
🎯 Theme: Online Course Learning Platform Analytics

📁 Dataset: `course_enrollments.csv`

```
EnrollID,UserID,CourseID,CourseName,Category,EnrollDate,CompletionDate,ProgressPercent,R

E001,U001,C001,Python Basics,Programming,2024-04-01,2024-04-10,100,4
E002,U002,C002,Excel for Finance,Productivity,2024-04-02,,45,
E003,U001,C003,ML with PySpark,Data Science,2024-04-03,,30,
E004,U003,C001,Python Basics,Programming,2024-04-04,2024-04-20,100,5
E005,U004,C004,Digital Marketing,Marketing,2024-04-05,2024-04-16,100,4
```

1️⃣ Ingestion & Time Fields

- Load into PySpark with inferred schema
- Convert EnrollDate and CompletionDate to `date` type
- Add `DaysToComplete` column if completed

2️⃣ User Learning Path Progress

- Group by `UserID`: count of courses enrolled
- Avg progress % across all enrollments
- Flag `IsCompleted = ProgressPercent = 100`

3️⃣ Engagement Scoring

- Create a score: `ProgressPercent * Rating` (if not null)
- Replace null `Rating` with 0 before computing

4️⃣ Identify Drop-offs

- Filter all records with `ProgressPercent < 50` and `CompletionDate is null`
- Create a view called `Dropouts`

5️⃣ Joins with Metadata

Create `course_catalog.csv`:

```
CourseID,Instructor,DurationHours,Level
C001,Abdullah Khan,8,Beginner
C002,Sana Gupta,5,Beginner
C003,Ibrahim Khan,10,Intermediate
C004,Zoya Sheikh,6,Beginner
```

- Join to find `average progress per instructor`
- Show who teaches the most enrolled course

6️⃣ Delta Lake Practice

- Save as Delta Table `enrollments_delta`

- Apply:
  - Update: Set all ratings to 5 where Course = 'Python Basics'
  - Delete: All rows where ProgressPercent = 0
- Show `DESCRIBE HISTORY`

---

7 Window Functions

- Use `dense_rank()` to rank courses by number of enrollments
- `lead()` to find next course by each user (sorted by EnrollDate)

---

8 SQL Logic for Dashboard Views

- Create views:
  - `daily_enrollments`
  - `category_performance` (avg rating by category)
  - `top_3_courses`

---

9 Time Travel

- View previous version before update/delete
- Use `VERSION AS OF` and `TIMESTAMP AS OF`

Export Reporting

- Write to JSON, partitioned by `Category`
- Create summary DataFrame:
  - CourseName, TotalEnrollments, AvgRating, AvgProgress
- Save as Parquet
