

Contents

CHATGPT 4 Job Prompts	3
Job Based Common answers	4
Work-Experience	5
1. Business Intelligence Engineer and Analyst	5
2.1 KPIs	5
2.2 Dashboard:.....	6
2. Junior Data Analyst.....	7
2.1 KPIs Included	7
2.2 Dashboard.....	7
Competency Questions.....	9
1. Have you ever made a recommendation based on data that was initially met with scepticism? How did you handle it?	9
2. DESCRIBE A TIME WHEN YOU HAD TO EXPLAIN COMPLEX DATA FINDINGS TO A NON-TECHNICAL AUDIENCE. HOW DID YOU MAKE IT UNDERSTANDING? HOW DID YOU ENSURE CLARITY? OR TELL ME ABOUT A TIME WHEN YOUR ANALYSIS DIRECTLY INFLUENCED A BUSINESS DECISION. WHAT WAS THE IMPACT? OR DESCRIBE HOW YOU HAVE USED DATA TO UNCOVER CUSTOMER BEHAVIOUR PATTERNS. HOW DID THESE INSIGHTS INFLUENCE BUSINESS STRATEGY?	10
3. TELL ME A TIME WHEN YOU HAD TO TELL A STAKEHOLDER NO? HOW DID YOU GO ABOUT DOING IT? OR CAN YOU TELL US ABOUT A TIME WHEN YOU NEEDED TO MANAGE THE EXPECTATIONS OF STAKEHOLDERS WHOSE VIEWS WERE DIFFERENT FROM YOUR OWN?.12	12
4. Give an example of when you collaborated with cross-functional teams to drive a data initiative.....	13
5. What Are the Main Challenges in Data Analytics and Dashboard Building?	14
6. How Do You Check for Data Accuracy and Validation?	15
7. Real-World Example: Ensuring Data Accuracy through File-Based ETL Triggering.....	16
8. APIs.....	17
9. Can you describe the kind of data work you've done in your current role and how you achieved it?.....	18
10. If you're provided with a piece of data to perform a data analytics test on, what should you consider first?.....	19
11. How do you currently apply your data analytics skills in your day-to-day role?	21
12. CAN YOU DESCRIBE A TIME WHEN YOU HAD TO MANAGE MULTIPLE PROJECTS WITH TIGHT DEADLINES? HOW DID YOU PRIORITISE YOUR TASKS, AND WHAT STRATEGIES DID YOU USE TO ENSURE EVERYTHING WAS COMPLETED ON TIME?	22
13. TELL ME ABOUT A TIME WHEN YOU HAD TO CLEAN AND TRANSFORM MESSY DATA. WHAT APPROACH DID YOU TAKE, AND WHAT WAS THE OUTCOME? OR DESCRIBE A TIME WHEN YOU LED A COMPLETE PROJECT, THE APPROACH, THE CHALLENGES YOU FACED AND HOW YOU PROCEEDED? OR GIVE AN EXAMPLE OF A TIME WHEN YOU HAD TO WORK WITH INCOMPLETE OR AMBIGUOUS DATA. HOW DID YOU MAKE DECISIONS IN THAT SCENARIO? OR FAVOURITE POWER BI DASHBOARD AND WHY?	23

14. DESCRIBE A SITUATION WHERE YOU HAD TO WRITE A COMPLEX SQL QUERY TO EXTRACT INSIGHTS FROM A DATABASE. WHAT CHALLENGES DID YOU FACE, AND HOW DID YOU RESOLVE THEM? OR HAVE YOU BUILT DASHBOARDS FOR MONITORING KEY METRICS? WALK ME THROUGH YOUR PROCESS, FROM REQUIREMENT GATHERING TO IMPLEMENTATION. OR DESCRIBE A TIME WHEN YOU WORKED WITH A LARGE DATASET. HOW DID YOU ENSURE PERFORMANCE AND EFFICIENCY IN YOUR ANALYSIS?	24
15. Describe a project where you used statistical analysis to identify trends or solve a business problem. What techniques did you apply?	26
16. Have you ever automated a repetitive data task? What was the process, and how did it improve efficiency?	27
17. Tell me about a time when you had to learn a new data tool or technology quickly. How did you go about it, and how did it help your work?	28
18. Can you give an example of a time when you identified a potential business risk using data analysis? How did you address it?	29
19. Have you ever run an A/B test? What was the hypothesis, how did you analyse the results, and what actions followed?	30
20. Describe a situation where you worked with teams from different departments to complete a data-related project. What was the biggest challenge, and how did you handle it?	
31	
21. Can you tell us about a time you received negative feedback on a Power BI report, or when a report wasn't well used after launch.	32
Project:	34
1. Project: Claims Pendency Report – Healthcare Insurance Client	34
2. Project: Monthly Travel Insurance Dashboard – Stakeholder Reporting	35
3. Project: Claims Summary & Compliance Reporting – B2B Partners and Brokers	36
4. Project: Daily Business Reporting – Executive Dashboard for MD & Department Heads	37
5. Company Project 5: CRM Dashboard – Customer Satisfaction & Service KPIs	38
6. Company Project 6: Repeated Claims Detection – Fraud Prevention	39
7. Company Project 7: Sales Incentive Dashboard.....	40
8. Company Project 8: ETL Pipeline Optimisation.....	41
9. Company Project 9: Policy Renewal Analysis and Customer Retention Dashboard	42
10. Company Project 10: Automated Reporting Optimisation & Stakeholder Communication ...	43
11. Company Project 11: Power BI Adoption & Stakeholder Enablement	44
12. Company Project 12: Power BI Dashboard Redesign Based on User Feedback	45
13. Company Project 13: Multi-Project Management.....	46
14. Academic Project 1: Predicting London Bike Usage Using Machine Learning (Python)	47
Dashboard Design	48

CHATGPT 4 Job Prompts

1. Analyse this job with my CV and compare it in a table format.
2. Make a cover letter for this job, in simple and direct language, saying who I am, my relevant skills for this role, how I add value, potential gaps and how I plan to bridge them. I have completed my post-graduation studies at the University of Southampton.
3. If you get an opportunity, could you please go through my portfolio? https://vyom20798.github.io/VyomKhanna_Portfolio/ . It has all the details about my work experience, skills, education, letter of recommendations, Power BI dashboards, Python Codes etc document. In short, you can find everything over here.
4. Now make an answer to Please use this space to let us know about the skills and experiences you have gathered that could make you a great match for this position. make it in points for each point mentioned in the JD

Job Based Common answers

1. Yes, I am offered the role i have to serve 4 weeks of notice, and I will use that time to relocate. I, at present, live in temporary accommodation so that I can move for the job.
2. Yes, I have Graduate Visa until 23rd January 2027.
3. If you get an opportunity, could you please go through my portfolio? https://vyom20798.github.io/vyomkhanna_portfolio/. It has every detail of my past experience, projects, Python code, Power BI dashboards, letters of recommendation and transcripts. In short, you can find everything you need to know.

Work-Experience

1. Business Intelligence Engineer and Analyst

I worked as a Business Intelligence Engineer and Analyst on-site with the client **Care Health Insurance**, where I partnered closely with multiple departments including Claims, Finance, Sales, Product, and HR to deliver data-driven solutions and strategic insights. I was also trained in Qlik Sense and Qlik NPrinting to support advanced dashboard development and reporting automation.

My role involved analysing raw data using SQL and Python, and creating interactive dashboards in Qlik Sense and Power BI. These tools significantly enhanced decision-making efficiency by 30%. I gathered and documented business requirements using both Agile and Waterfall methodologies and translated them into meaningful KPIs and data visualisations, leading to a 25% improvement in the delivery of business insights.

I was responsible for transforming and integrating data from various sources into centralised data warehouses using SQL, and Qlik ETL pipelines. By optimising SQL queries, Power BI DAX logic, and Qlik data models, I reduced data loading times by 3 hours while achieving 95% data accuracy.

One of my key contributions was the automation of over 100 recurring reports for internal stakeholders and B2B clients, cutting manual reporting efforts by 70% and improving operational efficiency. I also designed incentive reports aligned with business KPIs, contributing to a 15% increase in quarterly sales performance.

In addition to delivering technical solutions, I acted as a liaison between the Claims and HR teams, maintaining regular communication with senior leadership including the CHRO and Head of Claims. I was recognised for the high quality of my work and received a **Letter of Recommendation from the Head of Claims**.

I also mentored junior developers and interns in SQL optimisation, SSRS automation, and Power BI dashboarding, fostering knowledge sharing and strengthening team capability. Furthermore, I collaborated with IT to improve server and warehouse performance, reducing data retrieval times by 20%.

I regularly utilised advanced Excel functions such as pivot tables, VLOOKUP, and conditional logic for quick validations, ad-hoc insights, and detailed business reporting, ensuring agile responses to evolving business needs.

2.1 KPIs

Claims Department

- **Claim Settlement Ratio (%)** = (Number of Claims Settled / Total Claims Received) × 100
- **Average Claim Processing Time (Days)** = Total Time Taken to Settle Claims / Number of Claims Settled
- **Claim Rejection Rate (%)** = (Number of Claims Rejected / Total Claims Processed) × 100
- **Fraudulent Claims Identified (%)** = (Fraudulent Claims / Total Claims Processed) × 100
- **Pending Claims Ageing Analysis** = Categorise: 0–7 days, 8–15 days, 16–30 days, 30+ days

Finance Department

- **Expense Ratio (%)** = (Operating Expenses / Net Premium Earned) × 100
- **Loss Ratio (%)** = (Incurred Claims / Net Premiums Earned) × 100
- **Revenue vs Claims Paid** = Trend comparison over months or quarters
- **Policy Profitability per Product** = (Premium Collected – Claims Paid – Costs) / Number of Policies

- **Cash Flow Variance (%)** = (Actual Cash Flow – Forecasted Cash Flow) / Forecasted Cash Flow × 100

Sales Department

- **Sales Conversion Rate (%)** = (Number of Policies Sold / Number of Leads) × 100
- **Revenue per Sales Agent** = Total Revenue / Number of Agents
- **Sales Target Achievement (%)** = (Actual Sales / Sales Target) × 100
- **Lead Response Time (Hours)** = Total Time to First Contact / Number of Leads
- **Cross-sell/Upsell Rate (%)** = (Number of Add-on Sales / Total Sales) × 100

Product Team

- **Product Adoption Rate (%)** = (New Users or Policies / Total Target Audience) × 100
- **Claims per Product Line** = Total Claims Received for Product / Total Policies Sold
- **Churn Rate (%)** = (Policies Cancelled / Total Active Policies) × 100
- **Average Revenue per Product (ARPP)** = Total Revenue / Number of Product Policies Sold
- **Customer Feedback Score** = From surveys or Net Promoter Score (NPS)

2.2 Dashboard:

Claims:

- Claim settlement trend (line chart)
- Claims ageing breakdown (stacked bar)
- Top reasons for claim rejection (pie chart)
- Rejection % vs fraud detection % (KPI tiles)

Finance:

- Revenue vs claims paid (dual line chart)
- Expense & loss ratio tiles with trends
- Cash inflow/outflow waterfall chart
- Product-wise profitability comparison (bar chart)

Sales:

- Sales funnel with lead drop-offs
- Agent performance leaderboard
- Conversion rate by product (bar/heatmap)
- Daily sales vs targets (gauge or bullet chart)

Product:

- Product-wise revenue & claims ratio (bar chart)
- Product churn trend (line)
- Customer feedback distribution (pie chart)
- Comparison of product profitability (table)

2. Junior Data Analyst

I joined Nipa International during the COVID-19 pandemic, transitioning from a mechanical engineering background into analytics. To support this career change, I undertook one month of structured training in SQL, Excel, and Tableau, enabling me to contribute effectively to the data team.

In this role, I was responsible for extracting and cleaning manufacturing data from the ERP system using MySQL, which helped reduce data processing time by 20%. I designed and maintained performance reports in Excel and SQL, focusing on key operational metrics such as production efficiency, downtime, and defect rates, resulting in a 15% improvement in process monitoring.

By analysing raw material usage patterns and waste data, I identified potential annual cost savings of ₹25 lakhs (approx. £24,000). Additionally, I supported internal audit readiness by creating standardised reporting templates, which helped cut down audit preparation time by 30%.

2.1 KPIs Included

Production & Operational Efficiency

- **Production Efficiency Rate (%)** = $(\text{Actual Output} / \text{Standard Output}) \times 100$
- **Downtime Hours per Week** = Tracks unproductive machine or line time.
- **Defect Rate (%)** = $(\text{Number of Defective Units} / \text{Total Units Produced}) \times 100$
- **Cycle Time per Unit** = Measures how long it takes to produce one item.
- **On-Time Production Rate (%)** = Jobs completed within the scheduled time frame.

Cost & Resource Optimisation

- **Material Yield Rate (%)** = $(\text{Good Material Output} / \text{Raw Material Input}) \times 100$
- **Scrap or Waste Cost per Month**
- **Cost Savings Identified (INR or £)** = From raw material analysis or waste reduction efforts.
- **Variance Between Planned vs Actual Material Usage**

Reporting & Audit Support

- **Standard Report Reuse Rate** = % of reports/templates reused in audit cycles.
- **Audit Preparation Time (Hours)** = Compare before vs after implementing templates.
- **Data Processing Time Reduction (%)** = Time taken to clean and process data in your SQL pipeline.

2.2 Dashboard

Production Overview Dashboard

- **Top Tiles/KPIs:** Total Units Produced, Defect Rate, Downtime Hours, Efficiency Rate
- **Charts:**
 - Line chart for daily/weekly production output
 - Bar chart for downtime by machine or shift
 - Pie chart for defect types

Cost & Material Efficiency Dashboard

- **Top Tiles:** Material Cost per Unit, Savings Identified, Scrap Volume
- **Charts:**
 - Bar chart of scrap by raw material

- Line graph for material usage trend over months
- Variance table (planned vs actual usage)

Audit & Reporting Dashboard

- **Top Tiles:** Report Reuse Rate, Avg. Time to Prepare Report, % of Data Cleaned
- **Tables:**
 - Before vs after comparison of audit prep time
 - List of reusable templates and their impact

Data Quality & Process Dashboard

- **KPIs:** % Null Values Cleaned, SQL Query Processing Time, Weekly Data Volumes Processed
- **Chart:**
 - Heatmap showing error-prone data fields
 - SQL processing trendline before vs after optimisation

Competency Questions

1. Have you ever made a recommendation based on data that was initially met with scepticism? How did you handle it?

Situation:

While working as a Business Intelligence Engineer at Team Computers Pvt Ltd, I was assigned to analyse the claims pendency report for a healthcare insurance client. The client was facing a backlog of pending claims, leading to delayed reimbursements, customer dissatisfaction, and regulatory pressure. Initial reports indicated that over 30% of claims were pending beyond the standard processing time, but the root cause was unclear.

Task:

When I presented my findings to senior stakeholders, they were initially skeptical, believing that the issue was purely due to high claim volumes during the pandemic. My task was to identify the true bottlenecks causing delays and propose data-driven process optimisations to improve claims turnaround time.

Action:

To address this, I:

- Extracted and analysed claims data using SQL and Power BI, segmenting pending claims by type, processing stage, and assigned teams.
- Discovered that 45% of delayed claims were stuck in manual verification, primarily due to missing documentation and unclear eligibility criteria.
- Designed an interactive Power BI dashboard that visualised claims pendency trends, processing time by team, and key bottlenecks.
- Recommended automating claims verification for certain policy types using business rules and AI-driven document validation, reducing reliance on manual checks.
- Proposed a triage system, prioritising claims based on complexity, to ensure that low-risk claims were processed faster while complex cases received dedicated attention.

Result:

- The automation and triage system reduced average claim processing time by 40%, significantly improving operational efficiency.
- The backlog of pending claims decreased by 55% within three months, leading to faster reimbursements and improved customer satisfaction scores.
- Regulatory compliance improved, as on-time claim resolution increased from 70% to 92%, reducing the risk of penalties.
- The client officially adopted the new claims processing framework, and I was recognised for providing a data-driven solution that optimised operational workflows.

2. DESCRIBE A TIME WHEN YOU HAD TO EXPLAIN COMPLEX DATA FINDINGS TO A NON-TECHNICAL AUDIENCE. HOW DID YOU MAKE IT UNDERSTANDING? HOW DID YOU ENSURE CLARITY? OR TELL ME ABOUT A TIME WHEN YOUR ANALYSIS DIRECTLY INFLUENCED A BUSINESS DECISION. WHAT WAS THE IMPACT? OR DESCRIBE HOW YOU HAVE USED DATA TO UNCOVER CUSTOMER BEHAVIOUR PATTERNS. HOW DID THESE INSIGHTS INFLUENCE BUSINESS STRATEGY?

Situation:

While working with a health insurance client, leadership noticed a worrying drop in policy renewal rates. The Sales and CRM teams had assumptions about the causes but lacked clear evidence, and they struggled to interpret the raw datasets themselves. I was asked to analyse customer behaviour, identify the real drivers behind non-renewals, and explain the findings in a way that non-technical teams could immediately understand and act on.

Task:

My responsibility was to clean and merge datasets from CRM, premiums, and claims, uncover patterns behind customer drop-off, and present the insights in a simple, relatable way. The goal was not only to diagnose *why* customers were leaving but also to recommend actions that leadership could confidently implement to improve retention.

Action:

To break down complex data into clear, actionable insights, I followed three steps:

1. Deep Behavioural Analysis

- Combined CRM, premium change %, and claim history into a single analytical model.
- Segmented customers into renewed vs non-renewed groups to spot behavioural differences.
- Identified four strong patterns behind non-renewals, and simplified them into digestible points:
 1. **40%** left due to premium hikes — mostly price-sensitive customers.
 2. **30%** didn't realise their policy was expiring — a clear communication gap.
 3. **20%** had unresolved claim disputes — creating dissatisfaction.
 4. Remaining customers were B2B partners who had switched insurers.

2. Translating Complex Data for Non-Technical Teams

To make the insights easy to understand:

- Built a Power BI dashboard with pie charts, bar charts, and trend lines showing renewal vs non-renewal patterns.
- Used side-by-side comparisons so Sales and CRM teams could visually see why certain customers behaved differently.
- Used relatable analogies — for example, comparing renewal behaviour to gym memberships: *people join with intention, but if costs rise or they don't feel supported, they stop seeing value.*
- Simplified all findings into a clear 3-minute narrative that sales teams could use directly during customer outreach.

3. Steering Business Strategy with Data

Based on the behavioural insights, I recommended three strategic interventions:

- **Automated renewal reminders** to address the 30% communication gap.
- **Soft-touch loyalty discounts** for price-sensitive segments.
- **Dedicated claim-support follow-ups** to rebuild trust among customers with disputes.

Result:

- Leadership quickly understood the insights and approved all three recommendations.
- After implementation, **renewal rates increased by 20% within three months.**

- The Sales and CRM teams began using the dashboard during weekly meetings to track progress.
- I was formally recognised by senior leaders for presenting complex data in a simple, actionable format and for directly shaping the company's customer-retention strategy.

3. TELL ME A TIME WHEN YOU HAD TO TELL A STAKEHOLDER NO? HOW DID YOU GO ABOUT DOING IT? OR CAN YOU TELL US ABOUT A TIME WHEN YOU NEEDED TO MANAGE THE EXPECTATIONS OF STAKEHOLDERS WHOSE VIEWS WERE DIFFERENT FROM YOUR OWN?

Situation:

While working with a healthcare insurance client at Team Computers Pvt Ltd, a stakeholder from the finance team requested an automated report that included a detailed data dump. However, the data file was over 70MB, and the company's email server had a size limit of 25MB, meaning it couldn't be sent via email.

Task:

I had to communicate to the stakeholder that their request wasn't feasible due to technical limitations, but also provide an alternative that still met their needs.

Action:

To address this, I:

- Explained the limitation clearly, saying, "*Our email system can't handle files over 25MB, so sending a 70MB report won't work.*"
- Suggested alternative solutions, including:
 1. Uploading the file to a secure shared drive (like SharePoint or Google Drive) and sending an automated link instead.
 2. Creating a Power BI dashboard where they could filter and download only the specific data they needed, reducing file size.
 3. Splitting the report into smaller, more manageable files that could be sent in multiple emails.
- Demonstrated how the new approach would be more efficient, showing that using a shared drive or dashboard would allow real-time access instead of waiting for an email.

Result:

- The stakeholder agreed to use a shared drive, making it easier for them to access updated reports anytime without file size issues.
- The new approach improved efficiency by eliminating manual email requests and ensuring secure data sharing.
- I was recognized for solving the issue proactively and improving reporting processes.

4. Give an example of when you collaborated with cross-functional teams to drive a data initiative.

Situation:

At Team Computers Pvt Ltd, I worked with a healthcare insurance client where the claims, medical, and CRM teams relied on manual data entry for processing claims and updating customer records. This led to errors, inconsistent data, and slow processing times, making it difficult to perform accurate analysis and reporting.

Task:

My goal was to eliminate manual data entry errors and speed up data processing by implementing a drop-down selection system in the CRM and claims processing tools. This would ensure standardised data entry and allow for faster and more accurate analysis.

Action:

To implement this, I:

- Worked with the claims team to identify the most common errors caused by free-text entry, such as misspelt provider names and incorrect claim categories.
- Collaborated with the medical team to define standardised dropdown options for diagnoses, procedures, and claim statuses, reducing ambiguity.
- Engaged with the CRM team to implement drop-down fields for customer interactions, policy status updates, and follow-up actions, ensuring consistency.
- Tested and refined the system with end-users, gathering feedback and making adjustments to improve usability.
- Trained teams on the new system, ensuring a smooth transition from manual entry to structured selection fields.

Result:

- Data accuracy improved by 50%, reducing errors in claims processing and customer records.
- Claims processing speed increased by 30%, as teams no longer had to manually correct data mistakes.
- Analysis and reporting became more efficient, as standardised data allowed for more reliable trend identification and decision-making.
- The client officially adopted the drop-down entry system, leading to long-term improvements in workflow efficiency.

5. What Are the Main Challenges in Data Analytics and Dashboard Building?

Building effective data analytics systems and dashboards involves navigating several technical, analytical, and organisational challenges. These include:

1. Technical Challenges

- Data Integration: Combining data from various sources like APIs, databases, and spreadsheets can be difficult due to inconsistent formats and structures.
- Data Quality: Incomplete, inconsistent, or duplicate data can compromise the reliability of insights.
- Real-Time vs. Batch Processing: Real-time data processing is complex and resource-intensive, while batch updates may not meet business needs for immediacy.
- Scalability & Performance: As data grows, dashboards can slow down or crash unless properly optimised.
- Tool Limitations: BI tools may lack support for specific visualisations or advanced customisations, requiring workarounds or custom development.

2. Analytical Challenges

- Defining the Right Metrics: Identifying meaningful KPIs that align with business goals is crucial but often difficult.
- Contextualising Data: Dashboards must present data in a way that adds context, comparisons, and clarity, not just numbers.
- Data Overload: Too much data can overwhelm users, making it hard to focus on what matters most.
- Bias and Misinterpretation: Poorly designed dashboards can mislead users, so careful attention to layout and narrative is essential.

3. Human and Organisational Challenges

- Stakeholder Alignment: Different teams may have conflicting needs or definitions for the same metrics, making consensus a challenge.
- User Adoption: If dashboards are not intuitive, useful, or relevant, they will be underused or ignored.
- Maintenance and Updates: Dashboards require ongoing maintenance as data sources, business logic, or goals evolve.
- Security and Access Control: Ensuring appropriate access to sensitive information without over-restricting usability is a fine balance.

6. How Do You Check for Data Accuracy and Validation?

Ensuring data accuracy and validating data at every step is critical to building trust in analytics.

Here's how I typically approach it:

1. Understand the Source Data

- Know your sources: Begin by understanding where the data is coming from (databases, APIs, flat files, etc.) and who owns it.
- Check source documentation: Validate against official data definitions and expected formats.

2. Schema Validation

- Data type checks: Ensure fields are using the correct types (e.g., dates, integers, strings).
- Required fields: Check for missing mandatory values (nulls or blanks).
- Range checks: Validate numeric values against expected min/max limits (e.g., age can't be negative).

3. Referential Integrity

- Ensure keys match across related tables (e.g., every order should match an existing customer).
- Look out for orphan records or duplicates in supposedly unique fields.

4. Business Rule Validation

- Cross-check against known business rules. For example:
 - A customer's signup date shouldn't be after their purchase date.
 - Discount % should not exceed 100%.

5. Summary Level Checks

- Compare aggregates: Cross-verify total counts, sums, averages with source system reports or known benchmarks.
- Trend analysis: Look at historical data trends to identify outliers or sudden spikes that might signal data issues.

6. Automated Validation Scripts

- Set up automated data validation pipelines with tools like:
 - SQL scripts for QA
 - dbt tests (if using dbt)
 - Great Expectations, or custom Python validation scripts

7. Manual Spot Checks

- Pull random records and trace them back to the source to ensure end-to-end accuracy.
- Especially useful during initial data onboarding or ETL changes.

8. Logging & Alerts

- Implement logging to track anomalies (e.g., null rates, schema changes).
- Set up alerts when data volumes drop unexpectedly or fields go missing.

7. Real-World Example: Ensuring Data Accuracy through File-Based ETL Triggering

In my previous organisation, we were dealing with multiple ETL jobs pulling data from various sources, often on overlapping schedules. To ensure data accuracy and consistency across pipelines, I implemented a file-based trigger mechanism.

What I Did:

I introduced a "file variable" check in our ETL workflow. Each job would only start once a specific control file (or data file) was updated. This acted as a validation gate, ensuring that:

- The upstream data had been fully refreshed.
- Jobs didn't start on stale, incomplete, or outdated datasets.
- Dependencies across different pipelines were respected.

Why It Worked:

- It prevented race conditions where one job might start before the data it depends on was ready.
- It ensured data consistency across related jobs, especially when combining data from multiple systems.
- It acted as a simple but effective form of data freshness validation before transformation began.

Tools/Tech Used:

- Shell scripts and job orchestration tools (like Airflow or cron)
- File timestamp checks (stat, ls -lt, etc.)
- Environment variables or flags in the ETL pipeline

Outcome

This approach significantly reduced downstream data issues, improved reliability in reporting, and made debugging much easier because we had a clear audit trail of when each dataset was last updated.

8. APIs

Data Source APIs

These allow you to pull data from various platforms into your analytics or BI tools.

- **Google Analytics API** – For tracking and analysing website traffic and user behavior.
- **Facebook Graph API** – Access data from Facebook for social media analytics.
- **Twitter API** – Pull tweets, trends, and engagement data for sentiment or performance analysis.
- **YouTube Data API** – For analysing video performance and audience engagement.
- **Salesforce API** – Access CRM data for customer and sales analytics.
- **Shopify API** – E-commerce data integration (orders, customers, inventory).
- **Stripe API / PayPal API** – Payment and transaction data analysis.
- **Google Ads / Facebook Ads API** – Campaign performance metrics.

Database and Storage APIs

APIs to interact with data stored in cloud services or traditional databases.

- **SQL-based APIs** (PostgreSQL, MySQL, Microsoft SQL Server) – Structured queries for analytics.
- **MongoDB API** – For accessing NoSQL document databases.
- **Amazon S3 API / Google Cloud Storage API / Azure Blob Storage API** – Manage and retrieve big data.
- **Firebase Realtime Database / Firestore API** – For mobile and web app analytics.

Data Integration and ETL APIs

Used to extract, transform, and load data between systems.

- **Apache Airflow API** – Workflow orchestration for data pipelines.
- **Talend API / Informatica API** – Enterprise-grade ETL solutions.
- **Fivetran / Stitch / Zapier APIs** – Simplified data integration services.

BI and Visualisation Tool APIs

These help embed dashboards, automate reports, or manipulate data programmatically.

- **Tableau REST API** – Automate dashboard management, refresh extracts, manage users.
- **Power BI REST API** – Embed reports, push data, manage workspaces.
- **Looker API** – Query LookML models, automate dashboards.
- **Google Data Studio API** – For managing data sources and reports.

Machine Learning & Analytics APIs

Used for predictive analytics, statistical modeling, and deeper insights.

- **Google Cloud ML / Vertex AI API**
- **Amazon SageMaker API**
- **Microsoft Azure ML API**
- **OpenAI API / Hugging Face API** – Natural language processing, classification, etc.
- **IBM Watson API** – For sentiment analysis, predictive analytics, etc.

Geospatial & External Data APIs

Enhance your BI with external data context.

- **Google Maps API / Mapbox API** – For location intelligence.
- **OpenWeatherMap API** – Weather data for impact analysis.
- **Quandl API / FRED API** – Financial and economic data.

9. Can you describe the kind of data work you've done in your current role and how you achieved it?

Answer:

Sure. In my current role as a part-time Sales Assistant at Poundland and Dealz, I proactively sought opportunities to gain insight-related experience after receiving feedback that I lacked UK-based analytics exposure.

Situation & Task:

Although my main responsibility is operating tills, I expressed interest in supporting data analysis at the store level. While there was no formal analyst role, management allowed me to assist with weekly and monthly reporting tasks to identify underperforming products and suggest improvements.

Action:

I compiled and monitored reports on sales per SKU, product return rates, average basket size, and units sold per week. I cross-referenced this data with real-time customer interactions at checkout to explain product performance, like why some items weren't selling due to shelf placement or packaging. Based on these insights, I made several recommendations, including product repositioning and bundling, which led to noticeable sales lifts.

I also supported the creation of Tableau dashboards by providing input on what visual elements would help managers most, like highlighting peak shopping hours or top-selling SKUs by time of day. I didn't build the dashboards myself due to access limitations, but I contributed directly to the design logic and KPI selection.

Additionally, I actively participated in weekly team meetings, sharing data-backed insights and real-world observations. For example, I noticed perk app usage was low during certain shifts, which led to a targeted upselling strategy.

Result:

This hybrid role improved my analytical thinking, commercial awareness, and gave me hands-on experience translating data into decisions. Several of my suggestions were implemented and led to measurable performance improvements in product visibility and customer engagement.

10. If you're provided with a piece of data to perform a data analytics test on, what should you consider first?

Answer:

When I receive a dataset to perform an analytics test, the first thing I do is take a step back and understand the context and the objective. I ask myself: What question are we trying to answer? or What decision will this analysis support? Without a clear goal, even the most advanced analytics can miss the mark.

Once that's clear, I focus on understanding the structure and source of the data:

- What does each column mean?
- Are the data types correct, like dates, numbers, and categories?
- Where did the data come from, and how reliable is it?

Then I move to data quality checks, which are crucial:

- I look for missing values, outliers, duplicates, and inconsistencies.
- I also check for referential integrity. Do IDs link correctly between tables?
- If it's time series or transactional data, I validate the granularity and timestamps.

Once I've assessed quality, I think about relevance and usability. Are all the columns needed? Can any be derived or simplified? If I'm doing a test for insight generation, I'll do basic exploratory data analysis, summary stats, correlations, maybe a few visuals to get a feel for distributions and relationships.

Finally, I think about constraints and ethics:

- Do I need to anonymise anything?
- Are there limitations in how the data can be used due to GDPR or company policies?

To help you understand with an example:

Example:

S – Situation

At Care Health Insurance, the Claims department was under review because senior leadership noticed long processing times and customer complaints. I was handed a dataset from multiple sources — claim submissions, policy types, settlement records — and was asked to perform analytics to identify bottlenecks and recommend improvements.

T – Task

My task was to analyse this raw data to uncover why some claims took over 15 days to settle — and whether patterns existed around product types, locations, or agents. But before running any analysis or visualising trends, I had to evaluate the dataset carefully.

A – Action

Step 1: Objective

I clarified with the audit lead: What exactly are we measuring? Was it turnaround time from claim receipt to approval? Or from submission to settlement? That distinction affects how I calculate metrics like average processing time or SLA breaches.

Step 2: Data Quality

Next, I ran checks in SQL for **missing values**, especially in critical columns like claim submission or approval date. I found 7% of rows had null values, and a few cases had negative processing durations due to incorrect date stamps — those needed fixing before anything else.

I also removed **duplicate entries** using DISTINCT logic in SQL and validated data types for example, ensuring policy IDs were integers and dates were in a valid format.

Step 3: Feature Engineering

Then I derived new variables: processing time in days, product segment, and risk category. I also bucketed processing durations into ranges (0–7, 8–15, 16+ days) to identify which claim types were more delayed.

I then joined this data with employee assignment tables to explore correlations between performance and team structure a valuable angle for audit.

Step 4: Compliance

Because the dataset included sensitive customer and policy data, I ensured that PII was masked and that our servers complied with data protection standards. This included role-based access and storing interim outputs on secure drives.

R – Result

After cleaning and preparing the data, I built a dashboard in Qlik Sense that revealed a striking insight: **one region had a 35% higher proportion of claims delayed beyond 15 days**. We traced it to a manual verification step for certain policy types.

My recommendations led to an automated document check in that region and helped reduce the average processing time by 18%.

Additionally, this audit was escalated to the board as a case study in process improvement and I received a Letter of Recommendation from the Head of Claims.

Summary

So, when I get a dataset for analysis, the first thing I consider is: what's the business problem we're solving? Then I move into validating the dataset's quality, performing engineering to derive useful variables, and ensuring compliance before building any models or dashboards.

It's a structured, business-focused approach and one I've successfully applied all places I work and plan to work with ideology.

11. How do you currently apply your data analytics skills in your day-to-day role?

In my current role, I apply data analytics across the full lifecycle from maintaining data pipelines to delivering business insights and resolving stakeholder queries all with a strong focus on accuracy, efficiency, and business impact.

Each day starts with reviewing ETL logs and job schedulers to ensure all pipelines have run properly. If a job fails, I backtrack the issue, perform a root cause analysis, and resolve whether it's a schema change, data load failure, or connection timeout. In most cases, I'm able to fix the issue independently, but I also collaborate with senior developers or the team lead when deeper intervention is needed. This ensures data availability for dashboards and reports, which are essential for daily operations.

I use SQL to extract and transform data and tools like Power BI and Qlik Sense to build dashboards for teams across product, finance, and operations. These dashboards monitor KPIs like claim turnaround time, sales trends, or customer engagement helping teams make informed, real-time decisions.

I'm also actively involved in stakeholder collaboration. Business users or clients often raise ad hoc questions or data discrepancies. In those cases, I validate the concern, trace the issue through the pipeline, and resolve or explain it with transparency. I also document fixes and work to improve the long-term data quality process.

Overall, I apply my data analytics skills not just to build dashboards or write queries, but to ensure data integrity, reliability, and clear communication which I know are especially important in a financial environment like HSBC that prioritises compliance, transparency, and informed decision-making.

Here's a quick example using the STAR framework:

1. Situation: One Monday morning, a sales dashboard used by senior leadership showed zero values across all regions, just before their weekly performance meeting.
2. Task: I needed to quickly identify and fix the issue to ensure leadership had access to accurate performance data.
3. Action: I checked the QMC and ETL logs and found that an upstream job had failed due to a schema change from the data provider. I ran a root cause analysis, updated the field mappings, reran the pipeline, and validated the data to ensure accuracy.
4. Result: The dashboard was restored within 45 minutes, and I later implemented a fail-alert system and data validation layer to catch similar issues proactively. This helped build long-term trust and improved data reliability.

Beyond technical work, I actively contribute in Agile ceremonies sprint planning, backlog grooming, UAT, where I define data requirements, translate business goals into technical specifications, and ensure the final outputs align with user needs.

12. CAN YOU DESCRIBE A TIME WHEN YOU HAD TO MANAGE MULTIPLE PROJECTS WITH TIGHT DEADLINES? HOW DID YOU PRIORITISE YOUR TASKS, AND WHAT STRATEGIES DID YOU USE TO ENSURE EVERYTHING WAS COMPLETED ON TIME?

Situation:

As a BI Engineer at Team Computers, I was responsible for report automation, optimisation of ETL pipelines, and dashboard creation. Over a period of 18 months, I worked closely with cross-functional teams and frequently handled multiple tasks simultaneously.

Tasks:

At one point, I was managing three major tasks at the same time:

1. Preparing a sales executive report for the Sales and Payroll team.
2. Automating the claims processing report.
3. Optimising commission reports for the Finance team.

Action:

To manage these tasks effectively:

1. Prioritisation: I identified and focused on the task with the highest business impact, the sales executive report, as it was time-sensitive.
2. Time Management: I created a schedule to allocate time effectively across all tasks, accounting for priorities and unexpected challenges.
3. Communication: I maintained strong interpersonal relationships with stakeholders. I transparently communicated my workload and timeline constraints, which helped manage expectations.
4. Stakeholder Management: I kept all stakeholders regularly informed about progress, timelines, and any potential delays.

Result:

I successfully submitted the sales executive report on time, resolved the authorised claim pendency, and optimised the commission report, while also handling several ad-hoc requests.

13. TELL ME ABOUT A TIME WHEN YOU HAD TO CLEAN AND TRANSFORM MESSY DATA. WHAT APPROACH DID YOU TAKE, AND WHAT WAS THE OUTCOME? OR DESCRIBE A TIME WHEN YOU LED A COMPLETE PROJECT, THE APPROACH, THE CHALLENGES YOU FACED AND HOW YOU PROCEEDED? OR GIVE AN EXAMPLE OF A TIME WHEN YOU HAD TO WORK WITH INCOMPLETE OR AMBIGUOUS DATA. HOW DID YOU MAKE DECISIONS IN THAT SCENARIO? OR FAVOURITE POWER BI DASHBOARD AND WHY?

Situation:

At Care Health Insurance, I was working on automating the Claims Pending Report, which involved pulling data from multiple departments like TPA, Cashless, and Medical. The data was fragmented, inconsistent, and filled with missing values, especially for key dates and claim statuses.

Task:

My goal was to create a clean, unified dataset for accurate reporting and dashboarding, so that stakeholders could track pending claims, processing time, and departmental backlogs in real time.

Action:

- I used SQL to extract and join data from various source systems, ensuring keys like Policy Number and Claim ID were consistent across tables.
- I performed schema validation to enforce correct data types and removed duplicate or zero-value claims.
- I created derived columns such as "Claim Ageing" (using month-end date – last document received date) to enable trend and performance analysis.
- For missing or null values, I used conditional logic (e.g., fallback to alternative timestamps or manually flagged entries).
- I implemented ETL transformations in Qlik and introducing filters to exclude ineligible data (like closed or test records) and adding standardised status flags.
- Finally, I designed a Power BI dashboard to visualise pending claims by age bracket, department, and claim type.

Result:

- Reduced the dataset error rate by over 90%, enabling reliable reporting.
- Helped cut daily pending claims from 80,000 to 38,000 over 18 months.
- Improved claims processing productivity by ~52%.
- The cleaned data pipeline became a template for other departments, and the project was cited in internal reviews for operational improvement.

14. DESCRIBE A SITUATION WHERE YOU HAD TO WRITE A COMPLEX SQL QUERY TO EXTRACT INSIGHTS FROM A DATABASE. WHAT CHALLENGES DID YOU FACE, AND HOW DID YOU RESOLVE THEM? OR HAVE YOU BUILT DASHBOARDS FOR MONITORING KEY METRICS? WALK ME THROUGH YOUR PROCESS, FROM REQUIREMENT GATHERING TO IMPLEMENTATION. OR DESCRIBE A TIME WHEN YOU WORKED WITH A LARGE DATASET. HOW DID YOU ENSURE PERFORMANCE AND EFFICIENCY IN YOUR ANALYSIS?

Situation:

I worked on building an end-to-end executive reporting system that required integrating multiple large datasets — including claims, sales, finance, and policy information — which collectively held over five million rows. The initial reporting approach was slow, queries were taking too long to run, refreshes were inconsistent, and leadership needed a dashboard they could rely on every morning. The ETL processes were also failing occasionally because upstream data loads weren't synchronised, leading to stale or incomplete information.

Task:

My responsibility was to:

1. Create a unified, high-quality data model from inconsistent source systems.
2. Optimise several complex SQL queries that were essential for KPI calculations.
3. Redesign and stabilise the ETL pipelines so daily loads were smooth and automated.
4. Improve dashboard performance so that senior management could access insights quickly and confidently.

Action

1. SQL Development & Query Optimisation

To handle the complexity of the datasets, I started by analysing each schema and identifying reliable keys to link them. Many tables lacked a single unique join field, so I created composite keys — for example, combining policy number and start date — to make joins accurate.

I then rewrote the queries using:

- CTEs to break down the logic into readable sections.
- Window functions (ROW_NUMBER, RANK) to isolate the latest valid record where duplicate transactions existed.
- CASE statements to derive risk segments, ageing categories, and business logic.
- Early filtering, column pruning, and index-friendly joins to reduce scanning time.

For heavy tables, I built materialised views and pre-aggregated layers so Power BI would process far less data during refresh.

This work alone reduced execution time by more than half.

2. ETL Re-Engineering & Automation

The next step was making sure the data pipeline was reliable.

I redesigned the ETL architecture by separating it into:

- Staging layer – raw extract with schema validation
- Transformation layer – joins, business rules, calculated fields
- Presentation layer – clean tables for BI consumption

To address refresh failures, I introduced a file-based trigger mechanism. Jobs only executed after verifying that the upstream dataset had been updated. This prevented:

- race conditions
- stale data
- mismatched time windows

I also switched from full loads to incremental loading, meaning only new or changed records were pulled each day. This reduced processing time significantly.

To ensure quality at every step, I added:

- schema checks
- null audits
- referential integrity tests
- automated warning logs for unusual volume changes

As a result, ETL failures dropped sharply, and downstream issues became far easier to diagnose

3. Data Modelling & Star Schema Design

For the dashboard layer, I rebuilt the model using:

- a star schema
- one central fact table
- multiple dimension tables

This approach reduced relationship complexity and prevented circular dependencies.

I avoided creating heavy calculated columns in Power BI and moved most transformations into SQL or the ETL layer, where they were more efficient.

Additionally, I created summary tables to handle slicers and time-based partitions, which meant visuals loaded without scanning the entire fact table.

4. Dashboard Optimisation

To optimise performance for senior leadership:

- I simplified and reorganised visuals to eliminate unnecessary calculations.
- Rewrote slow DAX expressions into more efficient equivalents.
- Disabled auto-date hierarchies to improve model efficiency.
- Limited cross-filtering to reduce processing overhead.
- Used bookmarks and drill-through pages instead of data-heavy tables.

Users could now navigate metrics like settlement ratios, loss ratios, revenue trends, claim ageing, and sales KPIs in seconds.

Result

The combined improvements delivered significant operational value:

- Refresh time reduced by over 60%, from more than 30 minutes to under 10.
- Dashboard load time improved substantially, enabling reliable use during daily morning meetings.
- Data accuracy exceeded 95% due to strong validation and standardised transformations.
- Manual reporting effort dropped by over 70%, thanks to ETL automation.
- The architecture became a reusable template, adopted by multiple teams for their own analytics requirements.
- Leadership gained a fast, stable, and trustworthy view of organisation-wide performance.

15. Describe a project where you used statistical analysis to identify trends or solve a business problem. What techniques did you apply?

Situation:

As part of a project on London bike usage forecasting, I aimed to help city planners optimise bike distribution by analysing usage patterns and identifying demand trends based on weather and time factors.

Task:

The objective was to predict ride volume using historical trip data and external weather data, enabling smarter resource allocation at high-traffic stations across London.

Action:

- I collected trip data from Transport for London and weather data from Visual Crossing.
- I performed data preprocessing: removing outliers (e.g., trips under 1 minute), merging datasets, and engineering new features like day of week, hour bins, and temperature categories.
- Applied descriptive statistics to identify peak usage periods, and used correlation analysis to understand relationships between temperature and ride volume.
- Built and compared multiple regression models including Linear Regression, Random Forest, and Support Vector Machines (SVM).
- Used evaluation metrics like Mean Squared Error (MSE) and R^2 to assess model performance.

Result:

- Found that temperature was the strongest predictor of bike usage, with Thursday and Friday afternoons being peak periods.
- Identified Hyde Park Corner as the busiest station.
- Delivered actionable insights to suggest pre-positioning bikes on warm days, improving availability and customer satisfaction.
- This project demonstrated how statistical analysis can directly influence operational decisions and urban planning strategies.

16. Have you ever automated a repetitive data task? What was the process, and how did it improve efficiency?

Situation:

While working as a Business Intelligence Developer with Care Health Insurance, I noticed that several teams were manually preparing daily and weekly reports using Excel — particularly for **claims pendency, team productivity, and B2B compliance reporting**.

Task:

My goal was to **automate over 100 recurring reports** to eliminate manual effort, reduce human error, and ensure timely, accurate delivery to internal stakeholders and external clients.

Action:

- **Requirement Gathering:** Interviewed Claims, Medical, and HR teams to understand reporting needs, formats, and delivery schedules.
- **Data Preparation:** Used **SQL** to extract and clean data from multiple systems (claims database, HRMS, finance tools).
- **Automation Tools:**
 - Used **Qlik ETL and Qlik NPrinting** to create scheduled report generation and distribution workflows.
 - Applied filters, dynamic date parameters, and user-specific access controls.
 - Created reusable **Power BI templates** with auto-refresh for real-time dashboards.
- **Testing & Deployment:** Validated outputs with stakeholders and documented all workflows for future scalability.

Result:

- Cut manual reporting time by over **70%**, saving hundreds of hours per quarter.
- Achieved **95%+ data accuracy** by removing human error from report generation.
- Improved stakeholder satisfaction through on-time delivery and enhanced visual insights.
- The automation system was later scaled for compliance, B2B partners, and senior leadership reporting — becoming a **core reporting framework**.

17. Tell me about a time when you had to learn a new data tool or technology quickly. How did you go about it, and how did it help your work?

Situation:

When I transitioned from a mechanical engineering background into a **Junior Data Analyst** role at **Nipa International** during the COVID-19 pandemic, I had no prior hands-on experience with analytics tools like SQL, Excel, or Tableau — yet was expected to deliver insights from **ERP manufacturing data** within weeks.

Task:

I needed to **quickly upskill in key analytics tools** to clean, analyse, and visualise production data — and deliver reports that could help improve efficiency and reduce costs in a factory setting.

Action:

- I dedicated the **first month to structured self-learning**, using tutorials, practice datasets, and documentation to build a foundational understanding of **MySQL, Excel, and Tableau**.
- Simultaneously, I applied what I learned on real tasks — such as:
 - Writing SQL queries to extract data from the ERP system.
 - Building Excel reports tracking **downtime, defect rates, and production efficiency**.
 - Creating basic Tableau dashboards for visualising trends.
- I frequently sought feedback from senior analysts and cross-verified my results with manual reports to build trust in my outputs.
- I also documented my processes and reused SQL snippets and Excel templates to speed up recurring analysis.

Result:

- I was able to reduce data processing time by **20%** and improve **report accuracy** by standardising queries and validations.
- Identified **raw material inefficiencies**, resulting in potential cost savings of ₹25 lakhs (~£24,000).
- The team began relying on my dashboards for daily performance monitoring — and my efforts helped me establish myself as a reliable contributor, despite coming from a non-technical background.

18. Can you give an example of a time when you identified a potential business risk using data analysis? How did you address it?

Situation:

At **Care Health Insurance**, I was reviewing the **Claims Pendency Report** as part of an ongoing audit. During this analysis, I noticed an unusually high number of **pending claims** — particularly in one region — with some cases aging beyond **30 days**, which posed a risk of regulatory non-compliance and customer dissatisfaction.

Task:

My responsibility was to **identify the root cause** of the backlog, quantify the impact, and propose a solution before it escalated into a compliance issue or led to financial penalties.

Action:

- I used **SQL** and **Power BI** to segment pending claims by **region, product type, and processing stage**.
- Discovered that over **45% of delayed claims** were stuck in the **manual verification stage**, largely due to missing documentation and inconsistent data entry.
- Flagged this as a **business risk** during a stakeholder review, especially because regulatory guidelines require settlement within a specific timeframe.
- I proposed a solution involving:
 - **Automation of document verification** for low-risk policy types.
 - Implementation of **structured dropdowns** in the CRM to standardize data capture.
 - A **triage workflow** to fast-track simple claims while allocating complex ones to specialist teams.

Result:

- The proposed changes were implemented, leading to a **40% reduction in claim processing time**.
- The backlog of high-risk pending claims fell by **55%** within three months.
- The region returned to **regulatory compliance**, avoiding potential penalties.
- The success of this initiative earned me a **Letter of Recommendation from the Head of Claims** and the new framework was scaled to other teams.

19. Have you ever run an A/B test? What was the hypothesis, how did you analyse the results, and what actions followed?

Situation:

At Care Health Insurance, I was supporting the **sales and marketing teams** in boosting **policy renewal rates**. One idea they had was to test whether **reminder timing** influenced customer response rates — specifically whether sending a reminder **7 days vs. 3 days** before policy expiry would result in better engagement.

Task:

I proposed and executed an **A/B test** to validate this idea, aiming to identify the optimal timing for renewal communication and improve customer retention.

Action:

- **Hypothesis:** Sending reminders **7 days before expiry** will lead to **higher renewal rates** than sending them 3 days before, as it gives customers more time to respond.
- **Test Design:**
 - Randomly segmented customers nearing policy expiry into **two groups**:
 - **Group A** received a reminder 7 days prior.
 - **Group B** received it 3 days prior.
 - Both groups were similar in policy type, demographics, and previous claim history to ensure fairness.
- **Data Collection & Analysis:**
 - Tracked **open rates**, **click-through rates**, and **actual renewal conversion** for both groups.
 - Used **SQL** to extract the campaign results, and ran statistical tests (Chi-square test) to determine significance.
 - Visualised the results using **Power BI** for stakeholder presentation.
- **Findings:**
 - Group A had a **22% higher renewal rate** and 18% more early conversions.
 - Group B had more late renewals and a higher no-response rate.

Result:

- The company adopted the **7-day reminder strategy** as standard practice.
- Overall renewal rates improved by **12% over the following quarter**.
- I was asked to contribute to future CRM testing strategies, including message format and channel preference (SMS vs. email).
- The test also laid the foundation for **personalised reminder timing** based on past customer behaviour.

20. Describe a situation where you worked with teams from different departments to complete a data-related project. What was the biggest challenge, and how did you handle it?

Situation:

At **Team Computers Pvt Ltd**, while working with a **healthcare insurance client**, I led a project to improve **claims processing efficiency**. This required collaboration between the **Claims**, **Medical**, and **CRM teams**, all of whom had different workflows, terminologies, and data priorities.

Task:

The objective was to **eliminate manual data entry errors** that were causing reporting delays and claims disputes. We wanted to introduce **standardised dropdown fields** across systems to replace inconsistent free-text inputs.

Action:

- Held **requirement gathering sessions** with each team to identify common pain points, like inconsistent provider names or vague claim statuses.
- Collaborated with the **Medical team** to define standard terminology for diagnoses and procedures.
- Worked with the **CRM team** to implement dropdowns for customer interaction logs and policy statuses.
- Facilitated multiple **UAT (User Acceptance Testing)** sessions to get early feedback and adjust dropdown lists based on real-world usage.
- Created **training documentation** and conducted live walkthroughs to ensure all teams adopted the changes smoothly.

Biggest Challenge:

The biggest hurdle was **resistance to change** — especially from teams used to flexible, open-text entry. Some feared dropdowns would slow them down or limit their ability to describe edge cases.

How I Handled It:

- I addressed this through **empathy and transparency**, explaining how standardisation would actually reduce rework, audit flags, and delays.
- Highlighted **real data examples** showing error reduction potential, and worked with the CRM lead to add an “Other – please specify” fallback option to ease the transition.
- Maintained regular feedback loops and made iterative improvements based on frontline user input.

Result:

- **Data accuracy improved by over 50%**, and claims processing speed increased by 30%.
- Enabled faster, more reliable analytics by eliminating inconsistencies in key fields.
- The dropdown system was rolled out across multiple departments and cited as a model for future data governance initiatives.

21. Can you tell us about a time you received negative feedback on a Power BI report, or when a report wasn't well used after launch.

Situation:

While working for Care Health Insurance, I created a sales and CRM performance dashboard in Power BI (and Qlik Sense for some teams). After launch, I noticed that adoption was very low, the sales and CRM teams, who were mainly non-technical users, found the dashboard difficult to use. Instead, they started asking me to send manual Excel and PDF reports every week, which defeated the purpose of automation.

Task:

My goal was to understand why the dashboard wasn't being used and then redesign it to truly meet their needs.

Action:

When I conducted feedback sessions, several issues became clear:

1. There were no clear titles or subtitles, so users couldn't immediately tell what each chart represented.
2. The dashboard used area charts and scatter plots that looked nice visually but weren't meaningful for these users, for example, a scatter chart plotting "Sales vs Lead Response Time" didn't add value because most people just wanted to know "Which agent is meeting their targets?"
3. Formatting was inconsistent, fonts, colours, and alignment varied across visuals, which made it look cluttered.
4. Performance was poor, with long load times due to unnecessary visuals and too much data on a single page.
5. Slicers were placed randomly, and many fields were irrelevant to the users' day-to-day decisions.
6. Most importantly, there were too many charts on one page, around 12 visuals competing for attention, leaving users confused rather than informed.

So, I decided to completely redesign the dashboard from the user's perspective.

I grouped visuals logically into three pages:

1. Sales Summary: KPI tiles for total policies sold, conversion rate, and sales vs target. A clustered bar chart showed performance by region and agent, while a line chart tracked monthly trends in sales and renewals.
2. CRM Activity Overview: I replaced the scatter chart with a funnel visual to show lead progression, from new leads → contacted → converted. I also added a heatmap for call volumes and a table with conditional formatting to highlight overdue follow-ups.
3. Customer Insights: Used a simple donut chart to show churn vs retention rate and a stacked bar chart for customer feedback distribution (positive, neutral, negative).

I also:

1. Added clear titles and subtitles to every visual (e.g., "Monthly Sales Trend", "Agent Performance by Conversion Rate")
2. Standardised colours, blue for active, red for risk, green for achieved.
3. Placed slicers neatly in a left-hand filter pane with intuitive labels like Region, Product, and Month.
4. Optimised DAX measures and disabled unnecessary visuals to reduce page load time by 40%.

Result:

After relaunch, I conducted a short walkthrough with both the Sales and CRM teams, showing how they could use filters, export data, and track progress without manual reports.

The result was a complete turnaround; dashboard usage increased significantly, manual reporting requests dropped by more than 80%, and the sales managers started using the dashboard in their weekly performance meetings.

It also became the base template for other departmental dashboards because it combined clarity, performance, and visual storytelling.

Project:

1. Project: Claims Pendency Report – Healthcare Insurance Client

Introduction

This project focused on converting a fragmented, manual claims pendency reporting process into a unified and automated system. The goal was to consolidate claims data across multiple departments, correct inaccurate ageing logic, and provide reliable insights to reduce backlog and improve operational efficiency.

Key Contributions

- Consolidated pendency data from multiple claims functions including TPA, Cashless, Reimbursement, Medical, Travel, and Critical Illness into a single dataset.
- Standardised claim ageing logic and defined business-aligned ageing buckets to accurately identify high-risk and long-pending claims.
- Built automated SQL-based data pipelines to replace manual Excel reporting.
- Cleaned and transformed ambiguous and inconsistent data by removing duplicates, correcting invalid records, and resolving missing values using master data.
- Designed daily pendency reports and monthly productivity views showing pending vs processed claims by department.
- Collaborated closely with IT, Claims Heads, Medical teams, and TPA managers to validate data, conduct UAT, and refine reporting logic.

Results

- Reduced average daily pending claims from ~80,000 to ~38,000 over 20 months.
- Improved claims processing productivity by approximately 52%.
- Significantly reduced manual effort through automated daily and monthly reporting.
- Enabled teams to prioritise aged and high-risk claims, improving turnaround times.
- Established a single source of truth for claims pendency metrics across the organisation.

Competencies

- Handling ambiguous and messy data
- Data cleaning, transformation, and feature engineering
- End-to-end project ownership
- SQL-based automation and reporting
- Cross-functional collaboration with business and technical teams
- Translating business problems into data-driven solutions

Link: https://vyom20798.github.io/VyomKhanna_Portfolio/#Claim_Pendency

2. Project: Monthly Travel Insurance Dashboard – Stakeholder Reporting

Introduction

This project involved building a monthly business intelligence dashboard for travel insurance stakeholders to track sales performance, premium trends, policy renewals, and claims processing efficiency. The objective was to provide a single, interactive view of key metrics to support data-driven business and operational decisions.

Key Contributions

1. Collected and integrated sales, renewal, premium, and claims data from multiple systems using SQL.
2. Performed exploratory data analysis to identify trends in sales channels, renewal behaviour, and claims performance.
3. Engineered key business metrics such as renewal rate, premium per policy, new policy rate, and claim settlement time.
4. Designed an interactive dashboard presenting sales, premium, renewal, and claims insights in a stakeholder-friendly format.
5. Segmented performance by sales channel (Online, Agent, Corporate), region, product type, and policy lifecycle (New vs Renewal).
6. Communicated insights clearly to business stakeholders, translating data findings into actionable recommendations.

Results

1. Delivered a consolidated monthly dashboard, eliminating dependence on multiple disconnected reports.
2. Helped identify sales and renewal trends, contributing to a 15% increase in customer retention.
3. Enabled better tracking of premium growth by product, supporting pricing and sales strategy decisions.
4. Reduced average claim settlement time by 20% by highlighting bottlenecks and pending claim backlogs.
5. Improved stakeholder visibility into profitability by linking claim payouts with premium revenue.

Competencies

1. Stakeholder communication and requirement gathering
2. Business intelligence and dashboard design
3. Exploratory data analysis and trend identification
4. SQL-based data extraction and transformation
5. Translating complex data into clear business insights

Link: https://vyom20798.github.io/VyomKhanna_Portfolio/#Travel_Dashboard

3. Project: Claims Summary & Compliance Reporting – B2B Partners and Brokers

Introduction

This project focused on building an automated and privacy-compliant claims reporting solution for external B2B partners and insurance brokers. The objective was to provide transparent claims performance insights while strictly adhering to IRDAI regulations by removing all personal and sensitive health information.

Key Contributions

- Designed a compliant data-filtering framework to exclude all PII and PHI, retaining only anonymised insured member identifiers.
- Processed group policy claims data to generate structured summary reports and detailed raw data extracts.
- Built key claims metrics covering volumes, financials, settlement performance, rejection rates, fraud indicators, and ageing analysis.
- Developed automated batch-based pipelines to transform raw claims data into business-ready tables and KPI summaries.
- Generated PDF summary reports along with Excel data dumps to support audits, validation, and deeper analysis.
- Enabled secure report distribution through an external partner portal with controlled access rights.

Results

- Delivered fully IRDAI-compliant claims reports, ensuring data privacy while improving transparency for external stakeholders.
- Eliminated manual reporting effort through automation, significantly reducing turnaround time and errors.
- Improved operational efficiency by enabling partners to track settlement progress, ageing, and bottlenecks independently.
- Strengthened broker and partner relationships through consistent, accurate, and transparent reporting.
- Established a scalable reporting framework supporting both regulatory compliance and business performance monitoring.

Competencies

- Automation of repetitive reporting tasks
- Data privacy, compliance, and governance
- Claims data transformation and KPI development
- Process efficiency and operational improvement
- Secure data delivery and stakeholder enablement

Link: https://vyom20798.github.io/VyomKhanna_Portfolio/#B2b_and_Internal_Reportin

4. Project: Daily Business Reporting – Executive Dashboard for MD & Department Heads

Introduction

This project involved designing and implementing an enterprise-wide Daily and Month-To-Date (MTD) business performance reporting system for senior leadership, including the Managing Director and Heads of Claims, Finance, Sales, and Actuarial. The objective was to provide unified, real-time visibility into organisational performance using consistent KPIs and a shared data foundation.

Key Contributions

- Consolidated data from Claims, Finance, Sales, CRM, and Actuarial systems into a single reporting framework.
- Designed a structured data architecture with staging, transformation, and presentation layers to ensure data accuracy and scalability.
- Standardised key business KPIs such as Claim Ratio, Loss Ratio, Persistency, Lapse Rate, Expense Ratio, Sales Conversion, and Profitability.
- Built complex SQL pipelines and modular KPI logic to support daily and MTD reporting without manual intervention.
- Developed executive dashboards covering Claims performance, Financial overview, Sales performance, and Actuarial risk insights.
- Created dashboards in both Excel (print-ready) and Power BI (interactive) for broad stakeholder accessibility.
- Worked closely with multiple department heads to validate metrics, conduct UAT, and align reporting definitions across teams.

Results

- Delivered real-time, executive-ready visibility into business performance across all major departments.
- Enabled early identification of issues such as rising claim ratios, declining sales conversion, and product underperformance.
- Eliminated multiple manual Excel reports, significantly reducing reporting time and error rates.
- Established a single source of truth, improving cross-departmental alignment and decision-making.
- Built a scalable BI foundation that supports future dashboards, forecasting, and advanced analytics.

Competencies

- KPI-driven dashboard design and performance monitoring
- Advanced SQL querying and data modelling
- Business intelligence and data visualisation (Excel, Power BI)
- Handling large datasets efficiently
- Cross-functional stakeholder collaboration and communication

Link: https://vyom20798.github.io/VyomKhanna_Portfolio/#MD_Reports

5. Company Project 5: CRM Dashboard – Customer Satisfaction & Service KPIs

Introduction

This project involved designing and delivering an interactive CRM dashboard to monitor customer satisfaction, service performance, and retention trends. The objective was to provide leadership and operational teams with actionable insights to improve customer experience, reduce churn, and strengthen renewals.

Key Contributions

- Integrated customer, claims, service, and sales engagement data using SQL and Qlik ETL into a unified CRM data model.
- Analysed NPS trends, churn drivers, service inefficiencies, and renewal behaviour impacting customer satisfaction.
- Engineered key service and retention metrics such as NPS distribution, churn rate, retention rate, resolution time, and escalation percentage.
- Developed interactive dashboards in Power BI / Qlik Sense covering customer satisfaction, service KPIs, retention, and sales engagement.
- Implemented automated data refreshes and role-based access to ensure secure and reliable reporting for stakeholders.

Results

- Identified key churn drivers, including premium hikes, poor renewal reminders, and unresolved claims.
- Enabled process improvements such as automated renewal reminders and better claims communication.
- Increased customer retention by 15–20% within three months.
- Improved service efficiency by providing real-time visibility into escalations and resolution performance.
- Reduced ad-hoc CRM reporting by 25% through standardised KPIs and dashboards.

Competencies

- CRM analytics and customer journey analysis
- Dashboard design for customer satisfaction and service KPIs
- SQL, ETL, and BI tools (Power BI, Qlik Sense)
- Churn, retention, and NPS analysis
- Driving business impact through data insights

6. Company Project 6: Repeated Claims Detection – Fraud Prevention

Introduction

This project focused on detecting and preventing fraudulent health insurance claims by identifying repeated claims submitted by the same insured member for the same date of admission. The solution combined automated data checks with a predictive fraud detection model to proactively flag high-risk claims.

Key Contributions

- Extracted and analysed Month-to-Date claims data from Snowflake to identify duplicate and repeated claims.
- Designed a composite key using Insured Member ID and Date of Admission to detect repeated submissions.
- Built SQL-based duplicate detection logic with claim status prioritisation to retain valid records and flag fraudulent entries.
- Created automated exception reports for Medical and Payments teams to review flagged claims daily.
- Performed exploratory data analysis and feature engineering to identify fraud indicators.
- Developed and evaluated classification models, selecting Random Forest as the final model based on performance.
- Integrated fraud model outputs into dashboards and daily fraud risk reports for operational teams.

Results

- Prevented repeated and duplicate claims from being processed, reducing potential fraudulent payouts.
- Reduced manual duplicate claim checks by over 60% for Medical and Payments teams.
- Achieved ~80% fraud detection accuracy with a model AUC of 0.87.
- Improved compliance, audit readiness, and trust through stronger fraud controls.

Competencies

- Fraud analytics and anomaly detection
- SQL-based automation and Snowflake data processing
- Predictive modelling (Python, Pandas, Machine Learning)
- Feature engineering and model evaluation
- Cross-team integration with Medical and Payments functions

7. Company Project 7: Sales Incentive Dashboard

Introduction

This project involved building an automated, role-based sales incentive dashboard for Frontline Sales Officers (FSOs), Area Managers, and Zonal Managers. The objective was to provide real-time visibility into Target vs Actual performance, ensure accurate incentive payouts, and enable timely coaching to consistently exceed monthly and quarterly targets.

Key Contributions

- Integrated daily sales, renewals, and HR payroll data using SQL and Snowflake.
- Loaded and managed fixed monthly and quarterly targets by role and hierarchy.
- Defined KPI frameworks and tiered incentive logic for FSOs, Area Managers, and Zonal Managers.
- Built DAX measures for Target vs Actual, percentage achievement, and incentive tiers (Bronze/Silver/Gold).
- Designed interactive dashboards with role-based access, drilldowns from Zone → Area → FSO, and performance leaderboards.
- Automated data refreshes and generated PDF snapshots for monthly and quarterly performance and payout reviews.
- Supported adoption through sales and HR training, SOP documentation, and payout audit trails.

Results

- Achieved 100% accuracy in incentive payouts and reduced payroll preparation time from weeks to days.
- Enabled managers to provide mid-cycle coaching using real-time performance and pipeline visibility.
- Helped sales teams exceed fixed monthly and quarterly targets by ~20% in the following quarter.
- Sustained double-digit performance uplift over six months through transparent tracking and aligned incentives.

Competencies

- Sales performance and incentive analytics
- Target vs Actual KPI design and monitoring
- SQL, Snowflake, and DAX-based data modelling
- Role-based dashboards and access control
- Driving behavioural change through data transparency

8. Company Project 8: ETL Pipeline Optimisation

Introduction

This project focused on optimising the existing enterprise data warehouse and BI ecosystem by improving Qlik ETL and DBT pipelines. The objective was to reduce end-to-end data refresh time, improve dashboard performance, retire underused reports, and ensure accurate and consistent **T-1 (Today - 1)** reporting across Claims, Sales, Finance, and Product teams.

Key Contributions

- Profiled 40+ fact and dimension tables using SQL and Qlik ETL logs to identify redundant joins, hierarchies, and performance bottlenecks.
- Implemented **incremental extraction logic at Stage 1** using EXISTS / NOT EXISTS conditions on frequently changing fields to avoid full reloads and reduce extraction time.
- Analysed Qlik reload logs to identify heavy tables and built **parallel extraction Qlik apps** so large datasets could load independently without blocking other pipelines.
- Defined a clear **fact-level grain** to standardise joins and KPI calculations across datasets.
- Consolidated repeated transformations from multiple Qlik applications into a **single upstream transformation layer**, reducing duplication and improving consistency.
- Shifted selected KPI and derived calculations from the front-end into **backend Python and Qlik scripts** to improve performance and reduce dashboard load time.
- Reviewed and decommissioned **underused or redundant dashboards** to reduce refresh overhead and compute usage.
- Designed **simple, lightweight dashboard visuals** (KPI cards, basic bar and line charts, table/matrix views) using a light colour scheme with no conditional formatting.
- Implemented **parallel dimension build tasks** using Qlik QMC scheduling.
- Introduced and enforced **T-1 reporting logic**, ensuring all dashboards used validated and complete data.
- Cross-validated KPIs and report numbers between **Production and Replica environments** before release.
- Implemented a **file-based trigger mechanism** in ETL workflows to control job dependencies and prevent race conditions across multi-source pipelines.
- Documented ETL lineage, KPI logic, and validation checks, and trained BI and data engineering teams on monitoring and governance practices.

Results

- Reduced ETL runtime from ~8 hours to **4 hours 45 minutes** (~40% improvement).
- Improved dashboard refresh times by **66%** (from ~45 minutes to ~15 minutes).
- Achieved **95%+ data accuracy** across all published reports using T-1 logic.
- Eliminated manual job restarts through automated sequencing and dependency control.
- Enabled early detection and prevention of pipeline failures through log analysis and audit dashboards.

Competencies

- ETL pipeline optimisation and performance tuning
- Qlik ETL, SQL, DBT, and backend transformation logic
- Incremental loading and dependency management
- BI performance optimisation and dashboard design
- Data accuracy, validation, and governance
- Cross-functional collaboration with IT, BI, and Operations teams

9. Company Project 9: Policy Renewal Analysis and Customer Retention Dashboard

Introduction

This project focused on analysing health insurance policy renewals to identify key reasons for customer drop-offs and low retention. The objective was to convert complex analytical findings into clear, actionable insights for non-technical sales and customer service teams to improve renewal outcomes.

Key Contributions

- Analysed policy renewal patterns to identify behavioural drivers behind renewals and non-renewals.
- Integrated policy, claims, premium, and CRM data into a unified analytical dataset.
- Engineered features such as Renewal Status, Claim History, and Premium Change % to support retention analysis.
- Designed an interactive Power BI dashboard with simple visuals and business-friendly explanations for frontline teams.

Dashboard Charts & What They Showed

- Renewal vs Non-Renewal Bar Chart: Compared the count and percentage of renewed vs non-renewed policies.
- Top Reasons for Non-Renewal (Bar/Pie Chart): Highlighted major drop-off drivers such as premium hikes (40%), poor communication (30%), and claim disputes (20%).
- Premium Change vs Renewal Rate Chart: Showed how higher premium increases correlated with lower renewal likelihood.
- Claim Experience Comparison Chart: Compared renewal rates for customers with claims vs those without claims.
- Trend Line – Renewal Rate Over Time: Tracked monthly changes in renewal performance to assess improvement or decline.
- Customer Segmentation Table: Broke down renewal behaviour by region, product type, and customer segment for targeted action.

Results

- Increased policy renewal rates by 20% within three months of implementation.
- Enabled sales and service teams to prioritise high-risk customers using clear visual cues.
- Improved customer engagement through automated reminders, loyalty incentives, and better claim communication.
- Received leadership recognition for effective data storytelling and stakeholder communication.

Competencies

- Translating complex analysis into simple business visuals
- Customer behaviour and retention analytics
- Power BI dashboarding and data storytelling
- Influencing business decisions through insights
- Communicating effectively with non-technical stakeholders

10. Company Project 10: Automated Reporting Optimisation & Stakeholder Communication

Introduction

This project focused on resolving technical limitations in automated financial reporting while improving data accessibility for the Finance team. The objective was to design a secure and scalable reporting solution, manage stakeholder expectations effectively, and address challenges related to large report distribution.

Key Contributions

- Analysed report size constraints and system limitations impacting automated email-based delivery.
- Optimised financial data extracts by removing redundant fields and segmenting large datasets into modular outputs.
- Designed alternative report distribution methods using Power BI dashboards and secure shared drives.
- Automated uploads to SharePoint / Google Drive to enable secure, controlled stakeholder access.
- Clearly communicated technical limitations to stakeholders and proposed practical, business-friendly alternatives.
- Balanced stakeholder expectations by setting boundaries while ensuring reporting needs were met effectively.

Results

- Eliminated recurring failures in email-based report delivery by shifting to secure automated sharing mechanisms.
- Improved reporting efficiency and real-time data accessibility for Finance and audit teams.
- Increased stakeholder satisfaction through transparent communication and reliable reporting solutions.
- Implemented a scalable reporting framework suitable for large datasets and future growth.

Competencies

- Stakeholder expectation management and communication
- Problem-solving under technical constraints
- Automated reporting and process optimisation
- Secure data sharing and governance
- Confident decision-making and saying “no” with data-backed alternatives

[11. Company Project 11: Power BI Adoption & Stakeholder Enablement](#)

Introduction

This project focused on transitioning business users from Excel-based reporting to Power BI. The objective was to improve analytics maturity, enhance data accessibility, and build stakeholder confidence through intuitive dashboard design and hands-on enablement.

Key Contributions

- Analysed existing Excel reporting workflows to identify pain points and barriers to Power BI adoption.
- Rebuilt key datasets and KPIs in Power BI using structured queries and Power Query transformations.
- Designed user-friendly dashboards with familiar Excel-style layouts, matrix views, slicers, and summary cards.
- Standardised visual design elements to ensure consistency, clarity, and ease of navigation.
- Developed user guides and conducted live training sessions to explain filters, drilldowns, and export features.
- Supported change management by demonstrating tangible benefits and building trust in Power BI outputs.

Results

- Successfully transitioned stakeholders from Excel reports to Power BI dashboards.
- Improved reporting accuracy, usability, and user confidence through intuitive dashboard design.
- Increased organisation-wide data literacy and self-service analytics adoption.
- Recognised for simplifying BI adoption and improving cross-departmental reporting efficiency.

Competencies

- BI adoption and change management
- Power BI dashboard design and user experience
- Stakeholder enablement and training
- Translating user needs into intuitive analytics solutions
- Driving analytics maturity across teams

12. Company Project 12: Power BI Dashboard Redesign Based on User Feedback

Introduction

This project focused on redesigning a poorly adopted Power BI dashboard that had received negative user feedback. The objective was to improve clarity, performance, and business relevance by addressing usability issues and aligning the dashboard with stakeholder decision-making needs.

Key Contributions

- Conducted structured feedback sessions to identify root causes of low adoption, including visual clutter, inconsistent formatting, and slow performance.
- Reviewed and optimised existing SQL queries, Power Query steps, and data models to improve refresh efficiency.
- Standardised KPI definitions and ensured alignment with business requirements.
- Redesigned the dashboard using a clear visual hierarchy, consistent formatting, and brand-aligned colour schemes.
- Reorganised slicers and visuals to improve navigation and reduce cognitive load.
- Added high-value KPIs such as monthly sales trends, top regions, and top agents.
- Led UAT sessions and iteratively refined the dashboard based on stakeholder feedback.

Results

- Improved dashboard performance and report refresh speed.
- Significantly increased user adoption for daily monitoring and operational reviews.
- Enhanced decision-making by presenting relevant KPIs in a clean, intuitive format.
- Recognised for transforming a low-impact dashboard into a high-value analytical asset.

Competencies

- Dashboard redesign and UX optimisation
- Stakeholder-centric BI development
- Power BI performance tuning
- Translating feedback into actionable design improvements
- Driving adoption through user-focused analytics

13. Company Project 13: Multi-Project Management

Introduction

This project involved managing multiple high-priority business intelligence initiatives simultaneously under tight deadlines. The objective was to ensure on-time delivery of critical reports, streamline ETL processes, and maintain clear stakeholder communication to support timely decision-making.

Key Contributions

- Managed and delivered three BI initiatives in parallel: Sales Executive Report, Claims Report Automation, and Commission Report Optimisation.
- Analysed cross-departmental data dependencies across Sales, Claims, and Finance to prioritise workflows effectively.
- Built and automated SQL-based ETL pipelines to ensure consistent and timely data refreshes.
- Developed reusable data models and dynamic Power BI filters to reduce report development time.
- Implemented validation checks to maintain data consistency across multiple outputs.
- Created structured task plans and maintained transparent communication with stakeholders on progress and constraints.

Results

- Successfully delivered all three BI projects on time despite tight timelines.
- Improved reporting efficiency and visibility across Sales, Claims, and Finance teams.
- Reduced redundant data processing efforts through reusable models and automation.
- Recognised for strong prioritisation, time management, and stakeholder coordination.

Competencies

- Managing multiple projects under tight deadlines
- BI project planning and prioritisation
- SQL, ETL automation, and Power BI development
- Cross-functional stakeholder coordination
- Time management and delivery execution

14. Academic Project 1: Predicting London Bike Usage Using Machine Learning (Python)

Introduction

This project focused on analysing and forecasting bike usage patterns in London's bike-sharing system to support better operational planning. The objective was to combine trip data with weather information and apply machine learning models to predict demand and optimise bike availability.

Key Contributions

- Collected and analysed bike trip data from Transport for London (TfL) and weather data from Visual Crossing.
- Cleaned and preprocessed data by removing outliers and invalid trip durations.
- Engineered time-based and weather-related features such as day of week, hour of day, trip duration, and temperature.
- Normalised numerical features using MinMaxScaler.
- Built and evaluated multiple machine learning models including Linear Regression, Decision Trees, Random Forest, SVM, Neural Networks, and XGBoost.
- Assessed model performance using MSE, MAE, and R^2 metrics.

Results

- Identified clear usage patterns, with peak demand in weekday afternoons, especially Thursdays and Fridays.
- Found Hyde Park Corner to be the most active station during the study period.
- Determined that temperature was the strongest predictor of bike usage.
- Achieved the best predictive performance using Linear Regression, followed by Decision Trees and Random Forest.
- Recommended using weather forecasts to proactively allocate bikes to high-demand stations.

Competencies

- Machine learning and predictive analytics
- Python (Pandas, Scikit-learn)
- Feature engineering and data preprocessing
- Model evaluation and comparison
- Translating analytical findings into operational recommendations

Dashboard Design

How I Approach Dashboard Design in Power BI

When I design a Power BI dashboard, I think of it as a **storytelling tool**, not just a data display. My goal is to make insights clear, consistent, and immediately actionable for the user. Here's how I approach it step by step:

1 Understanding the Business Context

Before opening Power BI, I always clarify:

- What are the **key business questions** we're trying to answer?
- Who will use this dashboard — executives, analysts, or operational teams?
- What's the expected frequency — daily, monthly, or ad hoc?

This determines everything that follows — from layout to level of detail.

2 Data Modelling & DAX Design

Once the data sources are ready, I:

- Build a **star schema** in Power BI: fact tables for transactions (e.g., claims, sales) and dimension tables (e.g., product, region, date).
 - Avoid snowflake models for performance reasons and keep **one-to-many relationships** clean.
 - Create **measures using DAX** rather than calculated columns for efficiency.
-

3 Page Layout and Visual Hierarchy

For page design, I follow the **Z-pattern or F-pattern layout** — users naturally read left to right, top to bottom.

A typical Power BI page:

- **Top Section (Header):**
 - Dashboard title
 - Subtitle / last refresh date
 - Filter context (e.g., “Region: All | Month: September”)
 - Key KPI cards for instant insights (like claim ratio, revenue, churn rate)
 - **Middle Section (Trends and Breakdown):**
 - Time-series line charts for trends
 - Bar or column charts for category breakdowns
 - KPI comparisons vs. target or previous period
 - **Bottom Section (Details or Deep Dives):**
 - Tables or matrix visuals for drill-down details
 - Slicers or filter panels (either top ribbon or hidden side panel for clean design)
-

4 Visual Design & Chart Selection

I use **minimal colours and consistent formatting**:

- Limit colour palette to **2–3 core colours** aligned with company branding.
- Reserve **red/green** for performance indicators only.
- Always add **titles, subtitles, and labels** that explain the visual — not just what, but so *what* (e.g., “Claims Pending by Department – shows ageing trends across units”).

Common visuals and reasoning:

- **Bar/Column charts** – compare categories.
- **Line charts** – trends over time.
- **Pie/Donut charts** – limited use for composition (max 4–5 slices).
- **Cards/KPIs** – highlight key figures (with % change from last period).
- **Heatmaps or Tree maps** – distribution or hierarchical data.

- **Matrix tables** – for users who still like tabular detail (Excel transition).
-

5 Interactivity & Usability

I ensure dashboards are **interactive but not overwhelming**:

- Use slicers for time, region, and product.
 - Add **drill-through pages** so users can click a KPI and see detail by team or region.
 - Use **tooltips** for extra insights without cluttering visuals.
 - Implement **bookmarks and buttons** for navigation — e.g., “Back to Summary,” “Department View.”
-

6 Report Optimisation & Performance

I follow performance best practices:

- Use **aggregations and summarised tables** instead of raw-level visuals.
 - Minimise calculated columns in Power BI; push logic upstream to SQL or ETL.
 - Use **variables in DAX** for readability and speed.
 - Disable unnecessary interactions and reduce visuals per page (under 8–10).
 - Keep file size small (<100MB ideally) by using DirectQuery or incremental refresh if data is large.
-

7 Testing and Feedback

Before release, I:

- Validate KPIs with business users and compare against legacy reports (Excel or Qlik).
 - Conduct **UAT sessions** to check if the visuals match their expectations.
 - Adjust design based on feedback — often simplifying the visuals or renaming metrics for clarity.
 - Add a **data refresh timestamp** and notes on data source reliability for transparency.
-

8 Visual Consistency & Branding

- Apply a **corporate theme JSON** for fonts, colours, and alignment.
 - Use **consistent fonts** (Segoe UI, Arial, or company-approved font).
 - Align visuals precisely; leave adequate white space to reduce cognitive load.
 - Maintain uniform KPI tiles and icons for readability.
-

9 Documentation & Handover

Finally, I prepare:

- A **Data Dictionary** (source tables, joins, measures).
- A **Dashboard User Guide** with screenshots and instructions.
- Notes for refresh schedules, permissions, and known data limitations.