Data Analyst Training

Power BI, Data Warehousing & Interview Success

A Comprehensive Guide for Recent Graduates

Data Warehousing

Power BI

Interview Prep

Duration: 1 Hour Training Session
Target: Entry-Level Data Analyst Roles

Session Overview

What You'll Learn Today

📚 Data Warehousing (15 minutes)

- **What is a Data Warehouse?** - Understanding the foundation of business intelligence
- **ETL Pipeline Process** - How data flows from sources to warehouse
- **Data Modeling** - Star schema and dimensional modeling
- **Real-world Applications** - Practical use cases across industries

📊 Power BI Essentials (20 minutes)

- **Power BI Components** - Desktop, Service, and Mobile
- **Data Import & Transformation** - Power Query fundamentals
- **DAX Formulas** - Creating calculations and measures
- **Building Dashboards** - Visualizations and interactivity

💼 Interview Success (15 minutes)

- **Resume & Portfolio Tips** - Standing out as a new graduate
- **Technical Questions** - Common SQL, Power BI, and DW questions
- **Behavioral Strategies** - STAR method and storytelling

- **Mock Interview Practice** - Real scenarios and responses

🎯 Session Goals

By the end of this training, you will:

- Understand how data warehouses organize data for analytics
- Be able to create basic Power BI dashboards from scratch
- Know how to answer common data analyst interview questions
- Have 13 hands-on exercises to build your portfolio

Database vs Data Warehouse

Understanding OLTP vs OLAP

🗄️ OLTP (Database)

Online Transaction Processing

Purpose: Handles day-to-day operational transactions in real-time. Optimized for writing data quickly.

Example Uses:

- Processing online orders
- ATM withdrawals
- Booking airline tickets
- Hospital patient registration

Structure: Highly normalized (3NF) to minimize data redundancy and ensure data integrity. Many small tables with relationships.

Operations: INSERT, UPDATE, DELETE - Focus on writing and modifying records

Performance: Milliseconds response time, handles thousands of concurrent users

📊 OLAP (Data Warehouse)

Online Analytical Processing

Purpose: Stores historical data for complex analysis and reporting. Optimized for reading and querying data.

Example Uses:

- Yearly sales trend analysis
- Customer segmentation studies
- Financial forecasting
- Market basket analysis

Structure: Denormalized (star/snowflake schema) for fast query performance. Fewer large tables with redundant data.

Operations: SELECT with aggregations (SUM, AVG, COUNT), GROUP BY - Focus on reading and analyzing

Performance: Seconds to minutes for complex queries across years of data

📌 Key Difference

OLTP handles live business operations (the "now"), while **OLAP** analyzes historical data for insights (the "then"). Data flows from OLTP databases → ETL process → OLAP warehouse.
As a Data Analyst: You'll primarily work with OLAP systems to query and visualize business trends.

Real-World Examples

OLTP Example: E-commerce Order

Scenario: Customer adds item to cart
Action: Database immediately saves OrderID, CustomerID, ProductID, Quantity, Timestamp
Focus: Fast write, maintain inventory accuracy, process payment

OLAP Example: Sales Analysis

Scenario: Marketing asks "Which products sold best in Q4 2024?"
Action: Query DW aggregating millions of historical orders by product, region, time
Focus: Fast read, complex analytics, trend identification

The ETL Pipeline

How Data Flows into the Warehouse

> **ETL (Extract, Transform, Load)** is the process of moving data from multiple sources into a centralized data warehouse. This is the backbone of any analytics infrastructure, ensuring clean, consistent data is available for analysis.

ETL Process Flow

1️⃣

EXTRACT

Pull data from various source systems into a staging area.

Common Sources:

- **Databases:** MySQL, PostgreSQL, Oracle
- **Files:** CSV, Excel, JSON, XML
- **APIs:** Shopify, Salesforce, REST

2️⃣

TRANSFORM

Clean, standardize, and prepare data for analysis.

Transformation Steps:

- **Data Cleaning:** Remove duplicates, fix typos
- **Standardization:** "USA" → "United States"
- **Handle Nulls:** Fill, flag, or remove

3️⃣

LOAD

Insert transformed data into the target data warehouse.

Load Strategies:

- **Full Load:** Replace entire dataset (simple but slow)
- **Incremental:** Load only new/changed data (efficient)
- **Delta Load:** Track changes

APIs

- **Web Data:** Web scraping, logs
- **Streaming:** Kafka, real-time feeds

Challenge: Different formats, data types, update frequencies, access permissions

- **Type Conversion:** String to Date/Number
- **Business Logic:** Calculate profit, discounts
- **Join/Merge:** Combine multiple sources

Tools: Power Query, Python Pandas, Talend, Apache NiFi

over time (SCD)

- **Batch:** Scheduled runs (nightly, hourly)
- **Real-time:** Continuous streaming

Scheduling: Apache Airflow, Azure Data Factory, cron jobs

🏪 Complete Retail ETL Example

Extract: Pull sales from POS system database (MySQL), online orders from API (Shopify), inventory from Excel files

Transform:

- Standardize date formats (MM/DD/YYYY)
- Convert currency to USD
- Calculate Profit = Revenue - Cost
- Remove cancelled orders
- Join sales with product catalog

Load: Insert into Sales fact table in Snowflake DW, update dimension tables, run at 2 AM daily

🔄 ELT: Modern Alternative

ELT (Extract, Load, Transform) loads raw data first, then transforms within the powerful data warehouse.

Why ELT?

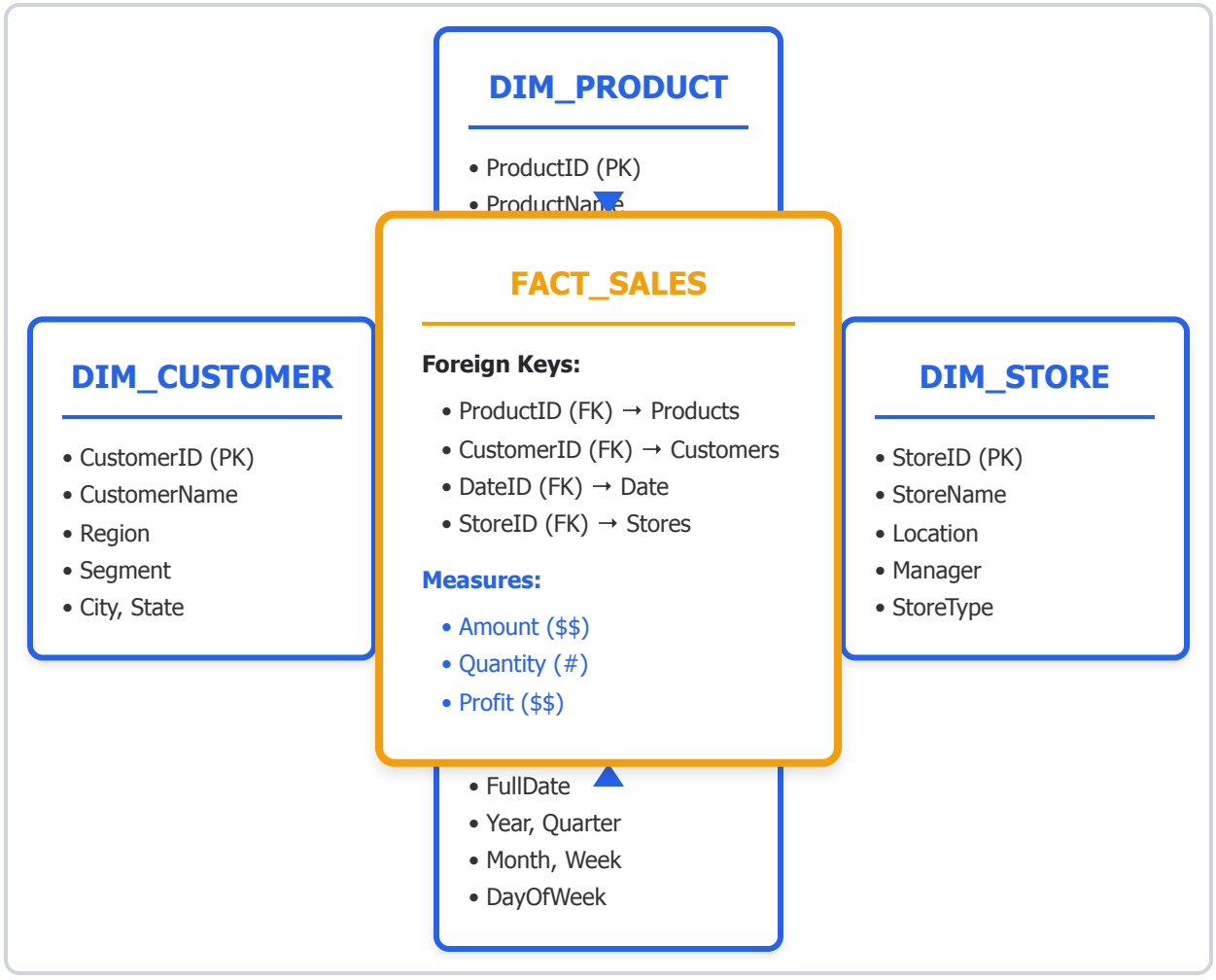
- Cloud DWs (BigQuery, Snowflake) have massive compute power
- Cheaper to store raw data than transform externally
- Flexibility to re-transform without re-extracting
- Faster initial data availability

Tools: dbt (data build tool), Fivetran, Stitch

Star Schema: Data Modeling

The Foundation of Analytics

> A **star schema** organizes data with a central **Fact Table** containing measurable business metrics (sales, revenue, quantity), surrounded by **Dimension Tables** that provide context (who, what, when, where). This denormalized structure optimizes query performance for business intelligence and reporting.



✓ Key Advantages

- **Fast queries:** Fewer JOINs needed
- **Simple structure:** Easy to understand
- **BI optimized:** Works great with Power BI
- **Flexible:** Easy to add dimensions

Example Business Question

"What are total sales by region for Q4 2024?"

```
SELECT
c.Region,
SUM(f.Amount)
as TotalSales
FROM FACT_SALES
f JOIN
DIM_CUSTOMER c
ON f.CustomerID
= c.CustomerID
JOIN DIM_DATE d
ON f.DateID =
d.DateID WHERE
d.Quarter =
'Q4' AND d.Year
= 2024 GROUP BY
c.Region;
```

Real-World Use Cases

- **Retail:** Sales analysis by product, store, time
- **Healthcare:** Patient visits, treatments, billing
- **Finance:** Transaction analysis, customer behavior
- **E-commerce:** Order tracking, customer segments

Power BI Workflow

From Data to Dashboard in 4 Steps

Power BI is Microsoft's leading business intelligence tool, ideal for beginners due to its drag-and-drop interface. It connects to data warehouses, transforms data, and builds interactive dashboards to visualize insights.
Download: powerbi.microsoft.com (Desktop version is free)

1️⃣

IMPORT

- Connect to sources
- Excel, SQL, APIs
- 100+ connectors
- Import vs Direct Query

2️⃣

TRANSFORM

- Power Query editor
- Clean data
- Merge tables
- Apply filters

3️⃣

MODEL

- Create relationships
- Write DAX formulas
- Build hierarchies
- Calculated columns

4️⃣

VISUALIZE

- Build charts
- Add slicers
- Create dashboards
- Publish & share

Power Query Transformations

Common Operations:

Example Workflow:

- Filter rows (remove nulls)
- Split columns (First/Last name)
- Change data types
- Remove duplicates
- Merge queries (JOIN tables)
- Pivot/Unpivot data

1. Remove rows where Sales < $10
2. Split "Full Name" → First, Last
3. Change "Date" to Date type
4. Merge with Customer table
5. Group by Region, sum Sales

Power BI Components

📊 Charts & Visuals

- Bar charts (categories)
- Line charts (trends)
- Pie charts (proportions)
- Maps (geography)
- Cards (KPIs)
- Tables & matrices

🎛️ Interactivity

- Slicers (filters)
- Drill-downs
- Tooltips
- Cross-filtering
- Bookmarks
- Buttons & navigation

🧱 DAX Formulas

- Measures (calculations)
- Calculated columns
- Time intelligence
- Aggregations
- Filtering context
- Advanced analytics

DAX: Data Analysis Expressions

Power BI's Formula Language

DAX (Data Analysis Expressions) is Excel-like formula language for creating calculations in Power BI. Used for measures, calculated columns, and advanced analytics. Think of it as Excel formulas on steroids!

Common DAX Functions

Basic Aggregation

```
TotalRevenue =
SUM(Sales[Amount])
```

Sums all values in the Amount column. Most basic and commonly used DAX function.

Calculated Column

```
Profit = Sales[Revenue] -
Sales[Cost]
```

Row-by-row calculation. Creates a new column in your table with the calculated value.

Time Intelligence

```
YoY Growth = DIVIDE(
[TotalRevenue] -
CALCULATE([TotalRevenue],
SAMEPERIODLASTYEAR()),
[TotalRevenue] )
```

Compares current period to same period last year. Essential for trend analysis.

Filtering with CALCULATE

```
WestSales = CALCULATE(
SUM(Sales[Amount]),
Sales[Region] = "West" )
```

Filters data to specific region. CALCULATE changes filter context.

Average with Conditions

```
AvgHighValue = AVERAGEX(
FILTER(Sales, Sales[Amount]
> 1000), Sales[Amount] )
```

Calculates average only for sales over $1,000.

Count Distinct

```
UniqueCustomers =
DISTINCTCOUNT(
Sales[CustomerID] )
```

Counts unique customers, ignoring duplicates.

💡 DAX Best Practices

- **Use measures for aggregations** (not calculated columns) - Better performance
- **Start simple, then add complexity** - Build incrementally and test
- **Name clearly** - Use descriptive names like "TotalRevenue" vs "Revenue1"
- **Use CALCULATE for context changes** - Most powerful DAX function
- **Test with small datasets first** - Easier to debug issues
- **Leverage IntelliSense** - Power BI suggests functions as you type
- **Comment complex formulas** - Use // for single-line comments

🎓 Learning Path

1. **Week 1:** Master SUM, AVERAGE, COUNT, DIVIDE
2. **Week 2:** Learn CALCULATE and basic filtering
3. **Week 3:** Time intelligence functions (SAMEPERIODLASTYEAR, DATESYTD)

4. **Week 4:** Advanced functions (FILTER, ALL, RELATED)

Interview Success Strategy

Standing Out as a New Graduate

> With no experience, highlight **transferable skills** from your Master's degree (thesis analysis, stats coursework, research projects). Entry-level interviews test **fundamentals, problem-solving, and enthusiasm**. Companies want to see potential and willingness to learn.

Interview Process (4 Stages)

1️⃣

Phone Screen

Duration: 15-30 min
Focus: Motivation & basics
Question: "Why data analyst?"

2️⃣

Technical

Duration: 45-60 min
Focus: SQL, Power BI
May include: Live coding

3️⃣

Behavioral

Duration: 30-45 min
Focus: STAR method
Stories: Past projects

4️⃣

Case Study

Duration: 1-2 hours
Focus: Real dataset
Deliverable: Analysis report

The STAR Method (Behavioral Questions)

> **STAR** is a structured way to answer behavioral questions by telling a complete story.

Situation

Set the context. Describe the scenario you were in.

Example: "In my thesis project..."

Task

Explain the challenge or goal you faced.

Example: "I needed to analyze 10K survey responses..."

Action

Describe what YOU did specifically (not "we").

Example: "I cleaned data in Excel and validated with SQL..."

Result

Share the outcome. Quantify when possible!

Example: "...which improved accuracy by 15%"

Complete STAR Example

Question: "Tell me about a time you dealt with complex data."

Answer:
Situation: "In my Master's thesis, I received a dataset of 10,000 customer survey responses with many inconsistencies."
Task: "I needed to prepare this data for statistical analysis to identify customer satisfaction trends."
Action: "I first cleaned the data in Excel by removing duplicates and standardizing response formats. Then I wrote SQL queries to validate data integrity and identify outliers. Finally, I used Python to perform correlation analysis."
Result: "This process improved data accuracy by 15% and allowed me to deliver reliable insights that informed my department's customer retention strategy. My professor praised the thorough approach and methodology."

Pre-Interview Preparation

📝 Resume Tips

💼 Portfolio Building

- **Structure:** Education → Projects → Skills → Certifications
- **Quantify:** "Cleaned 10K+ dataset, improving accuracy by 15%"
- **Projects:** Include Power BI dashboards from exercises
- **Keywords:** Match job posting (data visualization, ETL, SQL)
- **Action verbs:** Analyzed, Developed, Created, Optimized

- **GitHub:** Upload .pbix files and SQL scripts
- **Screenshots:** Dashboard images with descriptions
- **README:** Explain each project and insights
- **Power BI Service:** Publish interactive reports (free tier)
- **LinkedIn:** Share project links in experience section

❌ Common Mistakes to Avoid

- Saying "I don't know" without trying to reason through it
- Being too vague ("I worked with data" - be specific!)
- Not asking clarifying questions about the problem
- Forgetting to quantify results in STAR responses
- Not preparing questions to ask the interviewer
- Arriving late or having technical issues (test beforehand)

✅ What Interviewers Want

- **Problem-solving approach:** How you think through issues
- **Clear communication:** Explain technical concepts simply
- **Genuine enthusiasm:** Passion for data and insights
- **Willingness to learn:** Admit gaps and show eagerness
- **Cultural fit:** Teamwork and collaboration skills
- **Practical skills:** Can you actually do the work?

Technical Interview Questions

What to Expect & How to Answer

Data Warehousing Questions

Q1: "Explain the ETL process"

Answer: "ETL stands for Extract, Transform, Load. First, we **Extract** data from various sources like databases, APIs, or files. Then we **Transform** it by cleaning, standardizing formats, handling missing values, and applying business logic. Finally, we **Load** the transformed data into a data warehouse. For example, in a retail context, I would extract sales data from a POS system, transform it by standardizing date formats and calculating profit margins, then load it into a Snowflake data warehouse where analysts can query it."

Q2: "What's the difference between Star and Snowflake schemas?"

Answer: "Both are dimensional modeling approaches, but they differ in normalization. A **Star schema** has a central fact table surrounded by denormalized dimension tables - it's simpler and faster for queries, which makes it ideal for BI tools. A **Snowflake schema** normalizes dimension tables into multiple related tables, which saves storage space but requires more joins and can be slower. For most analytics use cases, I'd recommend a star schema for its simplicity and performance."

Q3: "What is a fact table vs dimension table?"

Answer: "A **fact table** contains quantitative business metrics like sales amount, quantity, or profit - the 'what happened' data. A **dimension table** provides descriptive context like customer name, product category, or date - the 'who, what, when, where' information. Fact tables typically have foreign keys linking to dimension tables and numeric measures we can aggregate. For example, a sales fact table would link to customer, product, date, and store dimension tables."

Power BI Questions

Q1: "How do you create relationships between tables in Power BI?"

Answer: "In Power BI Desktop, I go to the Model View where I can see all tables. I create relationships by dragging a common field from one table to another - for example, dragging ProductID from the Sales table to the Product table. I set the cardinality, typically one-to-many, where the dimension table has unique values and the fact table has many rows. I also enable cross-filtering as needed and ensure the relationship is active. It's important to check that the data types match between related fields."

Q2: "Write a DAX measure for a 3-month rolling average"

Answer:

```
ThreeMonthAvg = AVERAGEX( DATESINPERIOD( Date[Date], MAX(Date[Date]),
-3, MONTH ), [TotalSales] )
```

"This measure uses DATESINPERIOD to get the last 3 months from the latest date in context, then AVERAGEX calculates the average of TotalSales across those months. This is useful for smoothing out monthly fluctuations to see underlying trends."

Q3: "Explain Power Query and when you'd use it"

Answer: "Power Query is Power BI's ETL tool for data transformation before loading into the model. I use it to clean data by removing nulls, split columns like full names into first and last, change data types, merge tables similar to SQL joins, and filter rows. For example, if I import a CSV with inconsistent date formats, I'd use Power Query to standardize them to YYYY-MM-DD before loading. It's more user-friendly than writing code and creates repeatable transformation steps."

SQL Questions

Q1: "Write a query to find the top 5 regions by sales"

```
SELECT TOP 5 Region, SUM(Amount) as TotalSales FROM Sales GROUP BY
Region ORDER BY TotalSales DESC;
```

Explanation: "This query groups sales by region, sums the amounts for each region, then orders them in descending order and limits to the top 5 results. This is a common pattern for ranking analysis."

Q2: "How do you join sales data with customer information?"

```
SELECT c.CustomerName, c.Region, SUM(s.Amount) as TotalPurchases,
COUNT(s.OrderID) as OrderCount FROM Sales s INNER JOIN Customers c ON
s.CustomerID = c.CustomerID GROUP BY c.CustomerName, c.Region ORDER
BY TotalPurchases DESC;
```

Explanation: "This uses an INNER JOIN to combine sales and customer tables on CustomerID, then aggregates by customer to show total purchases and order count. INNER JOIN ensures we only get customers who have made purchases."

Q3: "Find customers who made purchases in both 2023 and 2024"

```
SELECT CustomerID FROM Sales WHERE YEAR(OrderDate) IN (2023, 2024)
GROUP BY CustomerID HAVING COUNT(DISTINCT YEAR(OrderDate)) = 2;
```

Explanation: "This filters to only 2023 and 2024 orders, groups by customer, then uses HAVING to keep only customers with purchases in both distinct years. This identifies loyal repeat customers."

General Data Questions

Question	Answer Approach
How do you handle missing data?	Options: (1) Impute with mean/median for numerical data, (2) Use mode for categorical, (3) Drop rows if minimal, (4) Flag for business review. Choice depends on data volume and business impact. Always document the approach.
What's your data analysis process?	(1) Understand requirements and business questions, (2) Explore data to assess quality, (3) Clean and prepare data, (4) Analyze using SQL or statistical methods, (5) Visualize insights, (6) Present findings with recommendations.
How do you ensure data quality?	Validate at source with constraints, implement data profiling to check distributions, use automated testing, set up monitoring alerts, document data lineage, and regularly audit for accuracy and completeness.

Explain the difference between WHERE and HAVING	WHERE filters rows before grouping (used with individual records), while HAVING filters groups after aggregation (used with GROUP BY). Example: WHERE filters to specific dates; HAVING filters to regions with sales > $10K.

13 Hands-On Exercises

Build Your Portfolio

> **Why These Matter:** Employers want to see practical skills. Completing these exercises gives you portfolio pieces to showcase in interviews and demonstrates your ability to work with real data. Each exercise takes 30-60 minutes and uses free tools.

Data Warehousing Exercises (4)

Exercise 1: SQL Query Drill

Tool: SQL Fiddle (sqlfiddle.com) - Free online SQL editor
Task: Use a sample retail database to write queries:

- Calculate total revenue by store
- Find average order size by customer type
- Identify top 10 products by profit margin

Goal: Practice GROUP BY, aggregations, and JOINs
Time: 30 minutes

Exercise 2: ETL Workflow

Tool: Talend Open Studio (free download) or Power Query
Task: Download Kaggle's e-commerce sales CSV. Create an ETL job:

- **Extract:** Read from CSV
- **Transform:** Remove null values, standardize country names, calculate profit margin
- **Load:** Write to SQLite database

Goal: Understand complete ETL pipeline

Time: 45 minutes

Exercise 3: Schema Design

Tool: Lucidchart or Draw.io (free)

Task: Design a star schema for a movie theater data warehouse:

- **Fact Table:** Ticket sales (date, movie, customer, theater, price, quantity