**Diferencias principales:**

1. **Carga de documentos**: En el primer código, se utiliza PyPDFDirectoryLoader para cargar documentos desde un directorio local (./us\_census). En el segundo código, se utiliza WebBaseLoader para cargar documentos desde una URL específica (https://docs.smith.langchain.com/).
2. **Embeddings**: En el primer código, se utiliza OpenAIEmbeddings para generar embeddings de los documentos. En el segundo código, se utiliza OllamaEmbeddings.
3. **Modelo de lenguaje**: En el primer código, se utiliza el modelo Llama3-8b-8192. En el segundo código, se utiliza el modelo mixtral-8x7b-32768.
4. **Carga de vectores**: En el primer código, se crea un vector store utilizando FAISS después de que el usuario hace clic en el botón "Documents Embedding". En el segundo código, se crea el vector store cuando se carga la aplicación.
5. **Estructura del código**: El primer código tiene una estructura más compleja, con una función vector\_embedding que se llama cuando se hace clic en el botón "Documents Embedding". El segundo código tiene una estructura más lineal, con la carga de documentos y la creación del vector store al inicio de la aplicación.

**Conclusión**: Ambos códigos implementan una aplicación de chatbot que utiliza RAG para responder a preguntas basadas en un conjunto de documentos. Sin embargo, hay diferencias en la carga de documentos, embeddings, modelos de lenguaje y estructura del código.