IBM Data Analysis Capstone Project

Kyle McCallum
11/21/2025

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Conclusion

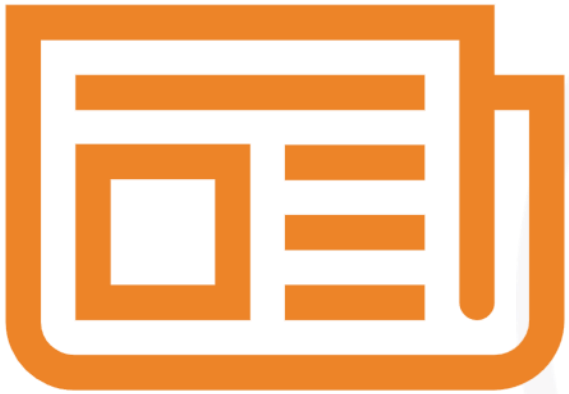
EXECUTIVE SUMMARY

- **Purpose:**
  - Analyze Stack Overflow Developer Survey to understand trends
- **Method**: Built three interactive dashboards covering current tech usage, emerging trends, and demographic distribution
  - Included bar charts, tree maps, bubble charts, and geospatial mapping
- **Current Technology Trends:**
  - JavaScript, SQL, and TypeScript dominate programming languages.
  - PostgreSQL, MySQL, and SQLite lead in database usage.
  - AWS and Google Cloud are the most widely used platforms.
- **Future Technology Trends:**
  - Growing interest in TypeScript, Go, and Rust for next year.
  - Redis, PostgreSQL, and cloud platforms such as AWS show rising demand.
  - Web frameworks like React and FastAPI gain traction.
- **Demographics:**
  - Majority of respondents are aged 25–34 with Bachelor's or Master's degrees.
  - Participation spans global regions, with concentration in North America, Europe, and India.

INTRODUCTION

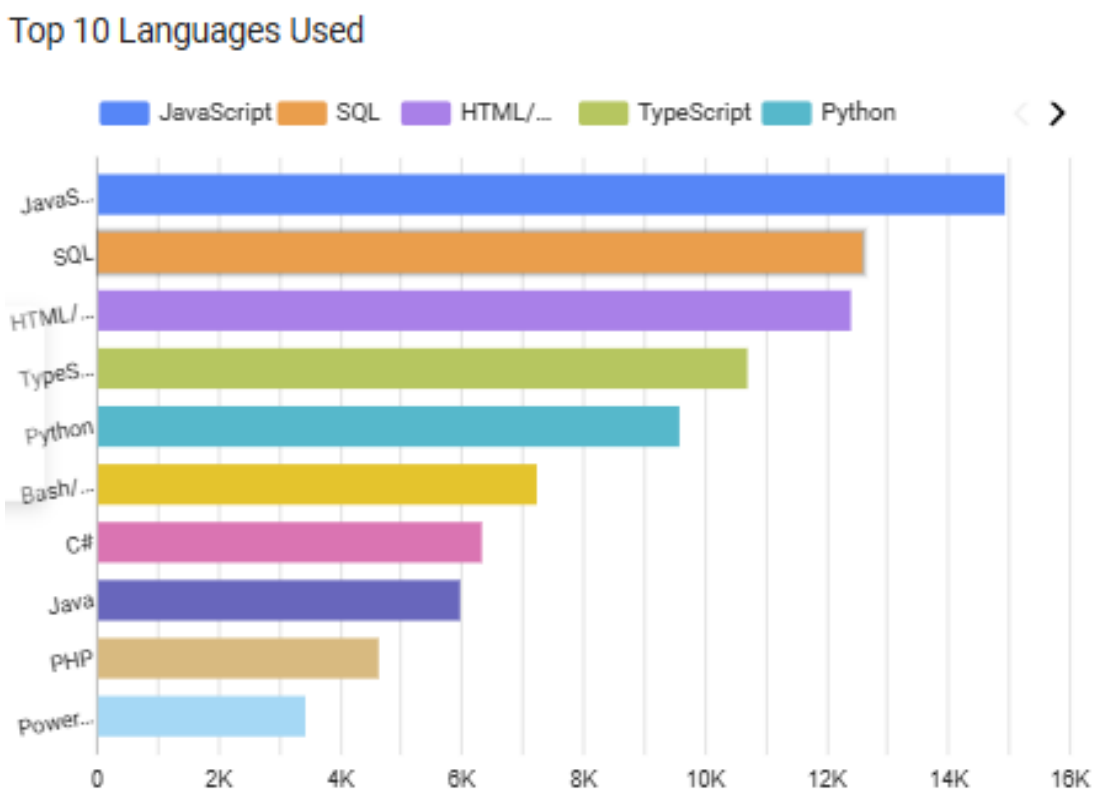
- **Target Audience:**
  - Data professionals, analysts, and hiring managers
  - Educators, curriculum designers, and tech leaders planning training or adoption strategies
  - Organizations interested in workforce skills and technology adoption
- **Purpose of the Project:**
  - Analyze the Stack Overflow Developer Survey
  - Identify current technology usage patterns and emerging future trends
  - Understand developer demographics that shape these trends
- **Value of the Analysis:**
  - Highlights the most in-demand programming languages, databases, platforms, and frameworks
  - Reveals which technologies developers plan to learn next
  - Provides demographic context for interpreting technology preferences
  - Supports data-driven decisions for skill development, hiring, and technology roadmap planning

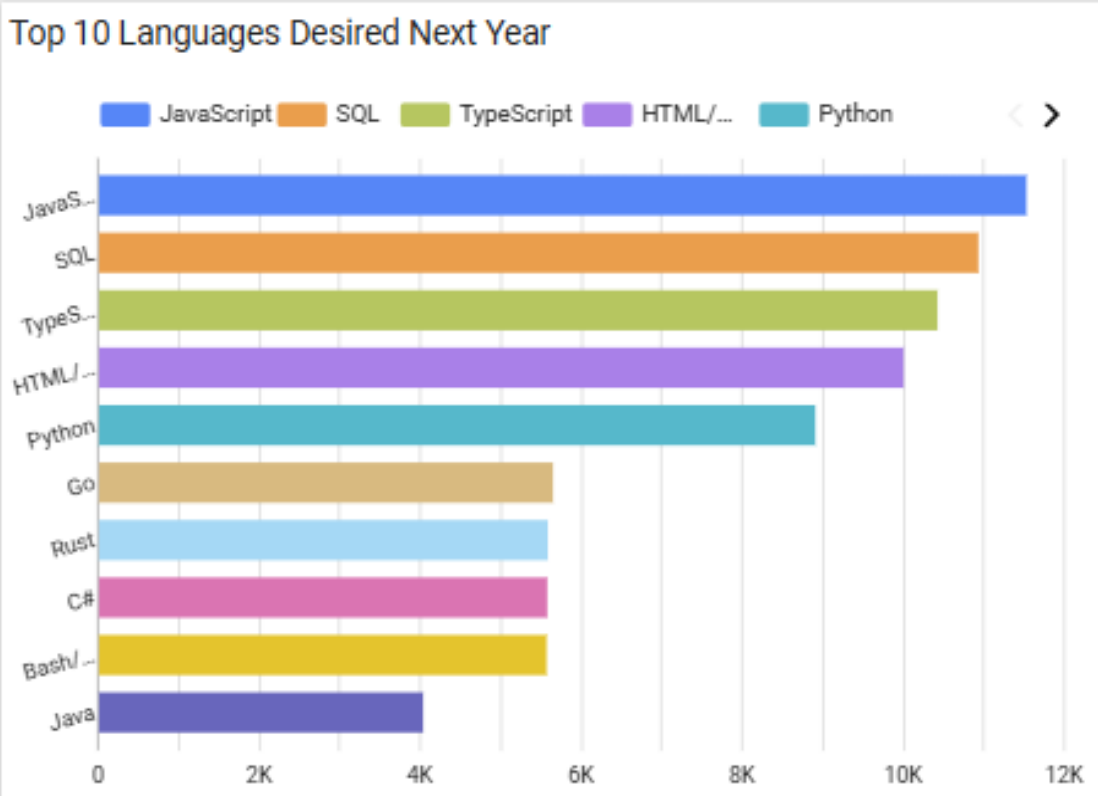
METHODOLOGY



- **Data Source:**
  - Stack Overflow Developer Survey (subset provided for the capstone)
  - Cleaned and prepared using Python (pandas)
- **Data Preparation & Wrangling:**
  - Removed or standardized long categorical labels
  - Aggregated top 10 technologies across languages, databases, platforms, and frameworks
  - Handled missing values and converted data types for visualization compatibility
- **Analytical Tools:**
  - Python (Pandas, Regex)
  - Google Looker Studio for interactive dashboard creation
- **Dashboard Development:**
  - Built three dashboards:
    - Current Technology Usage
    - Future Technology Trends
    - Demographics

PROGRAMMING LANGUAGE TRENDS

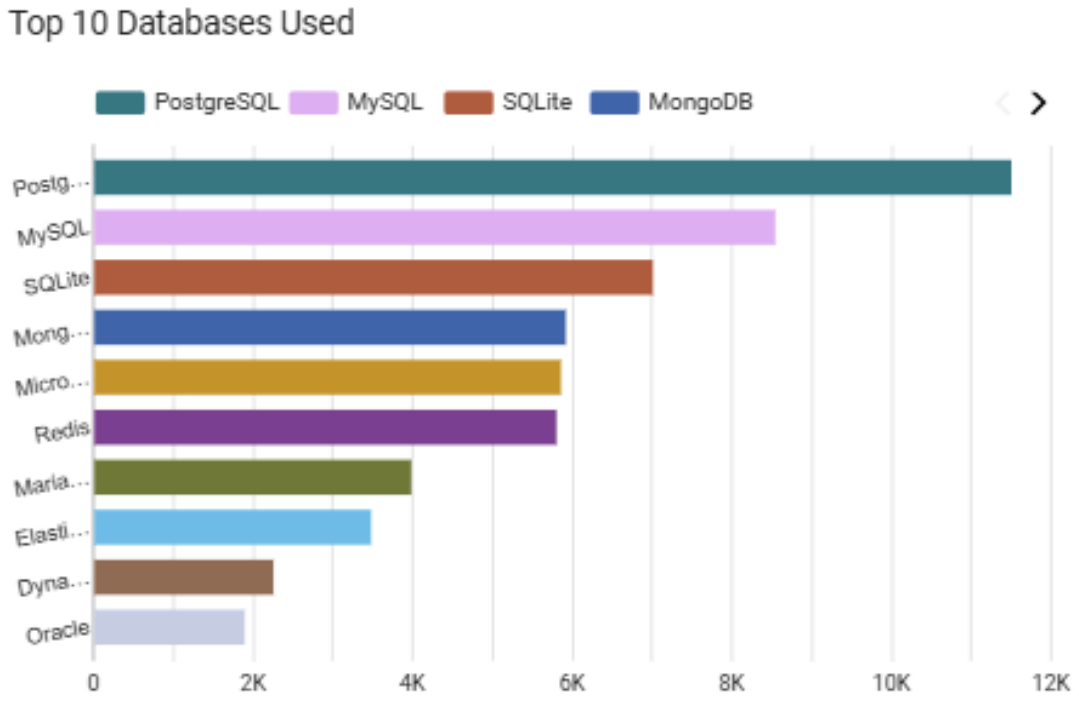


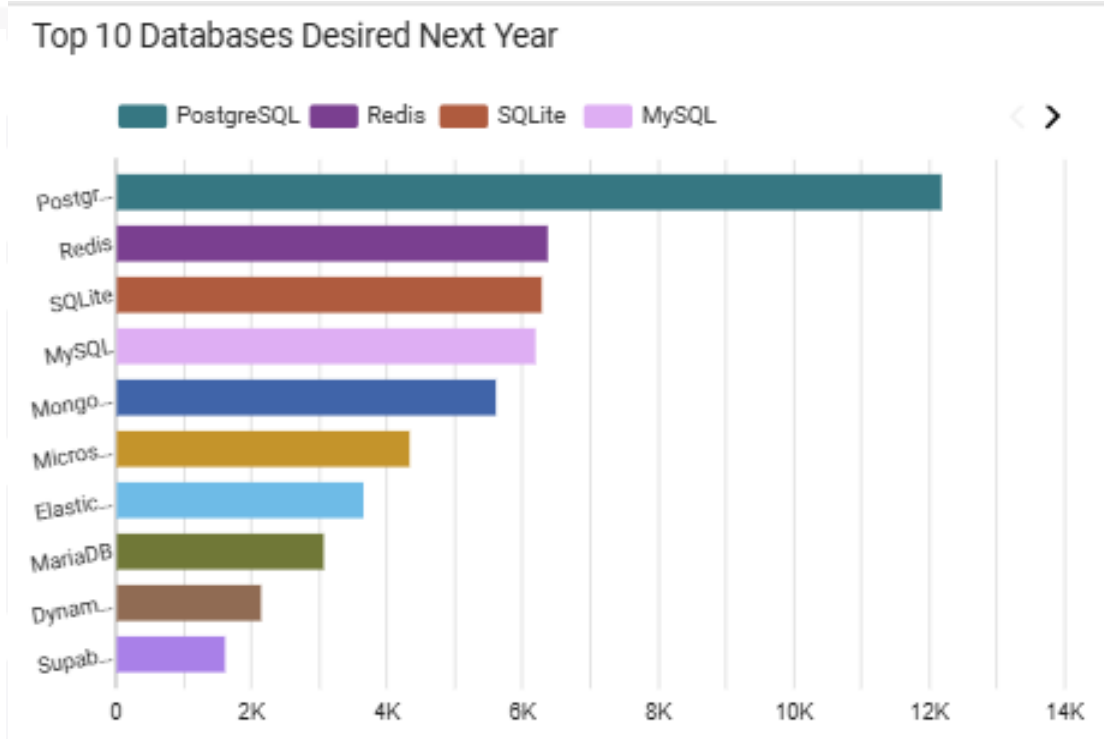


PROGRAMMING LANGUAGE TRENDS - FINDINGS & IMPLICATIONS

- **JavaScript and Python** remain the most widely used languages, reflecting their versatility across web and data workflows.
- **SQL** continues to be a core skill due to its central role in data access and backend systems.
- **TypeScript** usage is rising, indicating growing adoption of typed JavaScript ecosystems.
- **C#, Java, and C++** remain strong among enterprise and system-level developers.
- Overall, the distribution reflects a strong emphasis on general-purpose languages and established multi-paradigm technologies.
- **TypeScript** shows the largest anticipated growth, reinforcing its rising importance in modern web development.
- **Go and Rust** appear prominently among desired languages, reflecting interest in high-performance, cloud-native, and systems-level development.
- **Python** remains in strong demand, driven by continued growth in AI, data science, and automation fields.
- Many developers intend to expand into **strongly typed or low-latency languages**, indicating a shift toward scalable and secure software architectures.
- The future trend emphasizes performance, safety, and cloud-readiness over purely general-purpose usage.

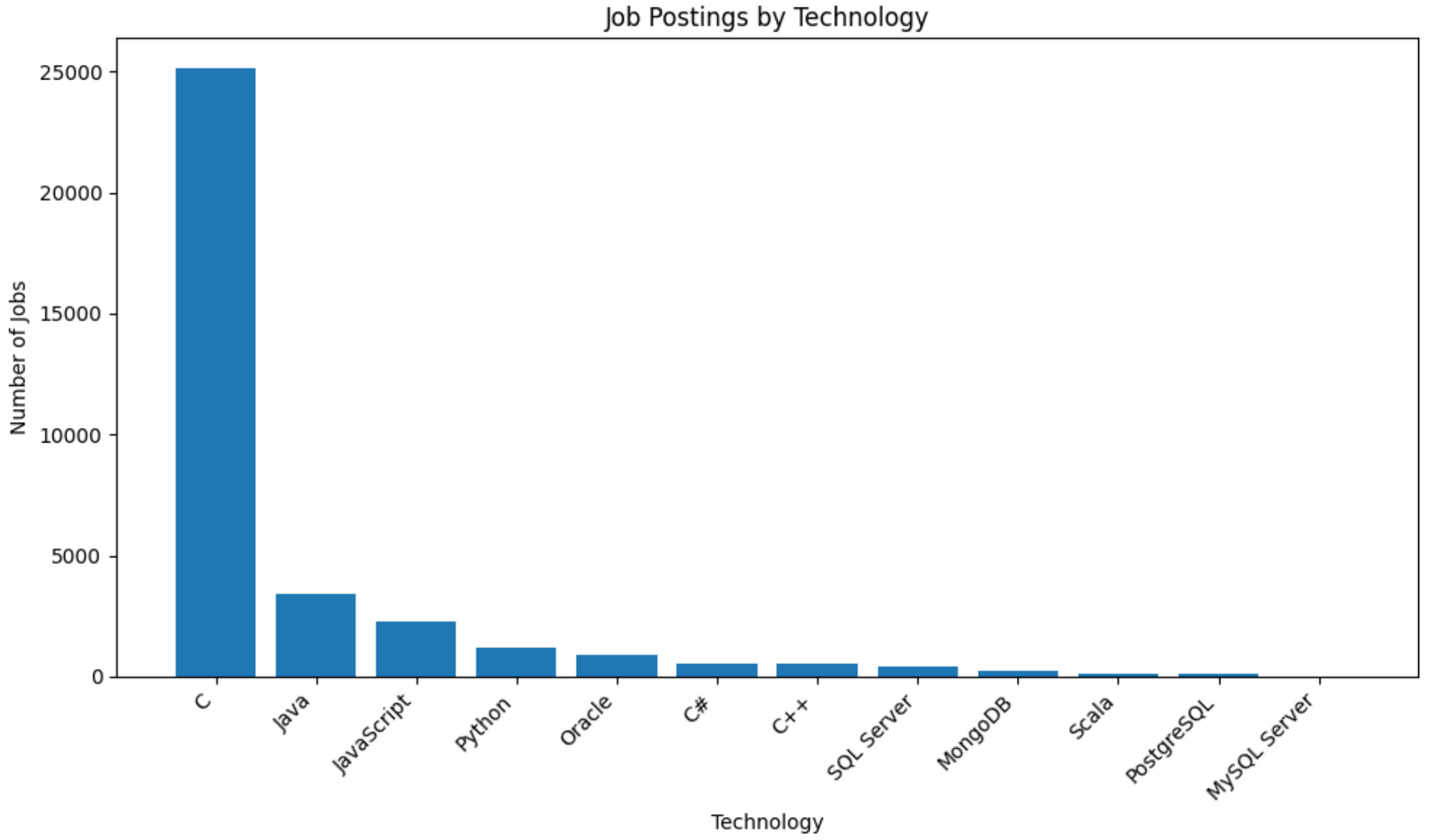
DATABASE TRENDS





DATABASE TRENDS - FINDINGS & IMPLICATIONS

- **PostgreSQL** is the most widely used database, reflecting its strength in scalability, reliability, and open-source adoption.
- **MySQL and SQLite** remain highly popular due to their ease of use, legacy support, and prevalence in web applications.
- **MongoDB** shows strong usage among developers working with document-based or flexible schema designs.
- **Microsoft SQL Server** maintains steady use in enterprise and corporate environments.
- Overall usage suggests a balanced ecosystem where relational databases dominate, with NoSQL options filling specialized needs.
- **Redis** is the most desired database moving forward, likely due to its speed, in-memory performance, and popularity in cloud architectures.
- **PostgreSQL** continues to rank high in future demand, indicating sustained confidence in its long-term viability.
- Developers show growing interest in **MongoDB and SQLite**, especially for lightweight, flexible data workflows.
- The rise in interest for performance-optimized and cloud-native databases suggests a shift toward scalable, distributed system design.
- Demand patterns indicate that developers are prioritizing speed, availability, and production-ready open-source tools in their future learning plans.

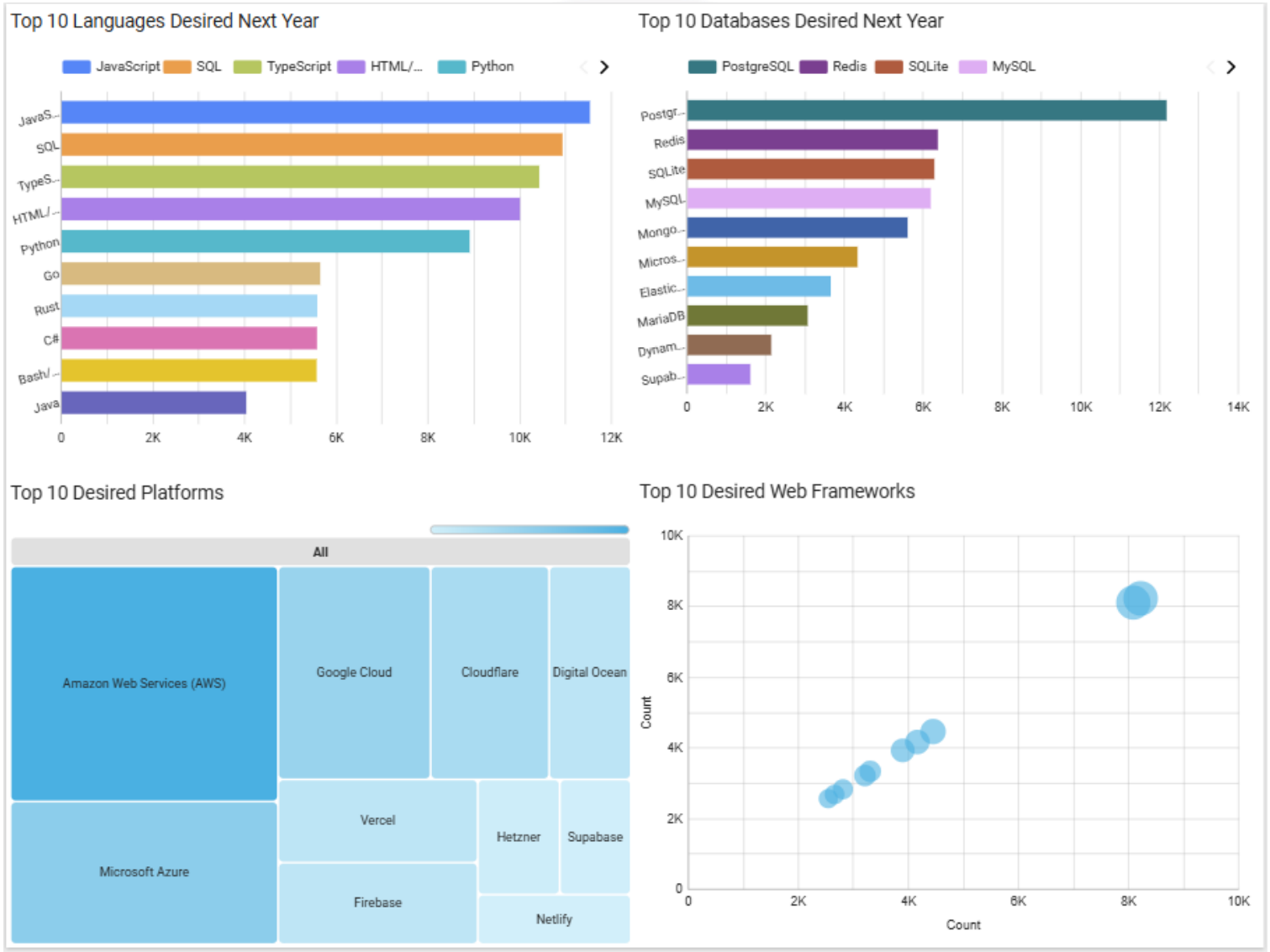

Job Postings by Technology
Number of Jobs
25000
20000
15000
10000
5000
0
C
Java
JavaScript
Python
Oracle
C#
C++
SQL Server
MongoDB
Scala
PostgreSQL
MySQL Server
Technology

DASHBOARD TAB 1

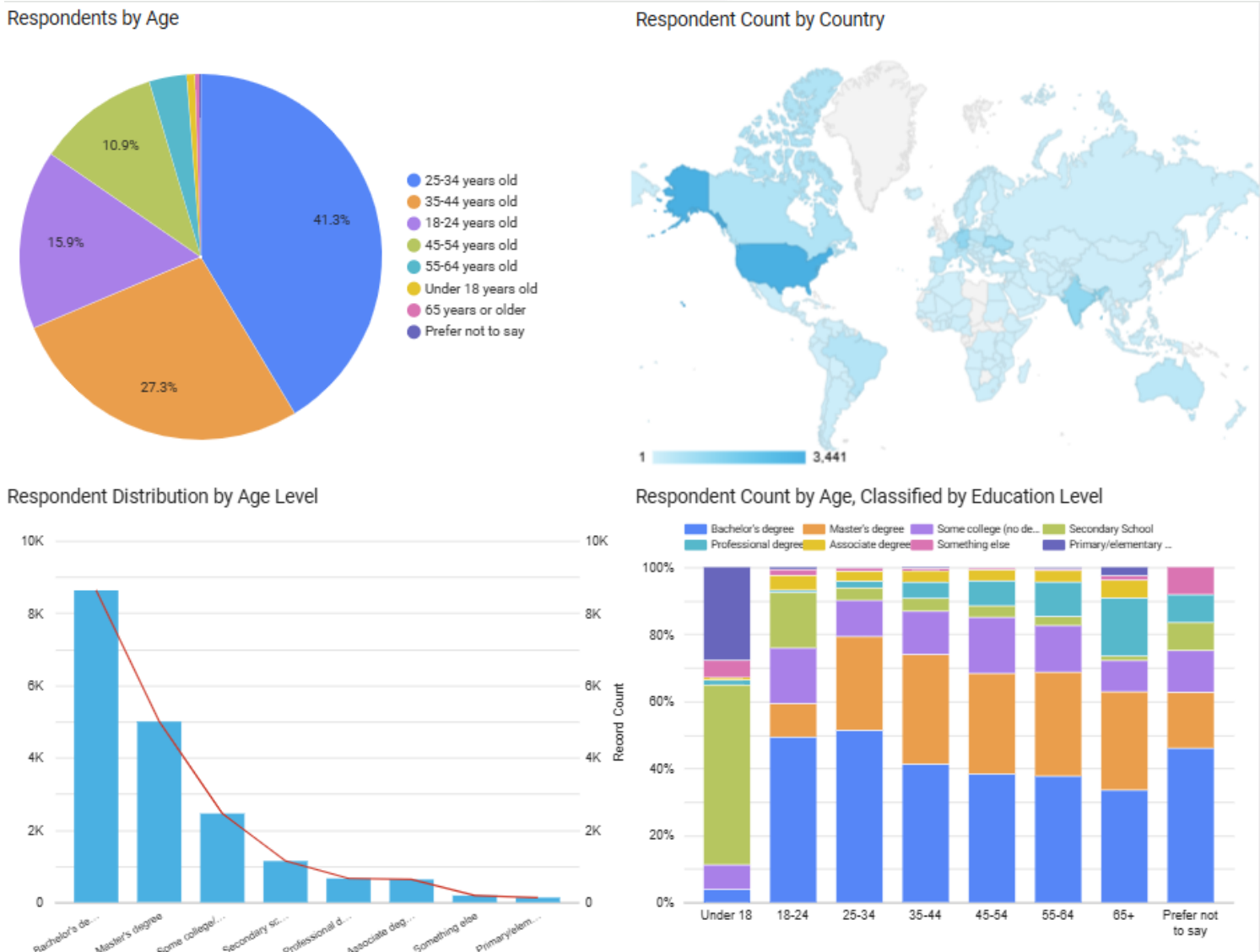
Skills Network

IBM

DASHBOARD TAB 2



DASHBOARD TAB 3



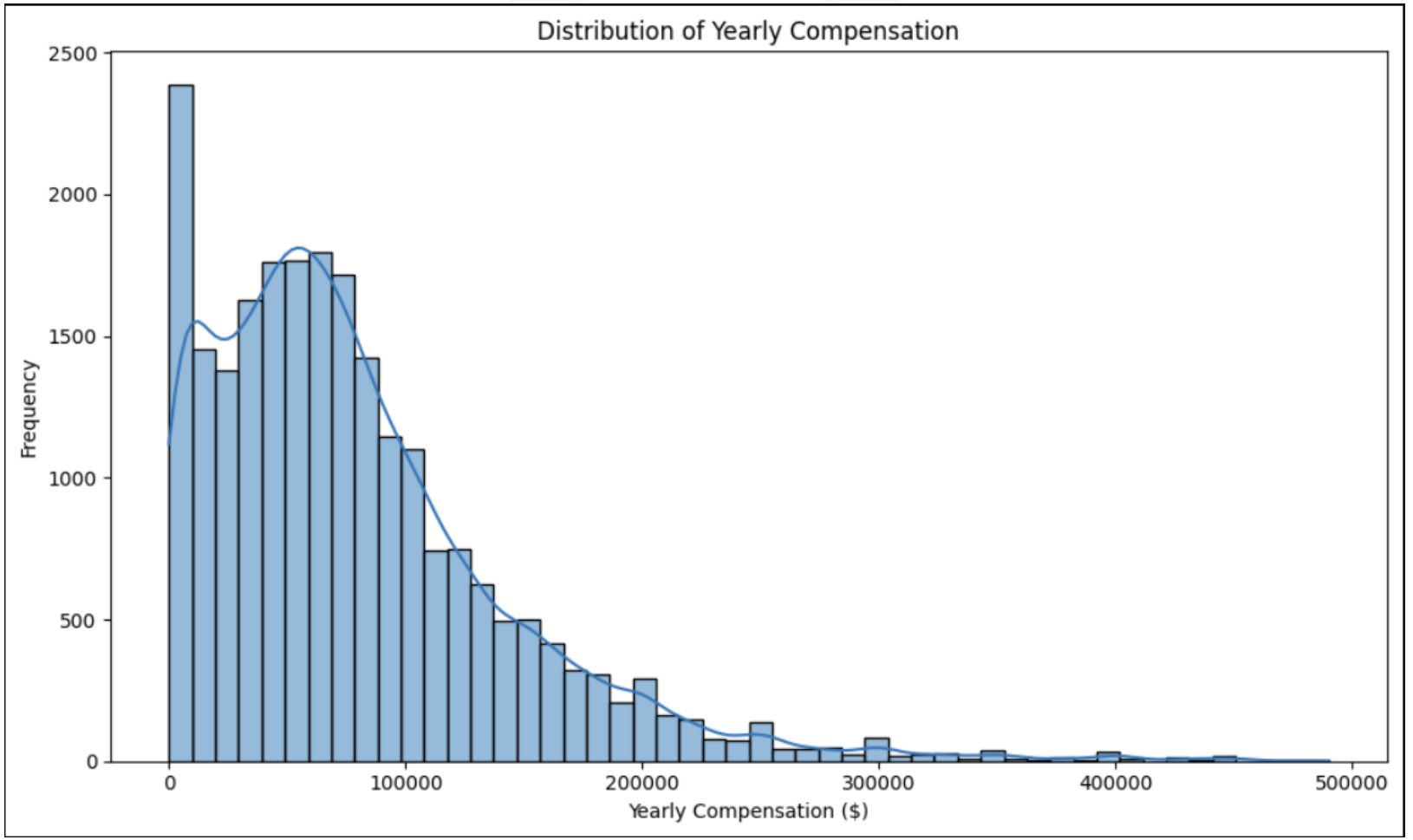
OVERALL FINDINGS & IMPLICATIONS

- **General-purpose languages dominate current usage**, with JavaScript, Python, and SQL forming the core developer toolkit.
- Organizations should expect these skills to remain foundational across most roles.
- **Future learning trends show a shift toward performance and scalability**, with rising interest in TypeScript, Go, and Rust.
- This reflects increasing emphasis on cloud-native systems, type safety, and high-efficiency architectures.
- **Relational databases continue to lead**, especially PostgreSQL and MySQL, due to reliability and broad industry adoption.
- NoSQL systems like Redis and MongoDB show growing momentum, particularly in real-time and distributed applications.
- **Cloud platforms and automation tools are becoming critical skills**, with AWS, Google Cloud, and modern frameworks ranking highly in desired technologies.
- Indicates continued movement toward DevOps, microservices, and AI-driven workflows.
- **Demographics skew toward early-career developers**, primarily ages 25–34 with Bachelor's or Master's degrees.
- Training and hiring strategies should target this demographic while addressing regional skill variations.
- **The ecosystem is rapidly evolving**, with developers actively preparing for tools that support scalability, reliability, and modern software engineering practices.
- Businesses must align technology roadmaps with these shifts to remain competitive.

CONCLUSION

- **The analysis highlights strong continuity in core technologies**, with JavaScript, Python, SQL, and PostgreSQL remaining central to modern development.
- **Future trends point toward performance-oriented and cloud-native tools**, including TypeScript, Go, Rust, Redis, and AWS, reflecting industry movement toward scalable, secure architectures.
- **Developers are actively preparing for emerging technologies**, indicating a healthy, forward-looking ecosystem with rapid skill evolution.
- **Demographic insights show a globally distributed but youthful workforce**, dominated by early-career professionals with formal education pathways.
- **Overall, the findings support data-driven planning** for hiring, curriculum design, and technology adoption strategies across organizations and developer communities.

APPENDIX


Distribution of Yearly Compensation
Frequency
2500
2000
1500
1000
500
0
0
100000
200000
300000
400000
500000
Yearly Compensation ($)