New LLD: Insurance Claim Processing Analysis

Domain: Insurance & Risk Analytics
Difficulty Level: Medium–High | Total Marks: 20
Concepts Tested:

- DataFrame creation and column typing
- Grouping and aggregation
- Filtering based on conditions
- Multi-condition column creation
- Summarizing by multiple fields
- Cleaning corrupted rows

Problem Statement

An insurance company stores daily claim processing logs. Each entry contains:

- Customer ID
- Claim Category (Health, Auto, Home, Life)
- Claim Amount
- Claim Status (Approved, Rejected, Pending)
- Claim Date

You must analyze this data to monitor trends, compute approval rates, flag expensive claims, and clean inconsistencies.

Functions to Implement

---

1. create_claims_df(claim_data: list) -> pd.DataFrame

Input: List of lists
Output: DataFrame with 5 columns: CustomerID, Category, Amount, Status, Date

---

2. approval_rate_by_category(df: pd.DataFrame) -> pd.DataFrame

Logic:

- Group by Category
- Count total + approved claims
- Compute Approval Rate = Approved / Total × 100 (rounded to 1 decimal)

- Output: DataFrame with columns: Category, Approval Rate

---

3. add_flag_high_amount(df: pd.DataFrame, threshold: float) -> pd.DataFrame

Logic:

- Add new column IsHighValue = 1 if Amount > threshold else 0

---

4. get_top_pending_claims(df: pd.DataFrame, n: int) -> pd.DataFrame

Logic:

- Filter Status == "Pending"
- Return top n claims sorted by Amount descending

---

5. claim_summary_by_status(df: pd.DataFrame) -> pd.DataFrame

Logic:

- Group by Status
- Aggregate Amount using sum, min, max, avg

---

6. clean_invalid_claims(df: pd.DataFrame) -> pd.DataFrame

Logic:

- Remove rows with:
    - Status not in [Approved, Rejected, Pending]
    - Amount is None or negative

📄 Sample Input for TC1

```
create_claims_df([
  [1001, "Health", 5000.0, "Approved", "2024-05-01"],
  [1002, "Auto", 2000.0, "Rejected", "2024-05-01"]
])
```

Expected Output:

CustomerID	**Category**	**Amount**	**Status**	**Date**
1001	Health	5000.0	Approved	2024-05-01
1002	Auto	2000.0	Rejected	2024-05-01

Test Case Matrix

TC ID	**Description**	**Function Name**	**Marks**
TC1	Create structured claims DataFrame	create_claims_df()	2.5
TC2	Approval rate by category	approval_rate_by_category()	2.5
TC3	Flag high-value claims	add_flag_high_amount()	2.5
TC4	Get top N pending claims	get_top_pending_claims()	2.5
TC5	Claim summary by status	claim_summary_by_status()	2.5
HTC1	Remove invalid claims	clean_invalid_claims()	2.5
HTC2	Handle missing/negative amounts	clean_invalid_claims()	2.5
HTC3	Tie in top pending claim amounts	get_top_pending_claims()	2.5