Assignment 1: Docker Containers

Commands Used
docker compose up --build

docker compose down -v

Stdout JSON (printed by the app)
== Summary ==


```
{
 "total_trips": 6,
 "avg_fare_by_city": [
   {
    "city": "Charlotte",
    "avg_fare": 16.25
   },
   {
    "city": "New York",
    "avg_fare": 19.0
   },
   {
    "city": "San Francisco",
    "avg_fare": 20.25
   }
 ],
 "top_by_minutes": [
   {
    "id": 6,
    "city": "San Francisco",
    "minutes": 28,
    "fare": 29.3
   },
   {
    "id": 4,
    "city": "New York",
    "minutes": 26,
    "fare": 27.1
   },
   {
```

```
    "id": 2,
    "city": "Charlotte",
    "minutes": 21,
    "fare": 20.0
   },
   {
    "id": 1,
    "city": "Charlotte",
    "minutes": 12,
    "fare": 12.5
   },
   {
    "id": 5,
    "city": "San Francisco",
    "minutes": 11,
    "fare": 11.2
   },
   {
    "id": 3,
    "city": "New York",
    "minutes": 9,
    "fare": 10.9
   }
  ]
}
```


Contents of out/summary.json


```
{
  "total_trips": 6,
  "avg_fare_by_city": [
   {
    "city": "Charlotte",
    "avg_fare": 16.25
   },
   {
    "city": "New York",
    "avg_fare": 19.0
   },
   {
    "city": "San Francisco",
    "avg_fare": 20.25
```

```
  }
 ],
 "top_by_minutes": [
  {
   "id": 6,
   "city": "San Francisco",
   "minutes": 28,
   "fare": 29.3
  },
  {
   "id": 4,
   "city": "New York",
   "minutes": 26,
   "fare": 27.1
  },
  {
   "id": 2,
   "city": "Charlotte",
   "minutes": 21,
   "fare": 20.0
  },
  {
   "id": 1,
   "city": "Charlotte",
   "minutes": 12,
   "fare": 12.5
  },
  {
   "id": 5,
   "city": "San Francisco",
   "minutes": 11,
   "fare": 11.2
  },
  {
   "id": 3,
   "city": "New York",
   "minutes": 9,
   "fare": 10.9
  }
```

```
  ]
}
```

Reflection

This assignment taught me how to use Docker Compose to manage a multi-container setup. I learned how to seed a PostgreSQL database automatically, connect a Python app to the database using environment variables, and ensure services are orchestrated correctly with health checks. I also practiced capturing outputs both in the console and in JSON files. The biggest takeaway was seeing how reproducible workflows simplify development and testing. In future work, I could expand this project with additional queries, more complex schemas, or integration with other services.