Data Engineer Cheat Sheet (SQL + Python + PySpark + Spark)

Ingestion Layer

```
SQL:
SELECT * FROM Sales.Orders WHERE OrderDate >= GETDATE()-1;
Python:
import pandas as pd
df = pd.read_csv('orders.csv')
PySpark:
df = spark.read.option('header', True).csv('.../bronze/orders.csv')
Spark Streaming:
stream_df = spark.readStream.format('eventhubs').option('eventhubs.connectionString','<conn>').load()
```

Transformation Layer

```
SQL:
SELECT c.CustomerName, o.OrderID FROM Customers c JOIN Orders o ON c.CustomerID=o.CustomerID;
Python:
df_filtered = df[df['amount'] > 1000]
df.groupby('customer_id')['amount'].sum()
PySpark:
df_filtered = df.filter(df['amount'] > 1000)
df_group = df.groupBy('customer_id').agg(F.sum('amount'))
Spark SQL:
spark.sql('SELECT customer_id, SUM(amount) FROM orders GROUP BY customer_id')
```

Optimization

```
Repartition:
df = df.repartition(10, 'customer_id')
Broadcast Join:
df_joined = big_df.join(broadcast(small_df), 'id')
Cache:
df.cache()
ZORDER:
df.write.format('delta').option('optimizeWrite','true').save(path)
```

Write Back

```
SQL:
INSERT INTO Sales.Summary SELECT CustomerID, SUM(Amount) FROM Sales.Orders GROUP BY CustomerID;
Python:
df.to_csv('gold/orders_summary.csv', index=False)
PySpark:
df.write.format('delta').mode('append').save('/mnt/datalake/gold/orders_summary/')
```