# **Coding Solution: Gemini Pro Financial Decoder - Feature Explanation**

## Application Overview:-

### Purpose :-

This application leverages the power of Google Gemini Pro to transform raw financial statements into digestible summaries and insightful visualizations. It's designed to simplify the analysis of balance sheets, profit & loss statements, and cash flow statements.

## Seamless File Upload:-

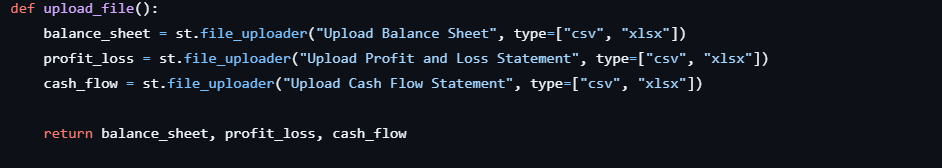
### Feature Description:-

Users can effortlessly upload their financial data files. The system supports both .csv and .xlsx formats, making it versatile for various reporting needs.

### Key Component: st.file\_uploader:-

Streamlit's built-in file uploader ensures a smooth and intuitive experience for selecting the necessary financial documents.

### Code Snippet:-



## Effortless Data Ingestion:-

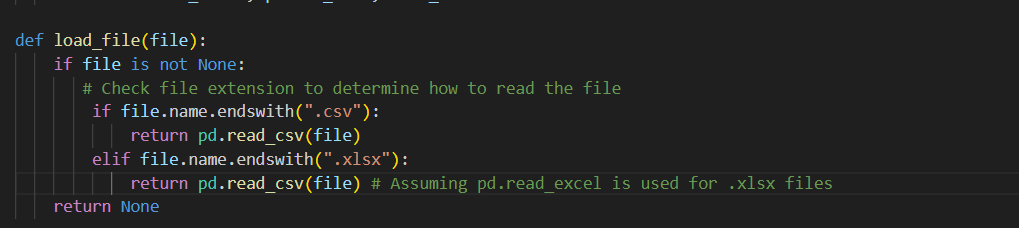
### Feature Description:-

Once uploaded, the files are promptly read and converted into Pandas DataFrames. This standardized format is crucial for subsequent processing, analysis, and interaction with the LLM.

### Core Logic: pd.read\_csv and pd.read\_excel (implicit)

The function intelligently handles both CSV and Excel formats to ensure data is correctly loaded.

### Code Snippet:-



Note: The original load\_file snippet only showed pd.read\_csv. For .xlsx to work, pd.read\_excel would also be needed.

## Intelligent Summarization with Gemini Pro:-

### Feature Description

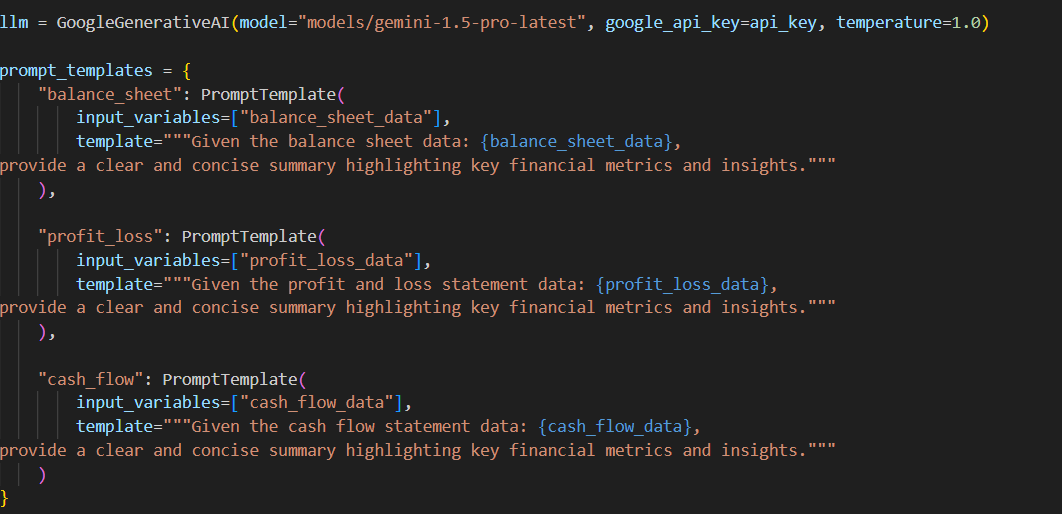
This is the heart of the application, where the Google Gemini Pro model, integrated via LangChain, transforms raw data into insightful summaries. Prompt templates are dynamically constructed to guide the LLM's response, focusing on key financial metrics.

### Key Components:

PromptTemplate: Structures questions for the LLM.

llm(prompt): Invokes the Gemini Pro model to generate content.

### Code Snippet (Prompt Construction):-



**Code Snippet (Summary Generation):-**

## 

## Visualizing Key Metrics:-

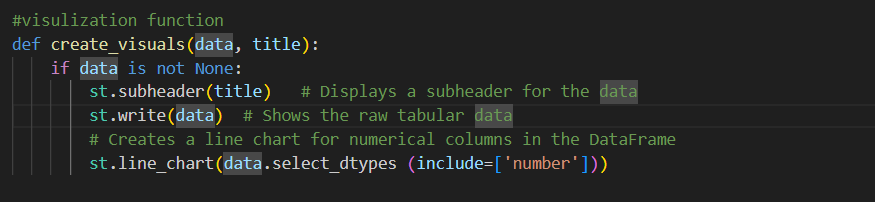
### Feature Description

To complement the LLM-generated summaries, the application provides dynamic data visualizations. Line charts are generated for numerical data, offering a quick visual overview of trends, alongside a tabular display for detailed data inspection.

### Key Component: st.line\_chart & st.write

Streamlit's intuitive functions make it simple to display both charts and raw data.

### Code Snippet:-



## Interactive User Experience (Main UI)

### Feature Description

The Streamlit-powered user interface seamlessly integrates all the backend logic. Users can upload files, trigger report generation with a single click, and then view the comprehensive summaries and interactive visualizations directly within the app.

### Workflow:

Title Display: st.title for clear identification.

File Upload: Calls upload\_file() for input.

Report Generation Button: st.button initiates the processing.

Summary & Visual Display: Results from generate\_summary() and create\_visuals() are presented.

### Code Snippet

