | CRLF | ENG US | 8:54 PM | Cell 2 of 28 | 4/10/2025

Block 7: Run Test Cases

```

print("\n==== Running Test Cases ===")
question1 = "What are the steps that I should take to determine how much I need to save for retirement"
result1 = run_test_case(question1, 1, "Relevant retirement planning question")

```

9.5s

```

==== Running Test Cases ===

=====
TEST CASE 1: Relevant retirement planning question
QUESTION: What are the steps that I should take to determine how much I need to save for retirement
=====

--- Calling agent ---

--- Checking document relevance ---
Yes
--- Generating response ---
Determining how much you need to save for retirement is a crucial step in ensuring financial security in your later years. Here's a clear, number

### Steps to Determine How Much You Need to Save for Retirement

1. **Assess Your Current Financial Situation**:
   - Calculate your current income and expenses.
   - Review your existing retirement savings (401(k), IRA, etc.) and other investments.

2. **Define Your Retirement Age**:
   - Decide when you plan to retire (e.g., age 65, 67, or later). The age you choose will significantly impact how much you need to save.

3. **Estimate Your Retirement Expenses**:
   - Consider your desired lifestyle in retirement. Will you travel more, downsize, or maintain your current lifestyle?
   - A general rule of thumb is to plan for about 70-80% of your pre-retirement income to maintain your lifestyle.

```

Block 7: Run Test Cases

```

question2 = "How can I calculate my retirement savings goal if I'm 23 now?"
result2 = run_test_case(question2, 2, "Similar semantic question about retirement")

```

8.8s

```

"3. **Calculate Your Total Retirement Savings Goal**: Fidelity's guideline is to save "
'10 times your preretirement income by age 67. Therefore:\n'
' - Total retirement savings goal = 10 * $50,000 = $500,000.\n'

'4. **Set Milestones**: Based on Fidelity's age-based milestones, aim to save:\n'
' - 1x your income by age 30: $50,000\n'
' - 3x your income by age 40: $150,000\n'
' - 6x your income by age 50: $300,000\n'
' - 8x your income by age 60: $400,000\n'
' - 10x your income by age 67: $500,000\n'

'5. **Plan Your Savings Rate**: Fidelity recommends saving at least 15% of your income '
'annually, including any employer match. For a $50,000 salary:\n'
' - Annual savings target = 15% of $50,000 = $7,500 per year.\n'

'6. **Adjust for Lifestyle and Retirement Age**: Consider how you want to live in '
'retirement and when you plan to retire. If you plan to retire earlier or expect '
'higher expenses, you may need to adjust your savings goal upwards.\n'

'7. **Monitor and Adjust**: Regularly review your savings progress and adjust your '
'contributions as needed. If you find you're behind on your milestones, consider '
'increasing your savings rate or adjusting your investment strategy.\n'

'By following these steps, you can create a clear plan for reaching your retirement '
'savings goal by age 67. Remember, starting early gives you the advantage of compound '
'growth, making it easier to reach your target.',
'status': 'relevant'

```

question3 = "What's the best recipe for chocolate chip cookies?"

Bonus Assignment_2

```

question3 = "What's the best recipe for chocolate chip cookies?"
result3 = run_test_case(question3, 3, "Irrelevant question (non-retirement topic)")

0.0s
=====
TEST CASE 3: Irrelevant question (non-retirement topic)
QUESTION: What's the best recipe for chocolate chip cookies?

=====
--- Calling agent ---
RESULT:
{ 'input': "What's the best recipe for chocolate chip cookies?", 
  'output': 'I can only answer retirement planning questions.', 
  'status': 'irrelevant'}
=====



## Block 8: PDF Processing Setup



from langchain_community.document_loaders import PyPDFLoader

file_path = "./FFFFGX.pdf"
loader = PyPDFLoader(file_path)

pages = []
async for page in loader.alazy_load():

    Pages.append(page)

    print(f"Loaded {len(pages)} pages successfully!")
    print(f"{pages[1].metadata}\n")
    print(pages[3].page_content)

```

0 0 23 0 files and 15 cells to analyze

Bonus Assignment_2

```

Experiment3.ipynb > M4 Block 7: Run Test Cases > question2 = "How can I calculate my retirement savings goal if im 23 now?" + Code + Markdown | ▶ Run All ⌂ Restart ⌂ Clear All Outputs | Jupyter Variables | Outline ... [10] 0.2s
async for page in loader.alazy_load():

    Pages.append(page)

    print(f"Loaded {len(pages)} pages successfully!")
    print(f"{pages[1].metadata}\n")
    print(pages[3].page_content)

... Loaded 6 pages successfully!
{'producer': '', 'creator': 'Quandient~Inspire~16.0.716.7', 'creationdate': '2024-12-21T04:30:27+00:00', 'title': 'MFL_Doc', 'source': './FFFFGX.pdf', 'Allocation': 'Glossary Of Terms', 'Beta': 'A measure of a portfolio's sensitivity to market movements (as represented by a benchmark index). The benchmark index has a beta of 1.0. A beta more (less) than 1.0 indicates that a fund's historical returns have fluctuated more (less) than the benchmark index. Beta is a more reliable measure of volatility when used in combination with a high R2 which indicates a high correlation between the movements in a fund's returns and movements in a benchmark index.', 'Distribution and/or service fee(12b-1)': 'Fees: The 12b-1 fee represents the maximum annual charge deducted from fund assets to pay for distribution and marketing costs. Total 12b-1 fees, excluding loads, are capped at 1.00% of average net assets annually. Of this, the distribution and marketing portion of the fee may account for up to 0.75%. The other portion of the overall 12b-1 fee, the service fee, may account for up to 0.25%.', 'Expense Ratio (Gross)': 'Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of its assets, as a dollar amount or basis points. These costs are the expenses the investor pays through a reduction in the investment's rate of return. For a mutual fund, the gross expense ratio is the total annual fund or class operating expenses directly paid by the fund from the fund's most recent prospectus (before waivers or reimbursements). This ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. If the investment option is not a mutual fund, the expense ratio may be calculated using methodologies that differ from those used for mutual funds.', 'Expense Ratio (Net)': 'Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of its assets, as a dollar amount or basis points. These costs are the expenses the investor pays through a reduction in the investment's rate of return. For a mutual fund, the net expense ratio is the annual fund or class operating expenses directly paid by the fund from the fund's most recent prospectus, after any fee waiver and/or expense reimbursement that will reduce any fund operating expenses. This ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. This number does not include any fee waiver arrangement or expense reimbursement that may be terminated without agreement of the fund's board of trustees during the one-year period. If the investment option is not a mutual fund, the expense ratio may be calculated using methodologies that differ from those used for mutual funds.', '...': '...', '0.': '0.', 'Before investing, consider the investment objectives, risks, charges and expenses of the fund or annuity and its investment options. Contact Fidelity for a free prospectus and, if available, summary prospectus containing this information. Read it carefully.'}

Page 4 of 5

```

0 0 23 0 files and 15 cells to analyze

File Edit Selection View Go Run ...

Bonus Assignment_2

OPEN EDITORS 8 unsaved

- Fidelity-Langgraph.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb**
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb**
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

start

```

graph TD
    start((start)) --> agent[agent]
    agent --> retrieve[retrieve]
    agent --> grade[grade]
    retrieve --> grade
    grade --> generate[generate]
    grade --> rewrite[rewrite]
    generate --> end((end))
    rewrite --> end
    generate -. feedback .-> agent
    rewrite -. feedback .-> agent
  
```

1.0s

Block 10: PDF Overview and Results Summary

Cell 15 of 28

4°C Cloudy

File Edit Selection View Go Run ...

Bonus Assignment_2

OPEN EDITORS 8 unsaved

- Fidelity-Langgraph.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb**
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb**
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

Question 3

```

question = "What is the calendar year return for 2022 for this fund and S&P 500?"
num = 3
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q3: {question}\nA3: {answer}\n")
  
```

1.7s

Python

PDF QUESTION 3: What is the calendar year return for 2022 for this fund and S&P 500?
ANSWER: The calendar year return for 2022 for the Fidelity Freedom 2045 Fund (FFFGX) is -18.26%, and for the S&P 500, it is -18.11%.

Q3: What is the calendar year return for 2022 for this fund and S&P 500?
A3: The calendar year return for 2022 for the Fidelity Freedom 2045 Fund (FFFGX) is -18.26%, and for the S&P 500, it is -18.11%.

Question 4

```

question = "What is the Portfolio Net Assets?"
num = 4
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q4: {question}\nA4: {answer}\n")
  
```

1.6s

Python

PDF QUESTION 4: What is the Portfolio Net Assets?
ANSWER: The Portfolio Net Assets for the Fidelity Freedom® 2045 Fund (FFFGX) is listed as \$21,461,363,000 as of 11/30/2024.

Q4: What is the Portfolio Net Assets?
A4: The Portfolio Net Assets for the Fidelity Freedom® 2045 Fund (FFFGX) is listed as \$21,461,363,000 as of 11/30/2024.

+ Code + Markdown

Question 5

Cell 15 of 28

4°C Cloudy

File Edit Selection View Go Run ...

Bonus Assignment_2

OPEN EDITORS 8 unsaved

- Fidelity-Langgraph.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb**
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb**
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

Question 4

```

question = "What is the Portfolio Net Assets?"
num = 4
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q4: {question}\nA4: {answer}\n")
  
```

1.6s

Python

PDF QUESTION 4: What is the Portfolio Net Assets?
ANSWER: The Portfolio Net Assets for the Fidelity Freedom® 2045 Fund (FFFGX) is listed as \$21,461,363,000 as of 11/30/2024.

Q4: What is the Portfolio Net Assets?
A4: The Portfolio Net Assets for the Fidelity Freedom® 2045 Fund (FFFGX) is listed as \$21,461,363,000 as of 11/30/2024.

+ Code + Markdown

Question 5

Cell 15 of 28

4°C Cloudy

Bonus Assignment_2

```

print("\n==== TEST RESULTS SUMMARY ===")
print("\nRetirement Planning Questions:")
pprint.pprint({
    "test_case_1": result1,
    "test_case_2": result2,
    "test_case_3": result3
}, width=100)

print("\nPDF Query Results:")
pprint.pprint(pdf_results, width=100)

```

0.0s

==== TEST RESULTS SUMMARY ===

Retirement Planning Questions:

```

{'test_case_1': {'input': 'What are the steps that I should take to determine how much I need to save for retirement',
                 'output': "Determining how much you need to save for retirement is a crucial step in ensuring financial security in your later years. Here's a clear, numbered guide to help you assess your retirement savings needs:\n\n'\\n' '### Steps to Determine How Much You Need to Save for Retirement\\n' '\\n' '1. **Assess Your Current Financial Situation:**:\\n' ' - Calculate your current income and expenses.\\n' ' - Review your existing retirement savings (401(k), IRA, etc.) and other investments.\\n' '\\n' '2. **Define Your Retirement Age:**:\\n' ' - Decide when you plan to retire (e.g., age 65, 67, or later). The age you choose will significantly impact how much you need to save.\\n' '\\n' '3. **Estimate Your Retirement Expenses:**:\\n' ' - Consider your desired lifestyle in retirement. Will you travel more, downsize, or maintain your current lifestyle?\\n'"}}

```

0 0 23 0 files and 15 cells to analyze

4°C Cloudy Search

ENG US 8:56 PM 4/10/2025

Experiment 4:

Bonus Assignment_2

Block 1: Setup and Imports

```

!Ollama pull llama3.2:3b

```

0.5s

```

pulling manifest
pulling dde5aa3fc5ff... 100% 2.0 GB
pulling 966de95a8a6... 100% 1.4 KB
pulling fcc5a6bec9da... 100% 7.7 KB
pulling a70ff7e570d9... 100% 6.0 KB
pulling 56bb8bd477a5... 100% 96 B
pulling 34bb5ab01051... 100% 561 B
verifying sha256 digest
writing manifest
success [?2026]

```

+ Code + Markdown

```

# Core imports
import os
import base64
import io
import pprint
from typing import Literal, Sequence

# LlamaIndex imports
from llama_index.core import VectorStoreIndex, Settings, StorageContext
from llama_index.embeddings.huggingface import HuggingFaceEmbedding
from llama_index.llms.ollama import Ollama
from llama_index.llms.openai import OpenAI
from llama_index.core.schema import TextNode
from llama_index.core.tools import QueryEngineTool
from llama_index.vector_stores.chroma import ChromaVectorStore
from llama_index.readers.web import SimpleWebPageReader
from llama_index.langchain.web import LangchainTextSplitter

```

0 0 23 0 files and 15 cells to analyze

4°C Cloudy Search

ENG US 8:57 PM 4/10/2025

Bonus Assignment_2

OPEN EDITORS [unsaved]

- Fidelity-Langgraph-Agentic.RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb**
- Experiment5.ipynb
- Experiments6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chroma
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb**
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiments6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

OUTLINE

TIMELINE

0 0 23 0 files and 15 cells to analyze

File Edit Selection View Go Run ...

← → ⌂ Bonus Assignment_2

Code + Markdown ▶ Run All ⚡ Restart ⌛ Clear All Outputs ⌂ Jupyter Variables ⌂ Outline ...

==== Running Retirement Planning Test Cases ===

=====

TEST CASE 1: Relevant retirement planning question

=====

TEST CASE 1: Relevant retirement planning question

QUESTION: What are the steps that I should take to determine how much I need to save for retirement

=====

... Batches: 100% 1/1 [00:00<00:00, 19.93it/s]

... Batches: 100% 1/1 [00:00<00:00, 23.16it/s]

... Batches: 100% 1/1 [00:00<00:00, 25.95it/s]

... Batches: 100% 1/1 [00:00<00:00, 23.84it/s]

...

TEST CASE 1 RESULT:

```
{'best_score': 0.88550532646661,
'messages': ["As a Fidelity financial advisor, I'd be happy to help you determine how much you "
"need to save for retirement.\n"
'\n'
'According to our research, here are some general guidelines to consider:\n'
'\n'
'* At 30, aim to save at least 1x your salary\n'
'* By 40, increase savings to 3x your salary\n'
'* By 50, aim for 6x your salary\n'
'* By 60, target 8x your salary\n'
'* By 67 (traditional retirement age), strive for 10x your salary\n'
'\n'
'To get a more accurate estimate, consider the following steps:\n'
'\n'
'* Calculate your desired retirement income based on your expenses and lifestyle '
'goals.\n'
'* Estimate how much you'll need to save each year to reach your goal.\n"
'* Consider using our Retirement Savings Calculator tool on Fidelity.com to get '
```

Spaces: 4 CRLF ⌂ Cell 2 of 30 8:58 PM 4/10/2025

File Edit Selection View Go Run ...

← → ⌂ Bonus Assignment_2

Code + Markdown ▶ Run All ⚡ Restart ⌛ Clear All Outputs ⌂ Jupyter Variables ⌂ Outline ...

==== Running Retirement Planning Test Cases ===

=====

TEST CASE 1: Relevant retirement planning question

=====

TEST CASE 1: Relevant retirement planning question

QUESTION: What are the steps that I should take to determine how much I need to save for retirement

=====

... Batches: 100% 1/1 [00:00<00:00, 19.93it/s]

... Batches: 100% 1/1 [00:00<00:00, 23.16it/s]

... Batches: 100% 1/1 [00:00<00:00, 25.95it/s]

... Batches: 100% 1/1 [00:00<00:00, 23.84it/s]

...

TEST CASE 1 RESULT:

```
{'best_score': 0.88550532646661,
'messages': ["As a Fidelity financial advisor, I'd be happy to help you determine how much you "
"need to save for retirement.\n"
'\n'
'According to our research, here are some general guidelines to consider:\n'
'\n'
'* At 30, aim to save at least 1x your salary\n'
'* By 40, increase savings to 3x your salary\n'
'* By 50, aim for 6x your salary\n'
'* By 60, target 8x your salary\n'
'* By 67 (traditional retirement age), strive for 10x your salary\n'
'\n'
'To get a more accurate estimate, consider the following steps:\n'
'\n'
'* Calculate your desired retirement income based on your expenses and lifestyle '
'goals.\n'
'* Estimate how much you'll need to save each year to reach your goal.\n"
'* Consider using our Retirement Savings Calculator tool on Fidelity.com to get '
```

Spaces: 4 CRLF ⌂ Cell 2 of 30 8:58 PM 4/10/2025

File Edit Selection View Go Run ...

← → ⌂ Bonus Assignment_2

Code + Markdown ▶ Run All ⚡ Restart ⌛ Clear All Outputs ⌂ Jupyter Variables ⌂ Outline ...

==== Running Retirement Planning Test Cases ===

=====

TEST CASE 1: Relevant retirement planning question

=====

TEST CASE 1: Relevant retirement planning question

QUESTION: What are the steps that I should take to determine how much I need to save for retirement

=====

... Batches: 100% 1/1 [00:00<00:00, 19.93it/s]

... Batches: 100% 1/1 [00:00<00:00, 23.16it/s]

... Batches: 100% 1/1 [00:00<00:00, 25.95it/s]

... Batches: 100% 1/1 [00:00<00:00, 23.84it/s]

...

TEST CASE 1 RESULT:

```
{'best_score': 0.88550532646661,
'messages': ["As a Fidelity financial advisor, I'd be happy to help you determine how much you "
"need to save for retirement.\n"
'\n'
'According to our research, here are some general guidelines to consider:\n'
'\n'
'* At 30, aim to save at least 1x your salary\n'
'* By 40, increase savings to 3x your salary\n'
'* By 50, aim for 6x your salary\n'
'* By 60, target 8x your salary\n'
'* By 67 (traditional retirement age), strive for 10x your salary\n'
'\n'
'To get a more accurate estimate, consider the following steps:\n'
'\n'
'* Calculate your desired retirement income based on your expenses and lifestyle '
'goals.\n'
'* Estimate how much you'll need to save each year to reach your goal.\n"
'* Consider using our Retirement Savings Calculator tool on Fidelity.com to get '
```

Spaces: 4 CRLF ⌂ Cell 2 of 30 8:58 PM 4/10/2025

File Edit Selection View Go Run ...

← → ⌂ Bonus Assignment_2

Code + Markdown ▶ Run All ⚡ Restart ⌛ Clear All Outputs ⌂ Jupyter Variables ⌂ Outline ...

==== Running Retirement Planning Test Cases ===

=====

TEST CASE 1: Relevant retirement planning question

=====

TEST CASE 1: Relevant retirement planning question

QUESTION: What are the steps that I should take to determine how much I need to save for retirement

=====

... Batches: 100% 1/1 [00:00<00:00, 19.93it/s]

... Batches: 100% 1/1 [00:00<00:00, 23.16it/s]

... Batches: 100% 1/1 [00:00<00:00, 25.95it/s]

... Batches: 100% 1/1 [00:00<00:00, 23.84it/s]

...

TEST CASE 1 RESULT:

```
{'best_score': 0.88550532646661,
'messages': ["As a Fidelity financial advisor, I'd be happy to help you determine how much you "
"need to save for retirement.\n"
'\n'
'According to our research, here are some general guidelines to consider:\n'
'\n'
'* At 30, aim to save at least 1x your salary\n"
'* By 40, increase savings to 3x your salary\n"
'* By 50, aim for 6x your salary\n"
'* By 60, target 8x your salary\n"
'* By 67 (traditional retirement age), strive for 10x your salary\n"
'\n'
'To get a more accurate estimate, consider the following steps:\n"
'\n'
'* Calculate your desired retirement income based on your expenses and lifestyle '
'goals.\n"
'* Estimate how much you'll need to save each year to reach your goal.\n"
'* Consider using our Retirement Savings Calculator tool on Fidelity.com to get '
```

Spaces: 4 CRLF ⌂ Cell 2 of 30 8:58 PM 4/10/2025

Bonus Assignment_2

```

=====
TEST CASE 2: Similar semantic question about retirement
=====

TEST CASE 2: Similar semantic question about retirement
QUESTION: How can I calculate my retirement savings goal if I'm 23 now?
=====

... Batches: 100% [green bar] 1/1 [0:00<0:00, 15.87it/s]
... Batches: 100% [green bar] 1/1 [0:00<0:00, 31.44it/s]
... Batches: 100% [green bar] 1/1 [0:00<0:00, 33.16it/s]
... Batches: 100% [green bar] 1/1 [0:00<0:00, 33.38it/s]
...

TEST CASE 2 RESULT:
{'best_score': 0.8499524372195334,
 'messages': ["As a Fidelity financial advisor, I'd be happy to help you calculate your retirement " +
             "savings goal.\n" +
             "\n" +
             "To determine your retirement savings goal, we'll use the following assumptions:\n" +
             "\n" +
             "* You plan to retire at age 67 (or later)\n" +
             "* You expect to live for 25-30 years in retirement\n" +
             "* You want to maintain your current lifestyle in retirement\n" +
             "\n" +
             "Based on Fidelity's guidelines, here are some steps to help you calculate your " +
             "retirement savings goal:\n" +
             "\n" +
             "* Determine your desired retirement age: If you plan to retire at age 67, that's a " +
             "good starting point.\n" +
             "* Estimate your annual expenses in retirement: Consider how much you'll need to " +
             "maintain your current lifestyle.\n" +
             "* Calculate your required income in retirement: Use Fidelity's Retirement Savings " +
             "Calculator or consult with a financial advisor to determine how much you'll need to " +
             "save.\n"]
}

```

Bonus Assignment_2

```

=====
TEST CASE 3: Irrelevant question (non-retirement topic)
inputs = [
    "messages": [
        {"user": "What's the best recipe for chocolate chip cookies?"}
    ]
]
result3 = run_test_case(inputs["messages"][0][1], 3, "Irrelevant question (non-retirement topic)")
formatted_result3 = format_test_result(result3)
print("\nTEST CASE 3 RESULT:")
pprint.pprint(formatted_result3, width=100)
print("=*80)

0.0s
=====

TEST CASE 3: Irrelevant question (non-retirement topic)
=====

TEST CASE 3: Irrelevant question (non-retirement topic)
QUESTION: What's the best recipe for chocolate chip cookies?
=====

TEST CASE 3 RESULT:
{'best_score': 0,
 'messages': ["I specialize in retirement planning questions. This doesn't appear to be related to " +
              "retirement. Could you ask about retirement savings or planning?"],
 'retrieved_docs': 0,
 'status': 'rejected'}
=====

Block 8: PDF Processing Setup
=====

print("\n==== Processing PDF Document ===")

```

File Edit Selection View Go Run ... Bonus Assignment_2

OPEN EDITORS [unsaved]

- Fidelity-Langgraph-Agentic.RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb**
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...**
- Bonus Assignment_2.pdf**
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb**
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf**
- Fidelity-Langgraph-Agen...

```

Experiment4.ipynb > M+ Block 8: PDF Processing Setup > #<#> Block 8: PDF Processing
+ Code + Markdown | ▶ Run All ⚡ Restart ⌛ Clear All Outputs | Jupyter Variables | Outline ...
D+ v
# Continue with standard processing
pdf_nodes = text_splitter.get_nodes_from_documents(pdf_docs[:3]) # Process first 3 pages
pdf_index = VectorStoreIndex(pdf_nodes)
pdf_engine = pdf_index.as_query_engine()

print("\nProcessing complete! Vector index created with first 3 pages.")

[12] ✓ 1.8s
=====
Page 2 Metadata: {'total_pages': 6, 'file_path': './FFFGX.pdf', 'source': '2'}
Content Preview (first 300 chars):
Allocation
Asset Allocation 1,2,9
AS OF 11/30/2024
U.S. Equities
54.95%
Non-U.S. Equities
40.75%
Bonds
9.84%
Short-Term Debt & Net Other Assets
-5.54%
Fidelity Freedom® 2045 Fund: Investment Approach
1 Fidelity Freedom® Funds (the Funds) are designed so that the target date referenced in the
Fund na...
=====

Page 3 Metadata: {'total_pages': 6, 'file_path': './FFFGX.pdf', 'source': '3'}
Content Preview (first 300 chars):
Allocation
Fund Overview (continued)
Additional Disclosures
This description is only intended to provide a brief overview of the mutual fund. Read the fund's
prospectus for more detailed information about the fund.
Page 3 of 6

```

0 0 23 0 0 files and 15 cells to analyze

File Edit Selection View Go Run ... Bonus Assignment_2

OPEN EDITORS [unsaved]

- Fidelity-Langgraph-Agentic.RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb**
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf**
- Fidelity-Langgraph-Agen...

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...**
- Bonus Assignment_2.pdf**
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb**
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf**
- Fidelity-Langgraph-Agen...

```

Experiment4.ipynb > M+ Block 8: PDF Processing Setup > #<#> Block 8: PDF Processing
+ Code + Markdown | ▶ Run All ⚡ Restart ⌛ Clear All Outputs | Jupyter Variables | Outline ...
D+ v
    return base64.b64encode(buffer.getvalue()).decode("utf-8")

base64_image = pdf_page_to_base64("./FFFGX.pdf", 1)

from IPython.display import Image as IPImage, display
display(IPImage(data=base64.b64decode(base64_image)))

print("PDF first page displayed!")

[13] ✓ 0.1s
=====
Displaying PDF Preview ==



```

0 0 23 0 0 files and 15 cells to analyze

Bonus Assignment 2

```

Fidelity-Langgraph-Agentic.RAG.ipynb • Experiment1.ipynb • Experiment2.ipynb • Experiment3.ipynb • Experiment4.ipynb × Experiment5.ipynb • Experiment6.ipynb • Experiment7.ipynb • Experiment8.ipynb
Experiment4.ipynb > M+ Block 8: PDF Processing Setup > # Block 8: PDF Processing
+ Code + Markdown | ▶ Run All ⚡ Restart ⚡ Clear All Outputs | Jupyter Variables | Outline ...
answer = query_pdf(question, num)
pdf_results = [{"question": question, "answer": answer}]
print(f"Q1: {question}\nA1: {answer}\n")
[15] ✓ 1.3s
...
== Running PDF Queries ==
PDF QUESTION 1: What is the name of this fund?
Batches: 100% 1/1 [00:00<00:00, 1.83it/s]
ANSWER: Fidelity Freedom® 2045 Fund
-----
Q1: What is the name of this fund?
A1: Fidelity Freedom® 2045 Fund

# Question 2
question = "Who is the fund manager?"
num = 2
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q2: {question}\nA2: {answer}\n")
[16] ✓ 2.6s
...
PDF QUESTION 2: Who is the fund manager?
Batches: 100% 1/1 [00:01<00:00, 1.68s/it]
ANSWER: The fund managers are Andrew J. Dierdorf, who has been with the fund since June 21, 2011, and Brett F. Sumson, who has been with the fund since ...
Q2: Who is the fund manager?
A2: The fund managers are Andrew J. Dierdorf, who has been with the fund since June 21, 2011, and Brett F. Sumson, who has been with the fund since ...

```

File Edit Selection View Go Run ... ↵ → 🔍 Bonus Assignment_2

EXPLORER OPEN EDITORS 8 unsaved

- Fidelity-Langgraph-...
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb ×
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chroma
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb ×
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

OUTLINE TIMELINE

0 23 0 files and 15 cells to analyze

4°C Cloudy Search

CR LF Cell 20 of 30

ENG US 8:59 PM 4/10/2025

Experiment 5:

Bonus Assignment_2

```

Fidelity-Langgraph-Agentic.RAG.ipynb • Experiment1.ipynb • Experiment2.ipynb • Experiment3.ipynb • Experiment4.ipynb • Experiment5.ipynb • Experiment6.ipynb • Experiment7.ipynb • Experiment8.ipynb
Experiment5.ipynb > M+ Block 1: Setup and Imports > # Core imports
+ Code + Markdown | ▶ Run All ⚡ Restart ⚡ Clear All Outputs | Jupyter Variables | Outline ...
vector_store = ChromaVectorStore(chroma_collection=chroma_collection)
storage_context = StorageContext.from_defaults(vector_store=vector_store)
index = VectorStoreIndex(nodes, storage_context=storage_context)

# Create retriever
retriever = index.as_retriever(similarity_top_k=3)
print("Document processing complete!")
[2] ✓ 17.1s
...
== Loading and processing retirement planning document ==
Batches: 100% 1/1 [00:02<00:00, 2.44s/it]
Batches: 100% 1/1 [00:02<00:00, 2.76s/it]
Batches: 100% 1/1 [00:02<00:00, 2.25s/it]
Batches: 100% 1/1 [00:02<00:00, 2.09s/it]
Batches: 100% 1/1 [00:02<00:00, 2.14s/it]
Batches: 100% 1/1 [00:02<00:00, 2.01s/it]
Batches: 100% 1/1 [00:00<00:00, 1.21it/s]
...
Document processing complete!

Block 3: Create Retriever Tool

print("\n== Creating retriever tool ==")

retriever_tool = QueryEngineTool.from_defaults(
    query_engine=retriever,
    name="retrieve_retirement_info",
    description="Search and return information for calculating retirement needs",
)
[3]

```

File Edit Selection View Go Run ... ↵ → 🔍 Bonus Assignment_2

EXPLORER OPEN EDITORS 8 unsaved

- Fidelity-Langgraph-...
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb ×
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

BONUS ASSIGNMENT 2

- > chroma
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb ×
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

OUTLINE TIMELINE

0 23 0 files and 15 cells to analyze

4°C Cloudy Search

Spaces: 4 CRLF Cell 2 of 29

ENG US 9:01 PM 4/10/2025

Bonus Assignment_2

```

EXPLORER OPEN EDITORS [unsaved]
  ● Fidelity-Langgraph-Agents.ipynb
  ● Experiment1.ipynb
  ● Experiment2.ipynb
  ● Experiment3.ipynb
  ● Experiment4.ipynb
  ● Experiment5.ipynb
  ● Experiment6.ipynb
  ● Experiment7.ipynb
  ● Experiment8.ipynb

BONUS A... > chrome
  > chroma_db
  > venv
  ● 2.6.1
  ↳ Bonus Assignment_2 - In...
    ↳ Bonus Assignment_2.pdf
    Experiment1.ipynb
    Experiment2.ipynb
    Experiment3.ipynb
    Experiment4.ipynb
    Experiment5.ipynb
    Experiment7.ipynb
    Experiment8.ipynb
    Experiment6.ipynb
    ↳ FFFGX.pdf
    Fidelity-Langgraph-Agen...
  ↳ OUTLINE
  ↳ TIMELINE
  0 0 23 0 files and 15 cells to analyze

Code + Markdown ▶ Run All ⚡ Restart ⌂ Clear All Outputs Jupyter Variables Outline ...
  ● Experiment5.ipynb > M+ Block 1: Setup and Imports > # Core imports
  === Running Retirement Planning Test Cases ===
  =====
  TEST CASE 1: Relevant retirement planning question
  QUESTION: What are the steps that I should take to determine how much I need to save for retirement
  =====
  Batches: 100% 1/1 [00:00<00:00, 1.93it/s]

  ...
  --- Checking document relevance ---
  Document relevance: yes

  --- Generating response ---
  RESULT:
  { 'input': 'What are the steps that I should take to determine how much I need to save for '
    'retirement',
    'output': '\n'
      '\n'
      "As a Fidelity financial advisor, I'd be happy to help you determine how much you need "
      "to save for retirement. Here are the steps to follow:\n"
      '\n'
      '**Step 1: Determine Your Retirement Goals**\n'
      '\n'
      "** Estimate how much you'll need to maintain your lifestyle in retirement. A general "
      "rule of thumb is to aim for 70% to 80% of your pre-retirement income.\n"
      "** Consider your retirement goals, such as travel, hobbies, or supporting loved ones.\n"
      '\n'
      '**Step 2: Calculate Your Retirement Savings Factor**\n'
      '\n'
      "** Use Fidelity's age-based retirement savings factors as a guideline:\n"
      '\t+ By age 30: Save 1x your annual income\n'
      '\t+ By age 35: Save 2x your annual income\n'
      '\t+ By age 40: Save 3x your annual income\n'
      '\n'
      'to save for retirement and creating a plan to achieve your goals.',

  ...
  
```

Spaces: 4 CRLF ENG US 9:01 PM Cell 2 of 29 4/10/2025

Bonus Assignment_2

```

EXPLORER OPEN EDITORS [unsaved]
  ● Fidelity-Langgraph-Agents.ipynb
  ● Experiment1.ipynb
  ● Experiment2.ipynb
  ● Experiment3.ipynb
  ● Experiment4.ipynb
  ● Experiment5.ipynb
  ● Experiment6.ipynb
  ● Experiment7.ipynb
  ● Experiment8.ipynb

BONUS A... > chrome
  > chroma_db
  > venv
  ● 2.6.1
  ↳ Bonus Assignment_2 - In...
    ↳ Bonus Assignment_2.pdf
    Experiment1.ipynb
    Experiment2.ipynb
    Experiment3.ipynb
    Experiment4.ipynb
    Experiment5.ipynb
    Experiment7.ipynb
    Experiment8.ipynb
    Experiment6.ipynb
    ↳ FFFGX.pdf
    Fidelity-Langgraph-Agen...
  ↳ OUTLINE
  ↳ TIMELINE
  0 0 23 0 files and 15 cells to analyze

Code + Markdown ▶ Run All ⚡ Restart ⌂ Clear All Outputs Jupyter Variables Outline ...
  ● Experiment5.ipynb > M+ Block 1: Setup and Imports > # Core imports
  === Running Retirement Planning Test Cases ===
  =====
  TEST CASE 2: Similar semantic question about retirement
  QUESTION: How can I calculate my retirement savings goal if im 23 now?
  =====
  Batches: 100% 1/1 [00:00<00:00, 23.91it/s]

  ...
  --- Checking document relevance ---
  Document relevance: yes

  --- Generating response ---
  RESULT:
  { 'input': 'How can I calculate my retirement savings goal if im 23 now?',
    'output': '\n'
      '\n'
      "As a Fidelity financial advisor, I'd be happy to help you calculate your retirement "
      "savings goal.\n"
      '\n'
      "Since you're 23 now, let's break down the steps to estimate your retirement savings "
      "goal based on Fidelity's guidelines:\n"
      '\n'
      '**Step 1: Determine your desired retirement age**\n'
      "You didn't mention a specific retirement age, but Fidelity assumes a retirement age "
      "of 67. If you want to retire earlier or later, adjust this number accordingly.\n"
      '\n'
      '**Step 2: Estimate your preretirement income**\n'
      "What's your current income? Let's assume it's $50,000 per year. As you progress in "
      "your career, your income will likely increase. For simplicity, let's assume your "
      "preretirement income will be around $80,000 per year.\n"
      '\n'
      '**Step 3: Calculate your retirement savings goal**\n'
      '\n'
      'track to meet your retirement goals.',

  ...
  'status': 'relevant'

  ...
  
```

Spaces: 4 CRLF ENG US 9:01 PM Cell 2 of 29 4/10/2025

Bonus Assignment_2

OPEN EDITORS [8 unsaved]

- Fidelity-Langgraph-Agentic.RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiments6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS A...

> chrome
 > chroma_db
 > venv
 ≡ 2.6.1
 ↳ Bonus Assignment_2 - In...
 ↳ Bonus Assignment_2.pdf
 Experiment1.ipynb
 Experiment2.ipynb
 Experiment3.ipynb
 Experiment4.ipynb
 Experiment5.ipynb
 Experiment7.ipynb
 Experiment8.ipynb
 Experiment6.ipynb
 ↳ FFFGX.pdf
 Fidelity-Langgraph-Agen...

OUTLINE

TIMELINE [10]

File Edit Selection View Go Run ...

Code + Markdown | ▶ Run All ⚡ Restart ⌛ Clear All Outputs | Jupyter Variables | Outline ...

```
What's the best recipe for chocolate chip cookies?  
3, "Irrelevant question (non-retirement topic)"  
)  
✓ 4.3s
```

TEST CASE 3: Irrelevant question (non-retirement topic)
QUESTION: What's the best recipe for chocolate chip cookies?

Batches: 100% 1/1 [00:01<00:00, 1.66s/it]

--- Checking document relevance ---
Document relevance: no
RESULT:
{ 'input': "What's the best recipe for chocolate chip cookies?",
 'output': 'I can only answer questions about retirement planning.',
 'status': 'rejected (irrelevant)'}

Block 8: PDF Processing Setup

```
print("\n==== Processing PDF Document ===")  
  
from llama_index.readers.file import PyMuPDFReader  
  
# Load PDF  
loader = PyMuPDFReader()  
pdf_docs = loader.load(file_path="./FFFGX.pdf")  
# Process PDF
```

Spaces: 4 CRLF ENG US 9:01 PM 4/10/2025

EXPLORER [8 unsaved]

- Fidelity-Langgraph-Agentic.RAG.ipynb
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiments6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS A...

> chrome
 > chroma_db
 > venv
 ≡ 2.6.1
 ↳ Bonus Assignment_2 - In...
 ↳ Bonus Assignment_2.pdf
 Experiment1.ipynb
 Experiment2.ipynb
 Experiment3.ipynb
 Experiment4.ipynb
 Experiment5.ipynb
 Experiment7.ipynb
 Experiment8.ipynb
 Experiment6.ipynb
 ↳ FFFGX.pdf
 Fidelity-Langgraph-Agen...

OUTLINE

TIMELINE [10]

File Edit Selection View Go Run ...

Code + Markdown | ▶ Run All ⚡ Restart ⌛ Clear All Outputs | Jupyter Variables | Outline ...

```
# QUESTION 3  
question = "What is the calendar year return for 2022 for this fund and S&P 500?"  
num = 3  
answer = query_pdf(question, num)  
pdf_results.append({"question": question, "answer": answer})  
print(f"Q3: {question}\nA3: {answer}\n")
```

1.6s

PDF QUESTION 3: What is the calendar year return for 2022 for this fund and S&P 500?

Batches: 100% 1/1 [00:00<00:00, 19.09it/s]

ANSWER: The calendar year return for 2022 for the Fidelity Freedom® 2045 Fund is -18.26%, while the return for the S&P 500 is -18.11%.

Q3: What is the calendar year return for 2022 for this fund and S&P 500?
A3: The calendar year return for 2022 for the Fidelity Freedom® 2045 Fund is -18.26%, while the return for the S&P 500 is -18.11%.

```
# Question 4  
question = "What is the Portfolio Net Assets?"  
num = 4  
answer = query_pdf(question, num)  
pdf_results.append({"question": question, "answer": answer})  
print(f"Q4: {question}\nA4: {answer}\n")
```

0.8s

PDF QUESTION 4: What is the Portfolio Net Assets?

Batches: 100% 1/1 [00:00<00:00, 24.55it/s]

ANSWER: The Portfolio Net Assets amount to \$21,466.13 million as of 11/30/2024.

Q4: What is the Portfolio Net Assets?
A4: The Portfolio Net Assets amount to \$21,466.13 million as of 11/30/2024.

Spaces: 4 CRLF ENG US 9:02 PM 4/10/2025

Experiment 6:

The screenshot shows a Jupyter Notebook interface with the title "Bonus Assignment_2". The left sidebar lists files in the "OPEN EDITORS" and "BONUS ASSIGNMENT 2" sections. The main area displays the code for Experiment 6.ipynb, which prints a test case about retirement planning. The output cell shows the result of the test case execution.

```
File Edit Selection View Go Run ... EXPLORE OPEN EDITORS 8 unsaved Experiment2.ipynb Experiment3.ipynb Experiment4.ipynb Experiment5.ipynb Experiment6.ipynb Experiment7.ipynb Experiment8.ipynb BONUS ASSIGNMENT 2 chroma chroma_db venv 2.6.1 Bonus Assignment_2 - In... Bonus Assignment_2.pdf Experiment1.ipynb Experiment2.ipynb Experiment3.ipynb Experiment4.ipynb Experiment5.ipynb Experiment7.ipynb Experiment8.ipynb Experiment6.ipynb FFFGX.pdf Fidelity-Langgraph-Agen... Code Markdown Run All Restart Clear All Outputs Jupyter Variables Outline ... venv (Python 3.10.0) What are the steps that I should take to determine how much I need to save for retirement? 1, "Relevant retirement planning question" 10.6s TEST CASE 1: Relevant retirement planning question QUESTION: What are the steps that I should take to determine how much I need to save for retirement ===== --- Checking document relevance --- Document relevance: yes --- Generating response --- RESULT: { 'input': 'What are the steps that I should take to determine how much I need to save for ' 'retirement', 'output': 'Determining how much you need to save for retirement can feel overwhelming, but ' 'breaking it down into manageable steps can help clarify your goals. Here's a ' 'structured approach to guide you through the process:\n' '\n' '### Steps to Determine Retirement Savings Needs\n' '\n' '1. **Estimate Your Retirement Age**:\n' '- Decide when you plan to retire. Common retirement ages are between 62 and 67, ' 'but this can vary based on personal circumstances.\n' '\n' '2. **Calculate Your Desired Annual Retirement Income**:\n' '- Consider how much money you will need each year in retirement to maintain your ' 'desired lifestyle. A common rule of thumb is to aim for about 70% to 80% of your ' 'pre-retirement income.\n' '\n' '3. **Determine the Length of Your Retirement**:\n' '- Estimate how long you expect to be in retirement. This can be based on your life ' }
```

The screenshot shows a Jupyter Notebook interface with the title "Bonus Assignment_2". The left sidebar lists files in the "OPEN EDITORS" and "BONUS ASSIGNMENT 2" sections. The main area displays the code for Experiment 6.ipynb, which runs a test case about calculating retirement savings goals. The output cell shows the result of the test case execution.

```
File Edit Selection View Go Run ... EXPLORE OPEN EDITORS 8 unsaved Experiment2.ipynb Experiment3.ipynb Experiment4.ipynb Experiment5.ipynb Experiment6.ipynb Experiment7.ipynb Experiment8.ipynb BONUS ASSIGNMENT 2 chroma chroma_db venv 2.6.1 Bonus Assignment_2 - In... Bonus Assignment_2.pdf Experiment1.ipynb Experiment2.ipynb Experiment3.ipynb Experiment4.ipynb Experiment5.ipynb Experiment7.ipynb Experiment8.ipynb Experiment6.ipynb FFFGX.pdf Fidelity-Langgraph-Agen... Code Markdown Run All Restart Clear All Outputs Jupyter Variables Outline ... venv (Python 3.10.0) result2 = run_test_case() 15.9s TEST CASE 2: Similar semantic question about retirement QUESTION: How can I calculate my retirement savings goal if im 23 now? ===== --- Checking document relevance --- Document relevance: yes --- Generating response --- RESULT: { 'input': 'How can I calculate my retirement savings goal if im 23 now?', 'output': 'To calculate your retirement savings goal at age 23, follow these steps:\n' '\n' '1. **Determine Your Target Retirement Age**: Decide when you plan to retire. For this ' 'example, let's assume you plan to retire at age 67, which gives you 44 years to ' 'save.\n' '\n' '2. **Estimate Your Preretirement Income**: Consider what your income will be when you ' 'are nearing retirement. For example, if you expect to earn $70,000 annually at that ' 'time, this will be your baseline for calculating your savings goal.\n' '\n' '3. **Calculate Your Savings Factor**: Fidelity suggests saving **10 times** your ' 'preretirement income by age 67 to maintain your lifestyle. Using the example income ' 'of $70,000:\n' '- **Savings Goal** = 10 x $70,000 = **$700,000**.\n' '\n' '4. **Set Milestones**: Fidelity recommends age-based savings milestones to help you ' 'stay on track:\n'
```

File Edit Selection View Go Run ... ↵ → 🔍 Bonus Assignment_2

OPEN EDITORS [8 unsaved]

- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFFGX.pdf
- Fidelity-Langgraph-Agen...

```

result3 = run_test_case(
    "What's the best recipe for chocolate chip cookies?",
    3, "Irrelevant question (non-retirement topic)"
)
[9] ✓ 0.7s
...
=====
TEST CASE 3: Irrelevant question (non-retirement topic)
QUESTION: What's the best recipe for chocolate chip cookies?
=====

--- Checking document relevance ---
Document relevance: no
RESULT:
{ 'input': "What's the best recipe for chocolate chip cookies?", 
  'output': 'I can only answer questions about retirement planning.', 
  'status': 'rejected (irrelevant)'}
=====

print("\n==== Processing PDF Document ====")

from llama_index.readers.file import PyMuPDFReader

loader = PyMuPDFReader()
pdf_docs = loader.load(file_path='./FFFFGX.pdf')

```

4°C Cloudy

File Edit Selection View Go Run ... 🔍 Bonus Assignment_2

OPEN EDITORS [12 unsaved]

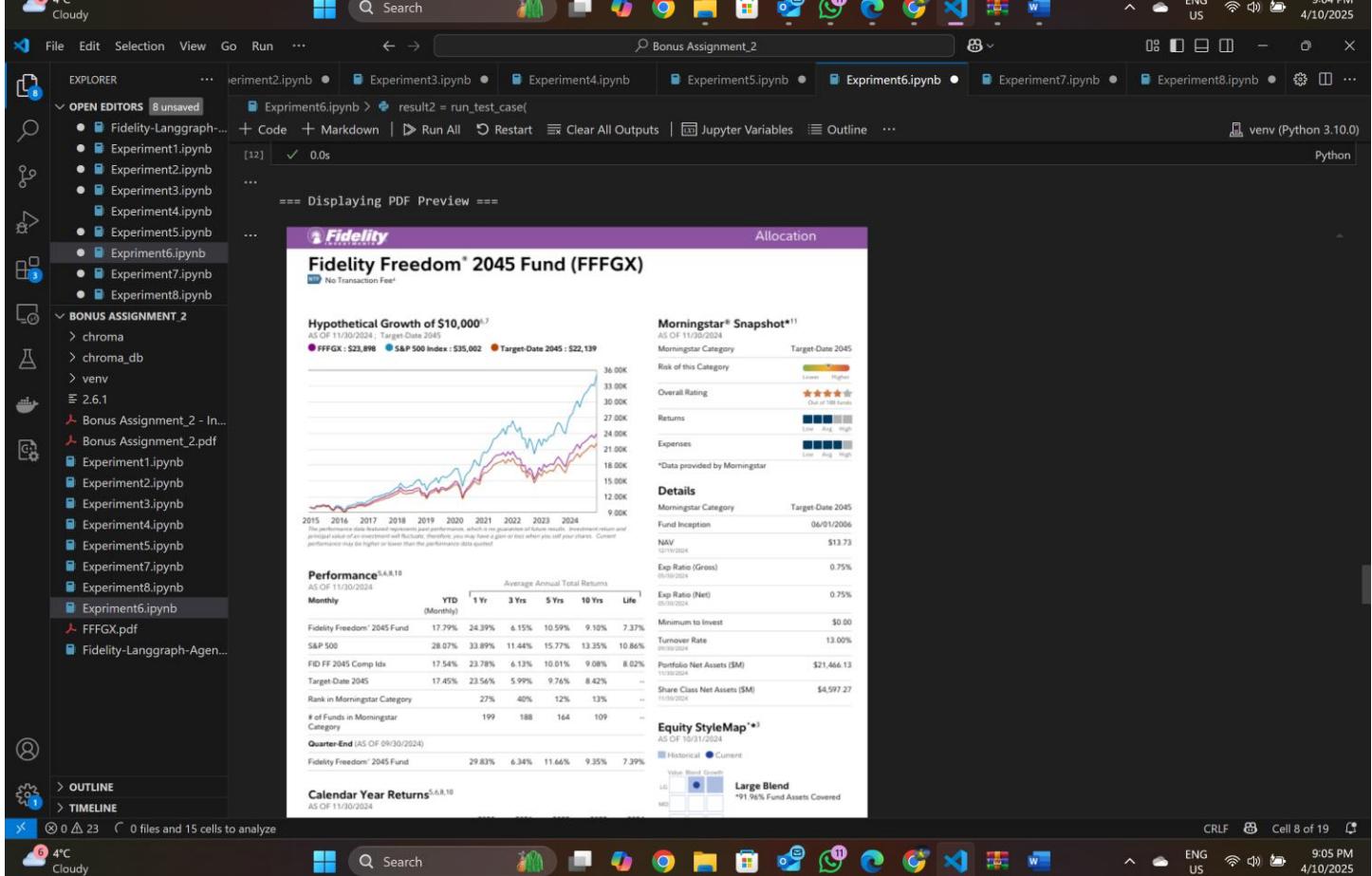
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFFGX.pdf
- Fidelity-Langgraph-Agen...

0 0 23 0 0 files and 15 cells to analyze

4°C Cloudy



Bonus Assignment_2

```

# Question 1
question = "What is the name of this fund?"
num = 1
answer = query_pdf(question, num)
pdf_results = [{"question": question, "answer": answer}]
print(f"Q1: {question}\nA1: {answer}\n")

[14] ✓ 0.8s
...  

    === Running PDF Queries ===  

PDF QUESTION 1: What is the name of this fund?  

ANSWER: Fidelity Freedom® 2045 Fund  

-----  

Q1: What is the name of this fund?  

A1: Fidelity Freedom® 2045 Fund

```



```

# Question 2
question = "Who is the fund manager?"
num = 2
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q2: {question}\nA2: {answer}\n")

[15] ✓ 1.3s
...  

PDF QUESTION 2: Who is the fund manager?  

ANSWER: The fund managers are Andrew J. Dierdorf, who has been with the fund since June 21, 2011, and Brett F. Sumson, who has been with the fund :  

-----  

Q2: Who is the fund manager?  

A2: The fund managers are Andrew J. Dierdorf, who has been with the fund since June 21, 2011, and Brett F. Sumson, who has been with the fund since

```

File Edit Selection View Go Run ... 🔍 Bonus Assignment_2 🌐 v 0 23 0 files and 15 cells to analyze

Experiment 7:

Block 3: Agent State Definition

```

class AgentState(TypedDict):
    messages: Annotated[Sequence[BaseMessage], add_messages]

```

Block 4: Node Functions

```

def grade_documents(state) -> dict:
    print("\n--- Checking document relevance ---")
    messages = state["messages"]
    question = messages[0].content
    docs = messages[-1].content

```

File Edit Selection View Go Run ... 🔍 Bonus Assignment_2 🌐 v 0 23 0 files and 15 cells to analyze

File Edit Selection View Go Run ... ↵ → Bonus Assignment_2

EXPLORER OPEN EDITORS 8 unsaved

- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chroma
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

Code + Markdown ▶ Run All ⚡ Restart ⌛ Clear All Outputs Jupyter Variables Outline ...

Experiment7.ipynb > M4 Block 1: Imports and Setup > # Setup

```
result1 = run_test_case(question1, 1, "Relevant retirement planning question")
```

[7] ✓ 33.1s

...
=====
TEST CASE 1: Relevant retirement planning question
QUESTION: What are the steps that I should take to determine how much I need to save for retirement
=====

--- Calling agent ---
--- Checking document relevance ---
<think>
Okay, let's see. The user is asking about the steps to determine how much they need to save for retirement. The provided document mentions Fidelity'</think>

The document provides specific savings targets by age (e.g., 10x income by 67) and mentions factors influencing retirement savings goals (retirement
Score: yes Document relevance: relevant
--- Generating response ---
<think>
Okay, so the user is asking about the steps to determine how much they need to save for retirement. Let me start by recalling Fidelity's guidelines
First, I need to break down the process into clear steps. The article mentions age-based milestones: 1x by 30, 3x by 40, 6x by 50, 8x by 60, and 10x
...
 on-demand webinars or consult a financial advisor.',
'status': 'relevant')

=====

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

Spaces: 4 CRLF ENG US 9:06 PM 4/10/2025

File Edit Selection View Go Run ... ↵ → Bonus Assignment_2

EXPLORER OPEN EDITORS 8 unsaved

- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

BONUS ASSIGNMENT 2

- > chroma
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

Code + Markdown ▶ Run All ⚡ Restart ⌛ Clear All Outputs Jupyter Variables Outline ...

Experiment7.ipynb > M4 Block 8: Run Test Cases > question2 = "How can I calculate my retirement savings goal if im 23 now?"

```
result2 = run_test_case(question2, 2, "Similar semantic question about retirement")
```

[8] ✓ 55.5s

...
TEST CASE 2: Similar semantic question about retirement
QUESTION: How can I calculate my retirement savings goal if im 23 now?
=====

--- Calling agent ---
--- Checking document relevance ---
<think>
Okay, let's see. The user is asking how to calculate their retirement savings goal now that they're 23. The retrieved document talks about saving
The user is 23, which is just before the starting age mentioned in the document (25). The document provides guidelines based on starting at 25, s</think>
yesDocument relevance: relevant
--- Generating response ---
<think>
Okay, let's tackle this retirement savings question. The user is 23 and wants to know how to calculate their retirement savings goal. From the co
First, I need to break down the steps. The user is 23, so they're actually starting a bit earlier than the assumed 25. That's a good head start.
Step 1: Determine the desired retirement age. The default here is 67, but if they retire earlier or later, the multiplier changes. Since the user
Step 2: Estimate pre-retirement income. This is tricky because income can change over time. Maybe use current income as a starting point, but ack
Wait, but how do they calculate 10x if income isn't known? Maybe use current income adjusted for inflation or expected growth. Alternatively, use
But how does that translate to the 10x multiplier? The 10x is the amount saved by 67, which, when combined with Social Security, replaces 45% of

CRLF ENG US 9:07 PM 4/10/2025

Bonus Assignment_2

```
question3 = "What's the best recipe for chocolate chip cookies?"
result3 = run_test_case(question3, 3, "Irrelevant question (non-retirement topic)")

[9] 0.0s
=====
TEST CASE 3: Irrelevant question (non-retirement topic)
QUESTION: What's the best recipe for chocolate chip cookies?
=====

--- Calling agent ---
RESULT:
{'input': "What's the best recipe for chocolate chip cookies?", 'output': 'I can only answer retirement planning questions.', 'status': 'irrelevant'}
```

Block 9: PDF Processing and Queries

```
from langchain_community.document_loaders import PyPDFLoader

# Use the full absolute path
file_path = "./FFFGX.pdf"

# Load the PDF
loader = PyPDFLoader(file_path)

# Store pages
pages = loader.load_and_split()

[10] 0.2s
...
Loaded 6 pages successfully!
```

Bonus Assignment_2

```
print(f"Loaded {len(pages)} pages successfully!")

[11] 0.0s
...
{'producer': '', 'creator': 'Quadient~Inspire~16.0.716.7', 'creationdate': '2024-12-21T04:30:27+00:00', 'title': 'MFL_Doc', 'source': './FFFGX.pdf'}
```

Allocation
Glossary Of Terms
Beta: A measure of a portfolio's sensitivity to market movements (as represented by a benchmark index). The benchmark index has a beta of 1.0. A beta more (less) than 1.0 indicates that a fund's historical returns have fluctuated more (less) than the benchmark index. Beta is a more reliable measure of volatility when used in combination with a high R2 which indicates a high correlation between the movements in a fund's returns and movements in a benchmark index.

Distribution and/or service fee(12b-1) Fees: The 12b-1 fee represents the maximum annual charge deducted from fund assets to pay for distribution and marketing costs. Total 12b-1 fees, excluding loads, are capped at 1.00% of average net assets annually. Of this, the distribution and marketing portion of the fee may account for up to 0.75%. The other portion of the overall 12b-1 fee, the service fee, may be used to pay for expenses such as legal, accounting, and audit fees.

Expense Ratio (Gross): Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of assets under management. It includes all operating expenses, such as management fees, administrative expenses, and distribution fees. The expense ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund when it sells shares. If the investment option is not a mutual fund, the expense ratio may be calculated using a different formula.

Expense Ratio (Net): Expense ratio is a measure of what it costs to operate an investment, expressed as a percentage of assets under management. It includes all operating expenses, such as management fees, administrative expenses, and distribution fees. The expense ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund when it sells shares. This ratio also includes Acquired Fund Fees and Expenses, which are expenses indirectly incurred by a fund through its ownership of shares in other investment companies. This number does not include the expense ratio of the fund itself. The expense ratio may be calculated using methodologies that differ from those used for mutual funds.

Fidelity Freedom 2045 Composite Index: Fidelity Freedom 2045 Composite Index is a customized blend of various asset classes, including equities, bonds, and cash equivalents, designed to provide a diversified investment opportunity.

FFFGX.pdf

Snipping Tool

Screenshot copied to clipboard
Automatically saved to screenshots folder.

Markup and share

File Edit Selection View Go Run ...

Bonus Assignment_2

OPEN EDITORS [8 unsaved]

- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

Experiment7.ipynb > M4 Block 8: Run Test Cases > question2 = "How can I calculate my retirement savings goal if im 23 now?"

```
# This requires some extra dependencies and is optional
pass
```

0.8s

...
BONUS ASSIGNMENT 2
> chrome
> chroma_db
> venv
2.6.1
Bonus Assignment_2 - In...
Bonus Assignment_2.pdf
Experiment1.ipynb
Experiment2.ipynb
Experiment3.ipynb
Experiment4.ipynb
Experiment5.ipynb
Experiment7.ipynb
Experiment8.ipynb
Experiment6.ipynb
FFFGX.pdf
Fidelity-Langgraph-Agen...

OUTLINE
TIMELINE

0 0 23 0 files and 15 cells to analyze

Cloudy 4°C 9:07 PM 4/10/2025

File Edit Selection View Go Run ...

Bonus Assignment_2

OPEN EDITORS [8 unsaved]

- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

Experiment7.ipynb > M4 Block 8: Run Test Cases > question2 = "How can I calculate my retirement savings goal if im 23 now?"

```
def query_pdf(question, question_number):
    print(f"\nPDF QUESTION {question_number}: {question}\n")
```

0.8s

def query_pdf(question, question_number):
 print(f"\nPDF QUESTION {question_number}: {question}\n")
 num = 4
 answer = query_pdf(question, num)
 pdf_results.append(("question": question, "answer": answer))
 print(f"Q4: {question}\nA4: {answer}\n")

2.3s

PDF QUESTION 4: What is the Portfolio Net Assets?
ANSWER: The Portfolio Net Assets for the Fidelity Freedom® 2045 Fund (FFFGX) is \$21,463,181,623 as of 11/30/2024.

Q4: What is the Portfolio Net Assets?
A4: The Portfolio Net Assets for the Fidelity Freedom® 2045 Fund (FFFGX) is \$21,463,181,623 as of 11/30/2024.

2.4s

Question 5
question = "What is the Morningstar rating for this fund?How many funds used to rate this fund?"
num = 5
answer = query_pdf(question, num)
pdf_results.append(("question": question, "answer": answer))
print(f"Q5: {question}\nA5: {answer}\n")

2.4s

PDF QUESTION 5: What is the Morningstar rating for this fund?How many funds used to rate this fund?
ANSWER: The Morningstar rating for the Fidelity Freedom® 2045 Fund (FFFGX) is **4 stars** as of 11/30/2024. The number of funds used to rate this fund is 5.

Q5: What is the Morningstar rating for this fund?How many funds used to rate this fund?
A5: The Morningstar rating for the Fidelity Freedom® 2045 Fund (FFFGX) is **4 stars** as of 11/30/2024. The number of funds used to rate this fund is 5.

20s

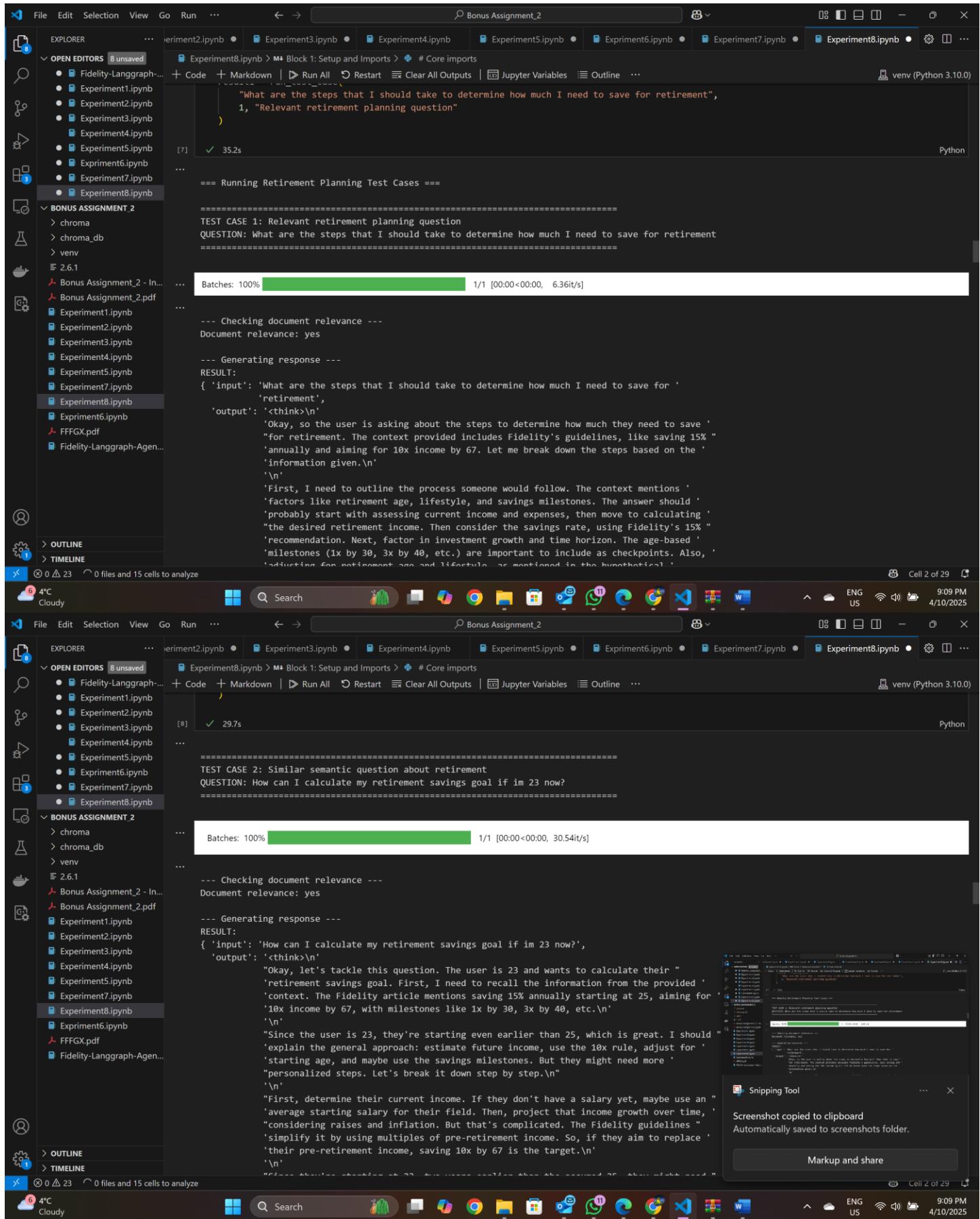
```
print("\n== TEST RESULTS SUMMARY ==")
print("\nRetirement Planning Questions:")
pprint.pprint([
    "test case 1": result1,
```

Code Markdown

0 0 23 0 files and 15 cells to analyze

Cloudy 4°C 9:08 PM 4/10/2025

Experiment 8:



The image shows two side-by-side screenshots of a Jupyter Notebook interface, both titled "Bonus Assignment_2".

Top Screenshot (Experiment 8.1):

- EXPLORER:** Shows multiple Jupyter notebooks: Experiment1.ipynb, Experiment2.ipynb, Experiment3.ipynb, Experiment4.ipynb, Experiment5.ipynb, Experiment6.ipynb, Experiment7.ipynb, and Experiment8.ipynb.
- Code Editor:** Displays code related to "Retirement Planning Test Cases".

```
    "What are the steps that I should take to determine how much I need to save for retirement",
    1, "Relevant retirement planning question"
)
[7] ✓ 35.2s
```

=====
TEST CASE 1: Relevant retirement planning question
QUESTION: What are the steps that I should take to determine how much I need to save for retirement
=====

Batches: 100% 1/1 [0:00<0:00, 6.36it/s]

```
...  
    --- Checking document relevance ---  
    Document relevance: yes  
  
    --- Generating response ---  
    RESULT:  
    { 'input': 'What are the steps that I should take to determine how much I need to save for '  
        'retirement',  
        'output': '<think>\n'  
            'Okay, so the user is asking about the steps to determine how much they need to save '  
            'for retirement. The context provided includes Fidelity's guidelines, like saving 15% '  
            'annually and aiming for 10x income by 67. Let me break down the steps based on the '  
            'information given.\n'  
            '\n'  
            'First, I need to outline the process someone would follow. The context mentions '  
            'factors like retirement age, lifestyle, and savings milestones. The answer should '  
            'probably start with assessing current income and expenses, then move to calculating '  
            'the desired retirement income. Then consider the savings rate, using Fidelity's 15% '  
            'recommendation. Next, factor in investment growth and time horizon. The age-based '  
            'milestones (1x by 30, 3x by 40, etc.) are important to include as checkpoints. Also, '  
            'adjusting for inflation, taxes, and lifestyle changes mentioned in the hypothetical' 
```
- Bottom Status Bar:** Shows "Cell 2 of 29", "4°C Cloudy", "ENG US", and the date "4/10/2025".

Bottom Screenshot (Experiment 8.2):

- EXPLORER:** Shows the same list of notebooks as the top screenshot.
- Code Editor:** Displays code related to "Similar semantic question about retirement".

```
    )  
[8] ✓ 29.7s
```

=====
TEST CASE 2: Similar semantic question about retirement
QUESTION: How can I calculate my retirement savings goal if im 23 now?
=====

Batches: 100% 1/1 [0:00<0:00, 30.54it/s]

```
...  
    --- Checking document relevance ---  
    Document relevance: yes  
  
    --- Generating response ---  
    RESULT:  
    { 'input': 'How can I calculate my retirement savings goal if im 23 now?',  
        'output': '<think>\n'  
            'Okay, let's tackle this question. The user is 23 and wants to calculate their "  
            'retirement savings goal. First, I need to recall the information from the provided '  
            'context. The Fidelity article mentions saving 15% annually starting at 25, aiming for '  
            '10x income by 67, with milestones like 1x by 30, 3x by 40, etc.\n'  
            '\n'  
            'Since the user is 23, they're starting even earlier than 25, which is great. I should '  
            'explain the general approach: estimate future income, use the 10x rule, adjust for '  
            'starting age, and maybe use the savings milestones. But they might need more '  
            'personalized steps. Let's break it down step by step.\n'  
            '\n'  
            'First, determine their current income. If they don't have a salary yet, maybe use an "  
            'average starting salary for their field. Then, project that income growth over time,  
            'considering raises and inflation. But that's complicated. The Fidelity guidelines "  
            'simplify it by using multiples of pre-retirement income. So, if they aim to replace '  
            'their pre-retirement income, saving 10x by 67 is the target.\n'  
            '\n'
```
- Bottom Status Bar:** Shows "Cell 2 of 29", "4°C Cloudy", "ENG US", and the date "4/10/2025".

File Edit Selection View Go Run ...

Bonus Assignment_2

EXPLORER OPEN EDITORS 8 unsaved

- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

Code Markdown Run All Restart Clear All Outputs Jupyter Variables Outline ...

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

```

result3 = run_test_case(
    "What's the best recipe for chocolate chip cookies?",
    3, "Irrelevant question (non-retirement topic)"
)
[9] ✓ 5.9s
...
=====
TEST CASE 3: Irrelevant question (non-retirement topic)
QUESTION: What's the best recipe for chocolate chip cookies?
=====

... Batches: 100% 1/1 [00:00<00:00, 25.79it/s]
...
--- Checking document relevance ---
Document relevance: no
RESULT:
{'input': "What's the best recipe for chocolate chip cookies?", 'output': 'I can only answer questions about retirement planning.', 'status': 'rejected (irrelevant)'}
=====
```

BONUS ASSIGNMENT 2

- > chrome
- > chroma_db
- > venv
- 2.6.1
- Bonus Assignment_2 - In...
- Bonus Assignment_2.pdf
- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment7.ipynb
- Experiment8.ipynb
- Experiment6.ipynb
- FFFGX.pdf
- Fidelity-Langgraph-Agen...

OUTLINE TIMELINE [10]

Cell 2 of 29

File Edit Selection View Go Run ...

Bonus Assignment_2

EXPLORER OPEN EDITORS 8 unsaved

- Experiment1.ipynb
- Experiment2.ipynb
- Experiment3.ipynb
- Experiment4.ipynb
- Experiment5.ipynb
- Experiment6.ipynb
- Experiment7.ipynb
- Experiment8.ipynb

Code Markdown Run All Restart Clear All Outputs Jupyter Variables Outline ...

ANSWER: The Portfolio Net Assets amount to \$21,466.13 million as of 11/30/2024.

Q4: What is the Portfolio Net Assets?
A4: The Portfolio Net Assets amount to \$21,466.13 million as of 11/30/2024.

```

print("\n== Processing PDF Document ==")
from llama_index.readers.file import PyMuPDFReader
...
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q4: {question}\nA4: {answer}\n")
...
PDF QUESTION 4: What is the Portfolio Net Assets?
... Batches: 100% 1/1 [00:00<00:00, 20.96it/s]
...
ANSWER: The Portfolio Net Assets amount to $21,466.13 million as of 11/30/2024.

Q4: What is the Portfolio Net Assets?
A4: The Portfolio Net Assets amount to $21,466.13 million as of 11/30/2024.

# Question 5
question = "What is the Morningstar rating for this fund?How many funds used to rate this fund?"
num = 5
answer = query_pdf(question, num)
pdf_results.append({"question": question, "answer": answer})
print(f"Q5: {question}\nA5: {answer}\n")
...
PDF QUESTION 5: What is the Morningstar rating for this fund?How many funds used to rate this fund?
... Batches: 100% 1/1 [00:00<00:00, 14.06it/s]
...
Retrying llama_index.llms.openai.base.OpenAI._chat in 1.0 seconds as it raised RateLimitError: Error code: 429 - {'error': {'message': 'Rate limit exceeded'}}
Retrying llama_index.llms.openai.base.OpenAI._chat in 1.6376906880985245 seconds as it raised RateLimitError: Error code: 429 - {'error': {'message': 'Rate limit exceeded'}}
ANSWER: The fund has a Morningstar rating of 3 stars, and it was rated among a total of 188 funds in its category.

Q5: What is the Morningstar rating for this fund?How many funds used to rate this fund?
A5: The fund has a Morningstar rating of 3 stars, and it was rated among a total of 188 funds in its category.
```

OUTLINE TIMELINE [18]

Cell 2 of 29