**PROJE:Airflow ve MySQL Kullanarak Otomatik Veri Yükleme ve Birleştirme**

**Proje Amacı**

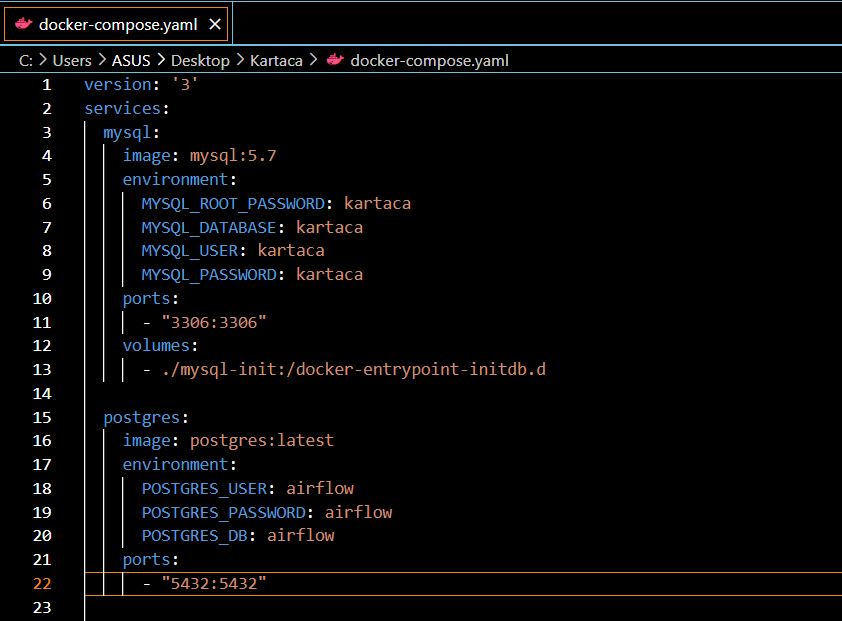
Bu projenin amacı, JSON dosyalarında bulunan ülke ve para birimi verilerini Airflow kullanarak otomatik bir şekilde MySQL veritabanına yüklemek ve bu verileri birleştirerek birleştirilmiş veriler tablosu oluşturmaktır. Proje kapsamında Docker kullanarak Airflow ve MySQL servislere ayağa kaldırılmış, üç farklı DAG (Directed Acyclic Graph) oluşturulmuş ve bu DAG’lar aracılığıyla verilerin yüklenmesi ve birleştirilmesi otomatikleştirilmiştir.

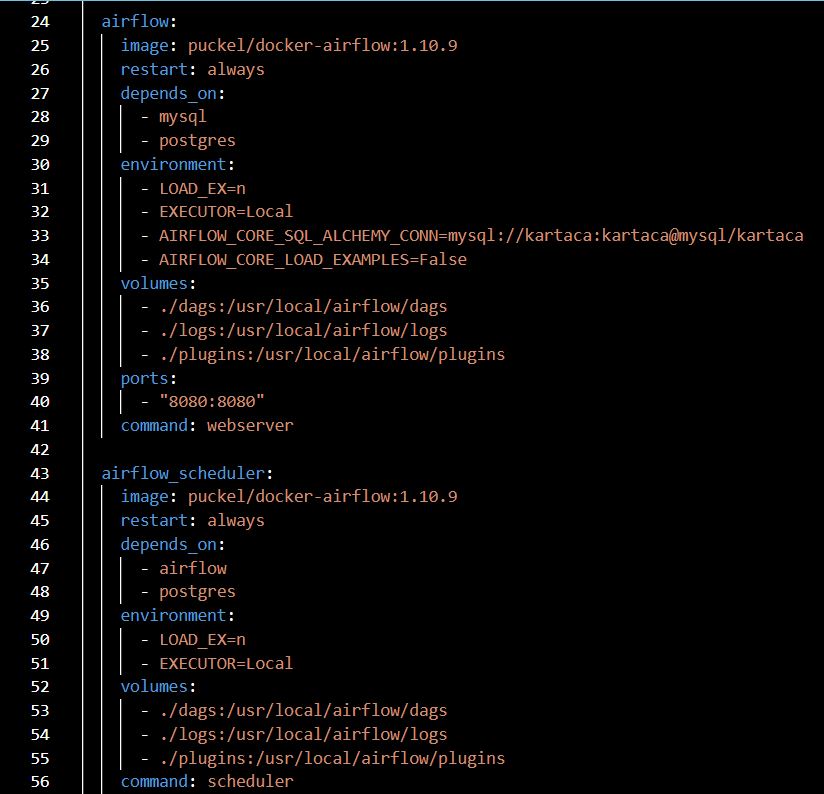
**Gereksinimler**

* Docker
* Docker Compose
* Airflow
* MySQL
* Python 3.9 ve üzeri

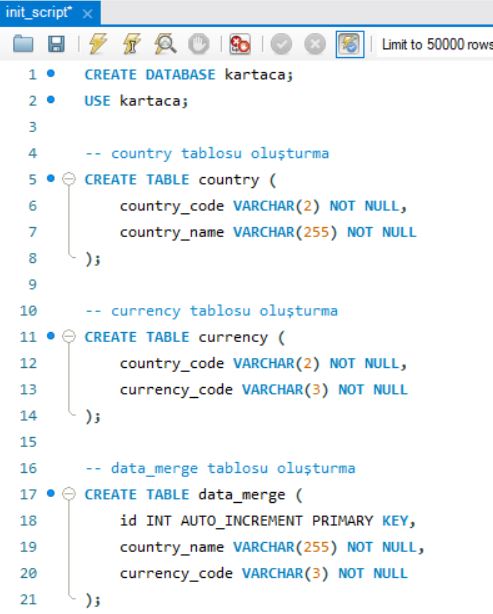
**Kurulum ve Çalıştırma**

1. **Docker ve Docker Compose Kurulumu:** Docker Compose kullanılarak Airflow ve MySQL servislerini ayağa kaldırmak için docker-compose.yaml dosyası oluşturulmuş ve servisler başlatılmıştır.



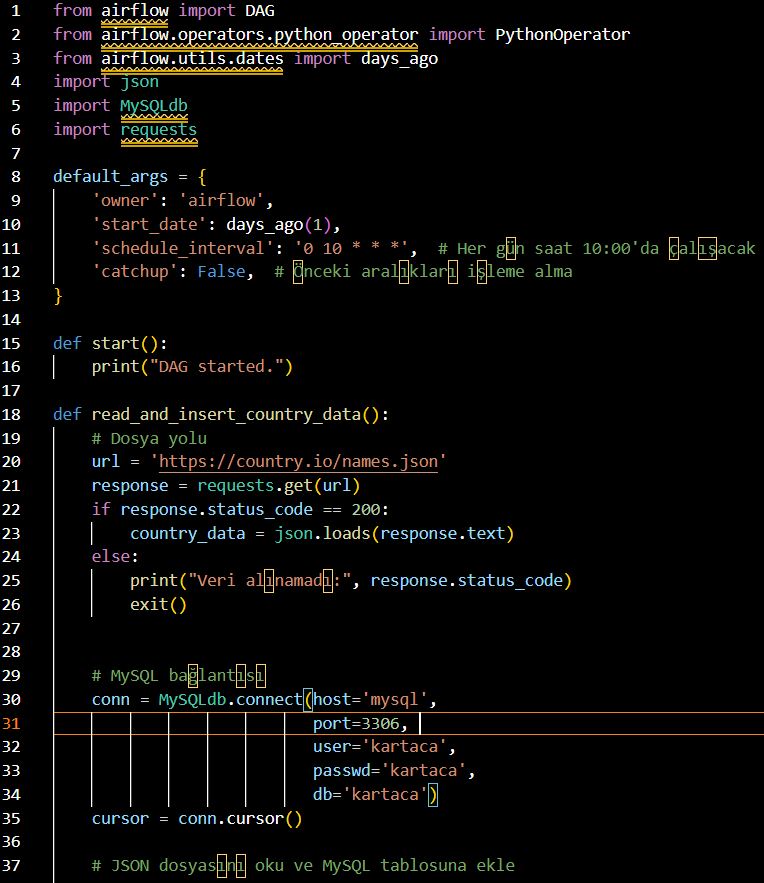


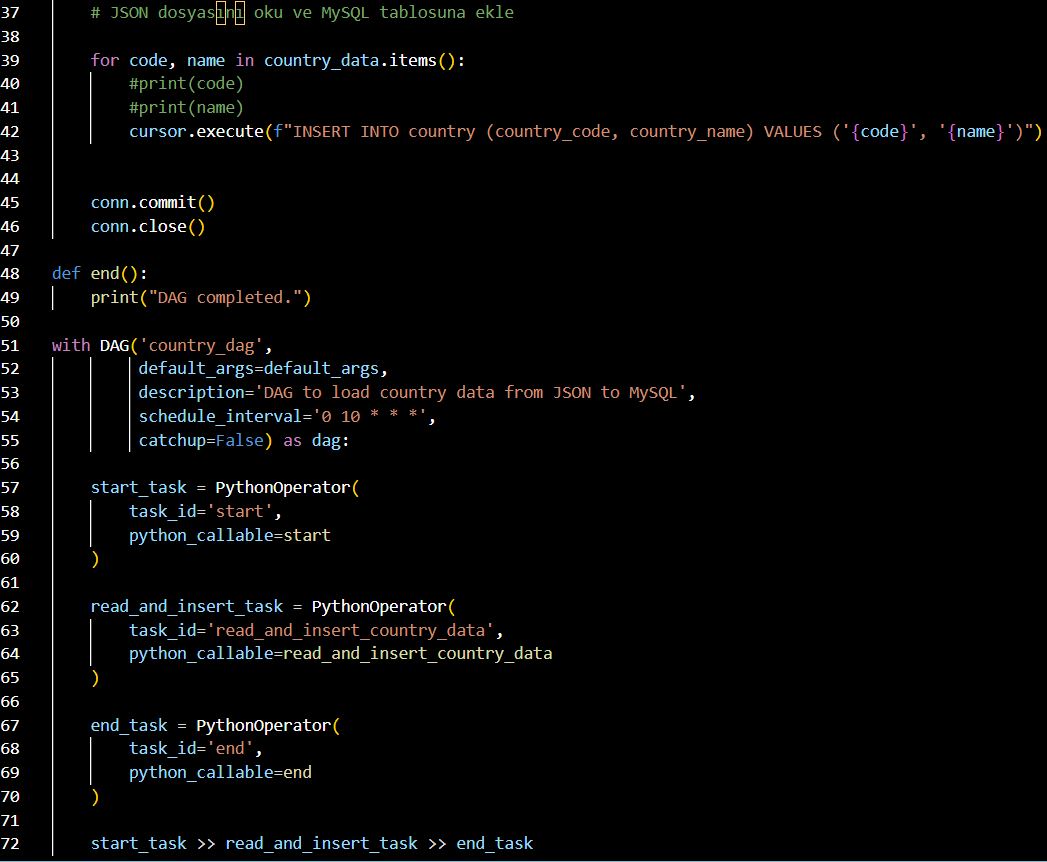
1. **MySQL Tablo Yapısı:** MySQL init script ile tablolar boş olacak şekilde şemalar tanımlanmıştır.



1. **Airflow DAG’ları Oluşturma:**Her sabah belirtilen saatte çalışan ve json dosyalarındaki veriyi okuyup tabloya yazan DAG’lar oluşturulmuştur.

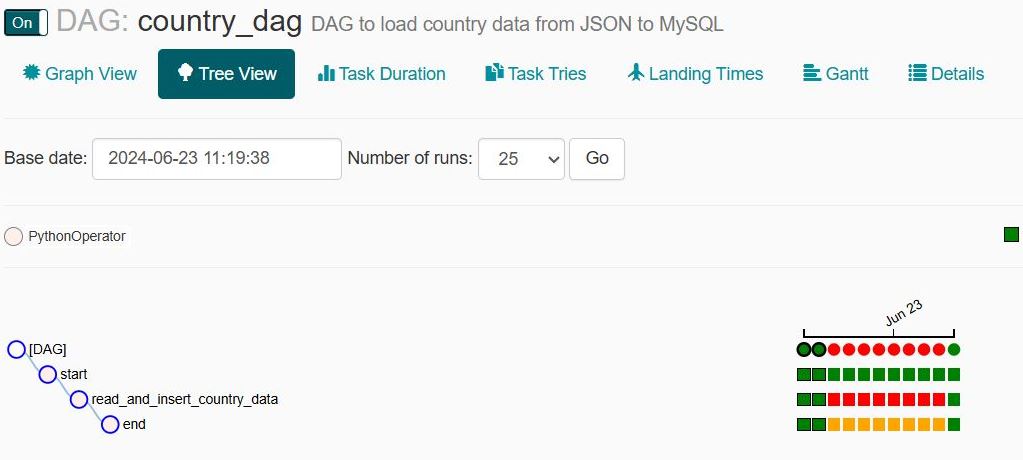
Örnek country\_dag.py:

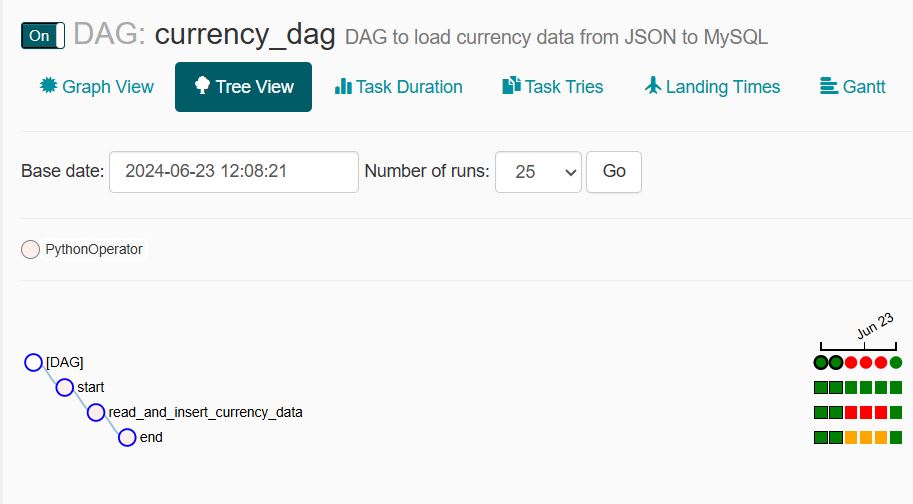


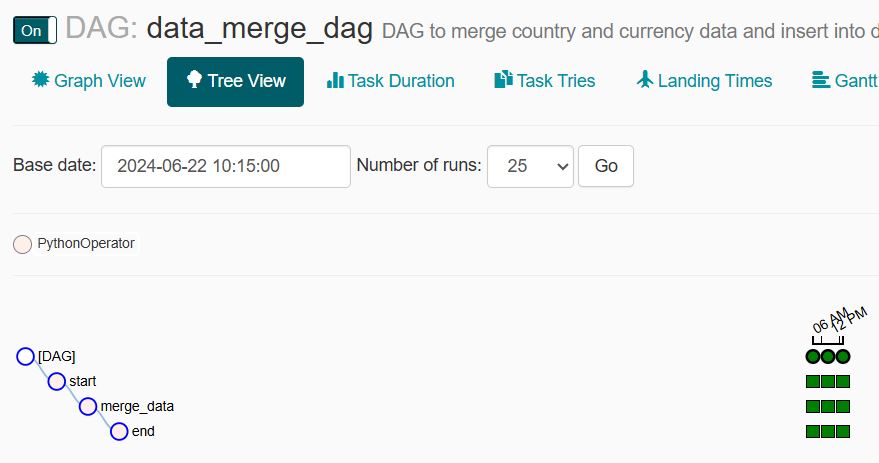


**Çıktılar**

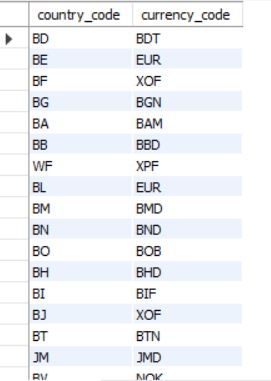
* **Airflow DAG’leri:** Verilerin düzenli çekilmesi, MySQL’e eklenmesi ve birleştirilmesi için üç farklı DAG oluşturulmuş ve başarıyla çalıştırılmıştır.
  + ‘country\_dag’: Ülke verilerini ‘country’ tablosuna ekler.
  + ‘currency\_dag’: Para birimi verilerini ‘currency’ tablosuna ekler.
  + ‘data\_merge\_dag’: Ülke ve para birimi verilerini birleştirerek ‘data\_merge’ tablosuna ekler.



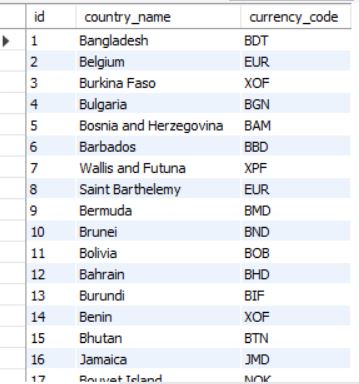




* **MySQL veritabanı:** ‘country’, ‘currency’ ve ‘data\_merge’ tabloları oluşturulmuş ve veriler bu tablolara başarıyla eklenmiştir.



Country tablosu Currency tablosu



Data\_merge tablosu

**Sonuç**

Bu proje, JSON dosyalarındaki verilerin MySQL veritabanına yüklenmesi ve birleştirilmesi sürecini otomatikleştirmiştir. Airflow kullanımı, bu süreçlerin düzenli ve otomatik bir şekilde yönetilmesini sağlamıştır. Docker ile servislerin kolayca ayağa kaldırılması ve taşınabilirliği artırılmıştır. Projede karşılaşılan hatalar ve istisnalar uygun şekilde ele alınarak, verilerin güvenilirliği ve doğruluğu sağlanmıştır.