🔍 What is Data Skewness?

In distributed systems like Apache Spark, data is processed in parallel across multiple nodes. Ideally, each node handles an equal "chunk" of data. **Data Skew** happens when the distribution is asymmetrical—one partition ends up with 80% of the data while others sit idle.

The Problem Statement (Scenario)

Imagine you are calculating the **sum(sales)** for a global organization across **5 countries**, grouped by **region**.

- **The Skew:** One specific region (e.g., "North America - Retail") has massive business volume, generating 100 million rows, while other regions only have 1 million each.
- **The Bottleneck:** When you run a groupBy("region").sum("sales"), Spark sends all 100M rows for that one region to a **single task**.
- **Result:** 4 executors finish in seconds, but the 5th executor runs for 2 hours, eventually failing with an **Out of Memory (OOM)** error.

💡 The Solution: Salting + 2-Level Aggregation

Since we can't split a single key ("Region") into multiple partitions naturally, we "salt" it to force a redistribution.

Level 1 Aggregation: Add a random "salt" to the region names to break the massive group into smaller, manageable sub-groups.
Level 2 Aggregation: Remove the salt and aggregate the sub-totals into the final result.

💻 Solution (Code)

```
from pyspark.sql import functions as F


# 1. Add a random salt (e.g., 0-19) to create 20 sub-groups per region

df_salted = df.withColumn("salt", (F.rand() * 20).cast("int")) \

        .withColumn("salted_region", F.concat(F.col("region"), F.lit("_"), F.col("salt")))


# 2. Level 1: Aggregate by the salted key (Parallel processing)

intermediate_df = df_salted.groupBy("salted_region") \

                .agg(F.sum("sales").alias("partial_sales"))
```

```
# 3. Level 2: Remove salt and aggregate for final result
final_df = intermediate_df.withColumn("original_region", F.split(F.col("salted_region"), "_")[0]) \
                .groupBy("original_region") \
                .agg(F.sum("partial_sales").alias("total_sales"))
```

✅ **Key Takeaway**

Salting turns a single "hot" partition into many smaller ones, ensuring all your CPU cores are working equally hard.