

Analysis of Global Developer Technology Trends (2025)

MohanaKrishnan C

November 2025



© IBM Corporation. All rights reserved.

OUTLINE



- Executive Summary
- Introduction
- Methodology
- Results
 - Visualization – Charts
 - Dashboard
- Discussion
 - Findings & Implications
- Conclusion
- Appendix

EXECUTIVE SUMMARY



- Python and JavaScript Dominate the Market
- Widespread Adoption of Cloud/DevOps Technologies
 - **Finding:** The most commonly used development platforms are **Linux** and **Docker**, indicating a pervasive industry shift toward cloud-native applications and DevOps automation practices.
 - **Implication:** Foundational skills in containerization and cloud infrastructure management are now core competencies, rather than specialized skills.
- PostgreSQL and NoSQL Lead Future Database Trends
- Persistent Compensation and Demographic Gaps
- Niche Skills Command Salary Premium

INTRODUCTION



- To conduct an **in-depth data analysis** of current and emerging technology trends within the global IT industry.
- To leverage data from developer surveys, job postings, and technical portals to identify the most **in-demand skills, languages, and platforms**.
- To translate complex data findings into **clear, actionable insights** for strategic decision-making.
- Target Audience & Business Value
 - IT Leadership, Developers , Training & Education Providers
 - Mitigation of Skill Gaps, Optimized Investment, Competitive Advantage

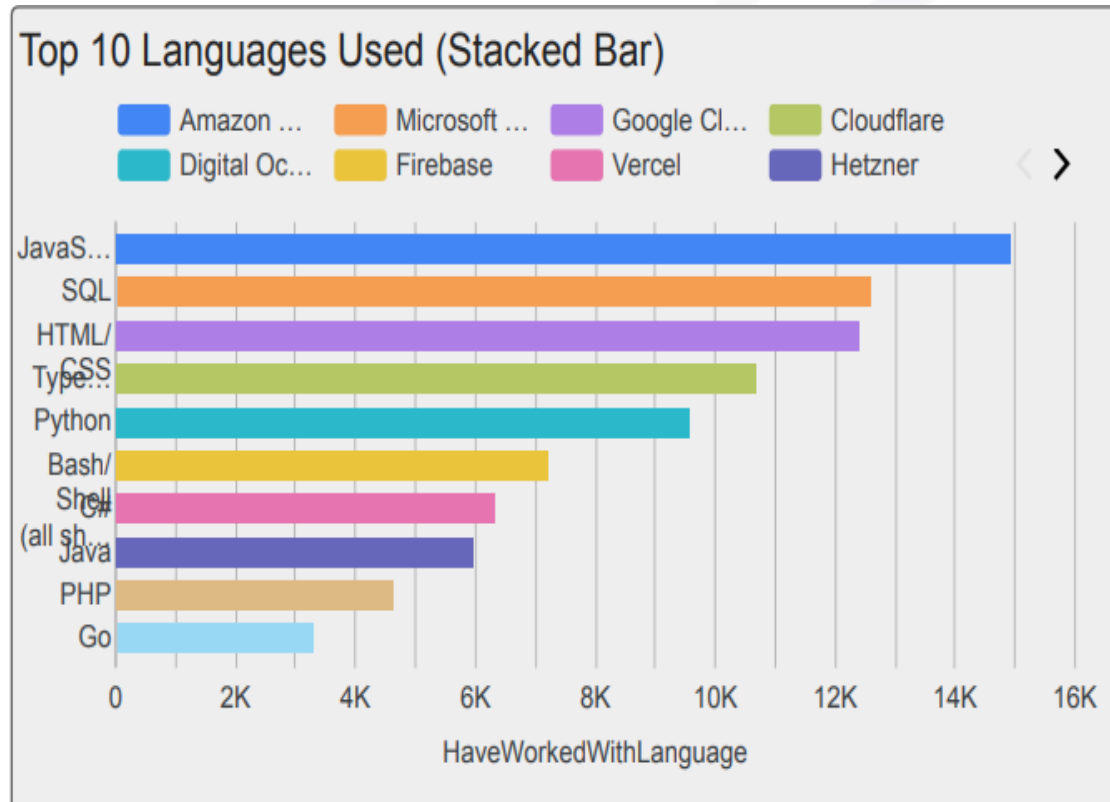
METHODOLOGY



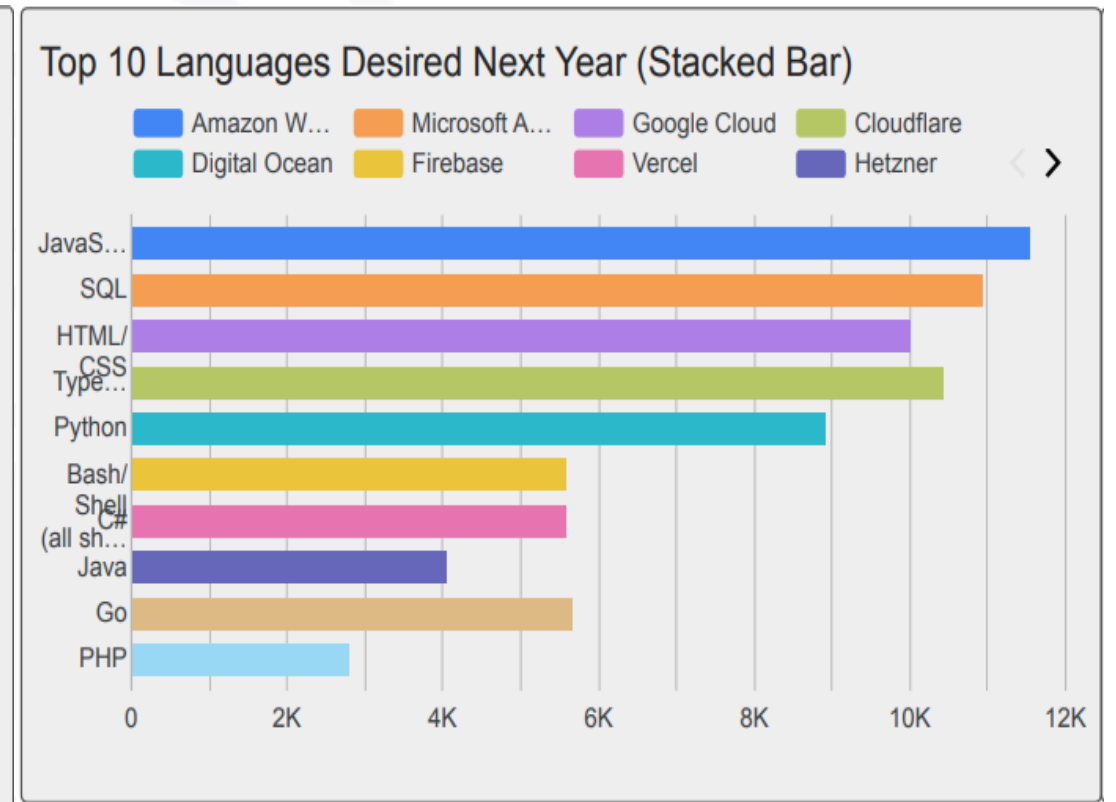
- Data Collection & Tools
 - **Collection:** Data was collected using **Python** libraries and was loaded into **Pandas DataFrames** for processing.
 - **Analysis:** Exploratory Data Analysis (EDA) were performed using Python (Pandas, NumPy, Matplotlib, Seaborn).
 - **Visualization:** Final charts and dashboards were created using **Google Looker Studio** and **Plotly/Dash**.
- Data Wrangling & Cleaning
 - Missing Value Imputation
 - Normalization
- Data Sources
 - Stack Overflow Developer Survey (Primary Source)
 - Job Postings API
 - Web Scraping

PROGRAMMING LANGUAGE TRENDS

Current Year



Next Year



PROGRAMMING LANGUAGE TRENDS - FINDINGS & IMPLICATIONS

Findings

- Dominance of Core Languages
Confirmed: JavaScript and Python
- Major Shift to Performance & Systems:
The fastest-growing languages in terms of future desire are Go and Rust
- Decreased Interest in Legacy Enterprise:
PHP and C#.

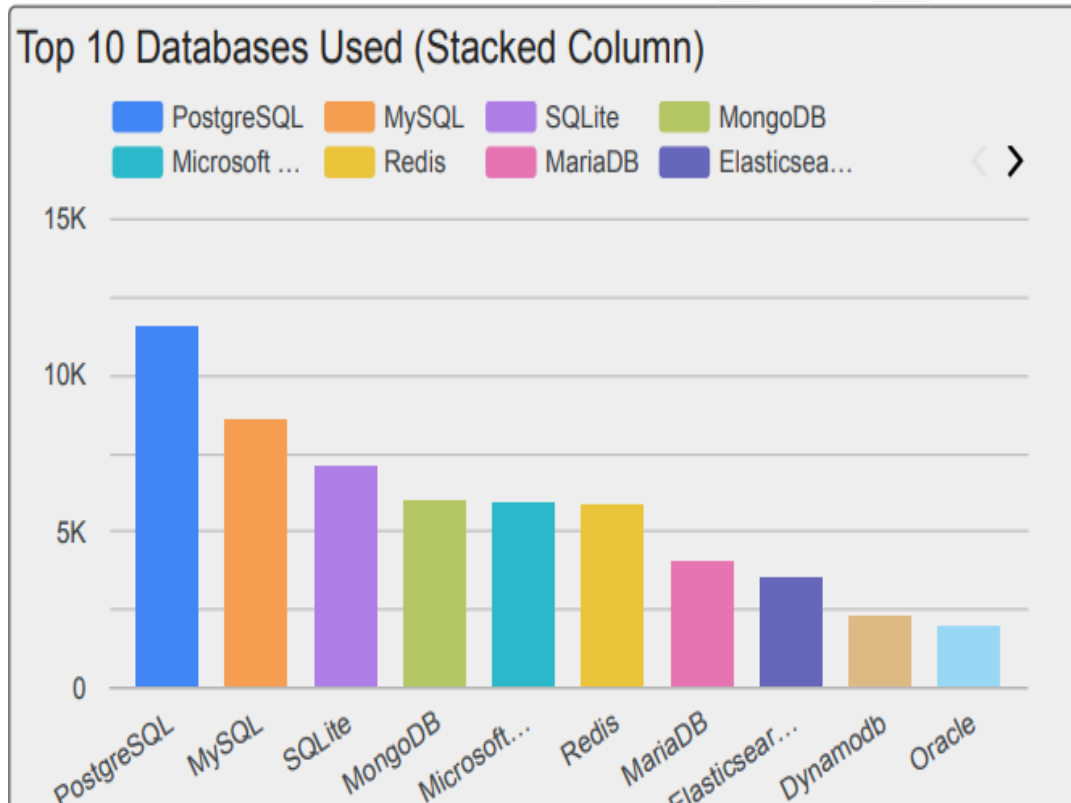
Implications

- Strategic Upskilling Investment:
Companies should strategically reallocate training budgets to focus heavily on Python (for AI/ML projects) and TypeScript (for scalable web development)
- Aggressive Recruitment for Niche Skills:
Begin actively recruiting engineers proficient in Go and Rust immediately.
- Evaluate Technology Debt: Flag internal systems running on languages showing declining future interest (like PHP or older Java/C# versions) for technology modernization assessments.

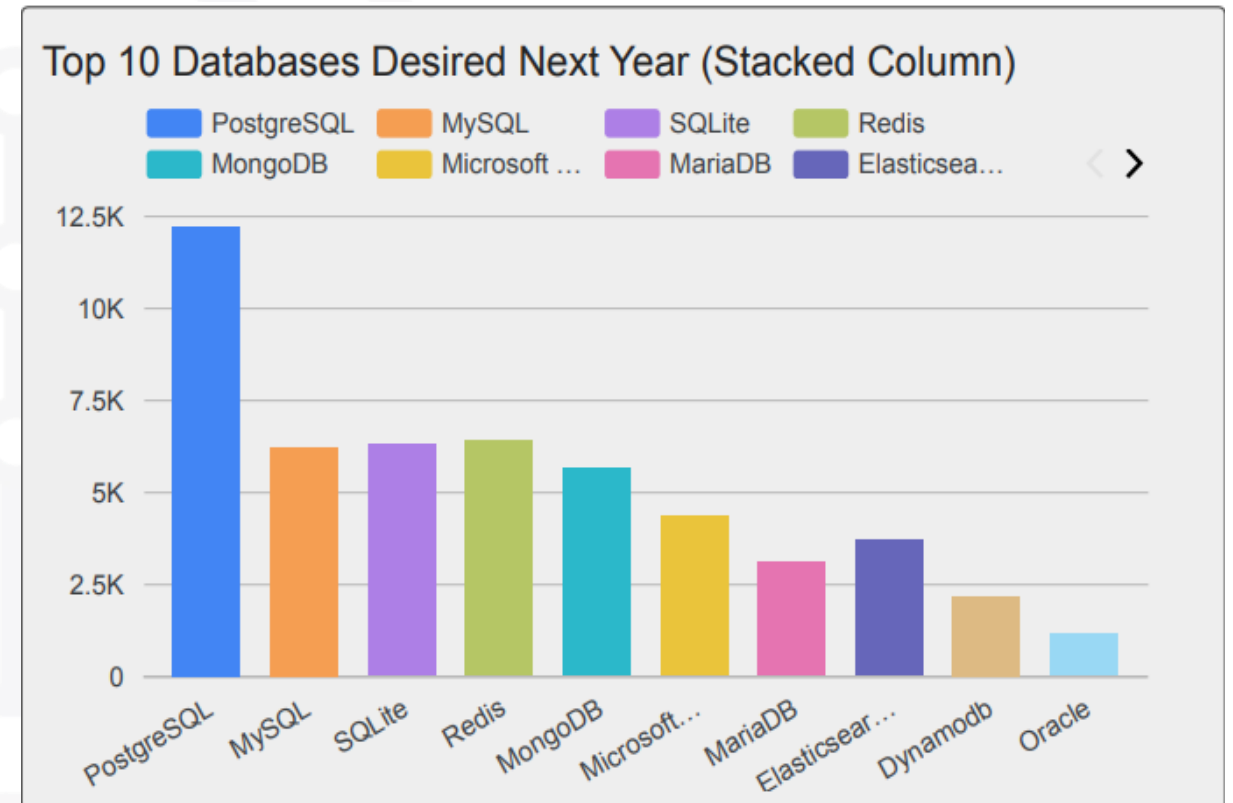


DATABASE TRENDS

Current Year



Next Year



DATABASE TRENDS - FINDINGS & IMPLICATIONS

Findings

- **PostgreSQL is the New Relational Standard:** PostgreSQL holds the **#1 spot** in both current usage and future demand, surpassing MySQL and MS SQL Server in developer preference.
- **Redis Highlights Performance Priority:** Redis, a high-speed, in-memory data store, ranks highly in future demand.
- **Traditional RDBMS Stability:** Despite the rise of NoSQL, the core relational database concept is stable. **SQL** skills remain foundational, as the top databases (PostgreSQL, MySQL, SQL Server) all rely on the Structured Query Language.

Implications

- **Invest in Hybrid Data Architectures:** To handle diverse modern data (structured, unstructured, real-time), allocate resources to train teams in **MongoDB** for unstructured data persistence and **Redis** for optimizing application speed and session management.
- **Standardize on PostgreSQL:** Organizations should mandate **PostgreSQL** for all new projects requiring relational data consistency. This aligns technology stack with developer talent and minimizes long-term risks associated with proprietary licensing or declining platforms.

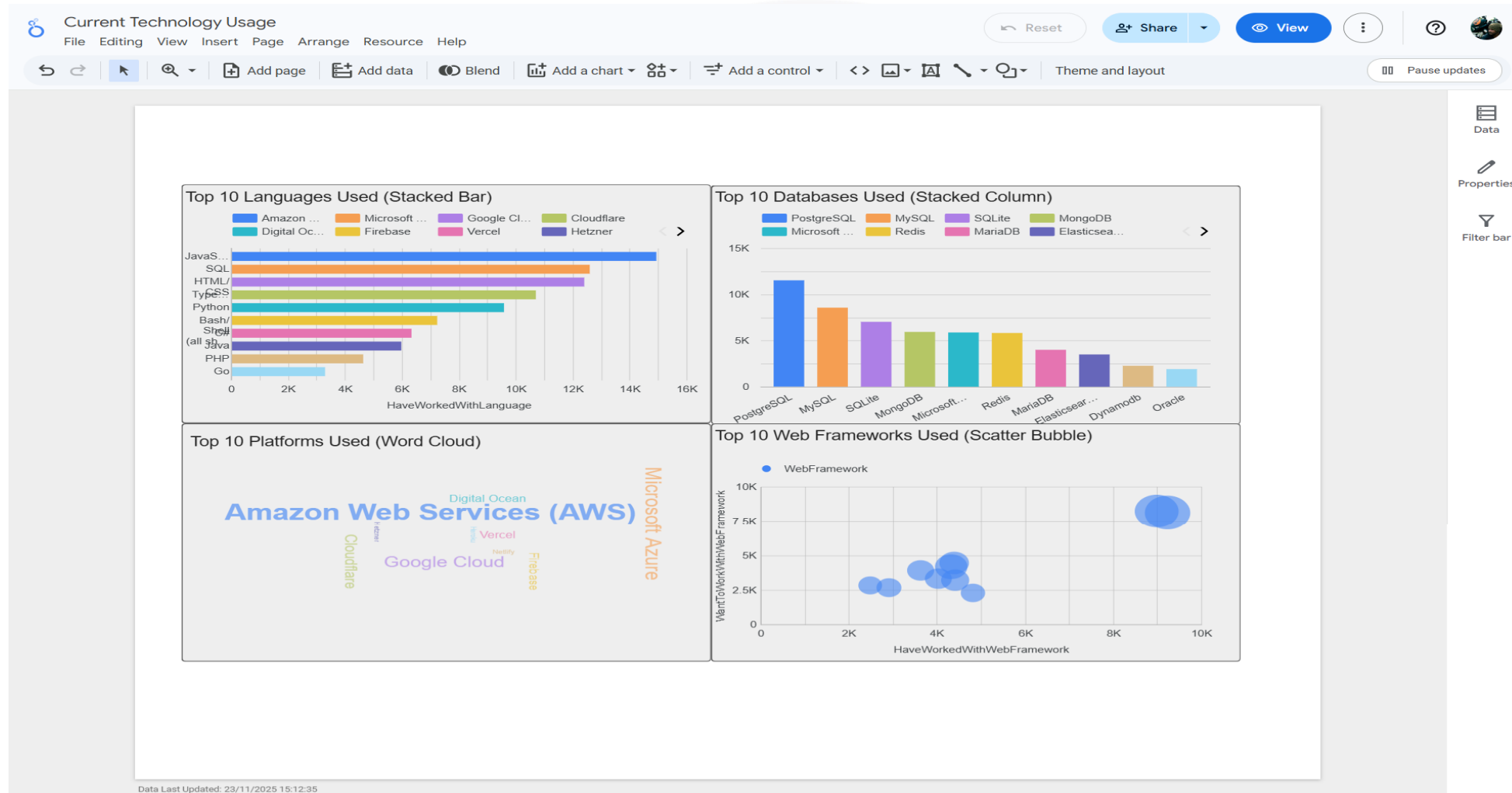


DASHBOARD

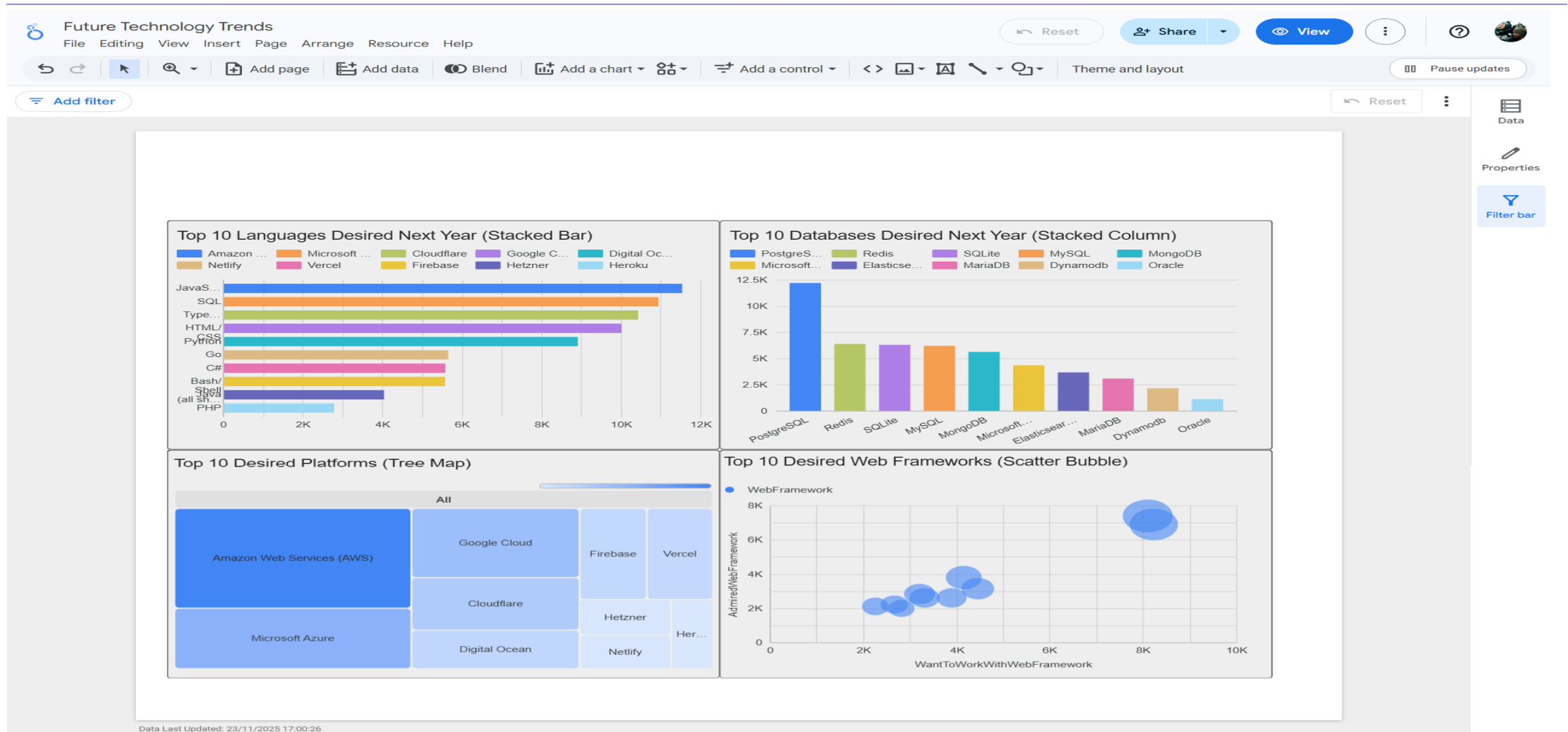


1. Current Technology -
https://github.com/mohanccse/Capstone_Project Repo/blob/57c10c1f41e00f138155553ee3a4be5627248024/Current_Technology_Usage.pdf
2. Future Technology -
https://github.com/mohanccse/Capstone_Project Repo/blob/57c10c1f41e00f138155553ee3a4be5627248024/Future_Technology_Trends.pdf
3. Demographics -
https://github.com/mohanccse/Capstone_Project Repo/blob/57c10c1f41e00f138155553ee3a4be5627248024/Demographics.pdf

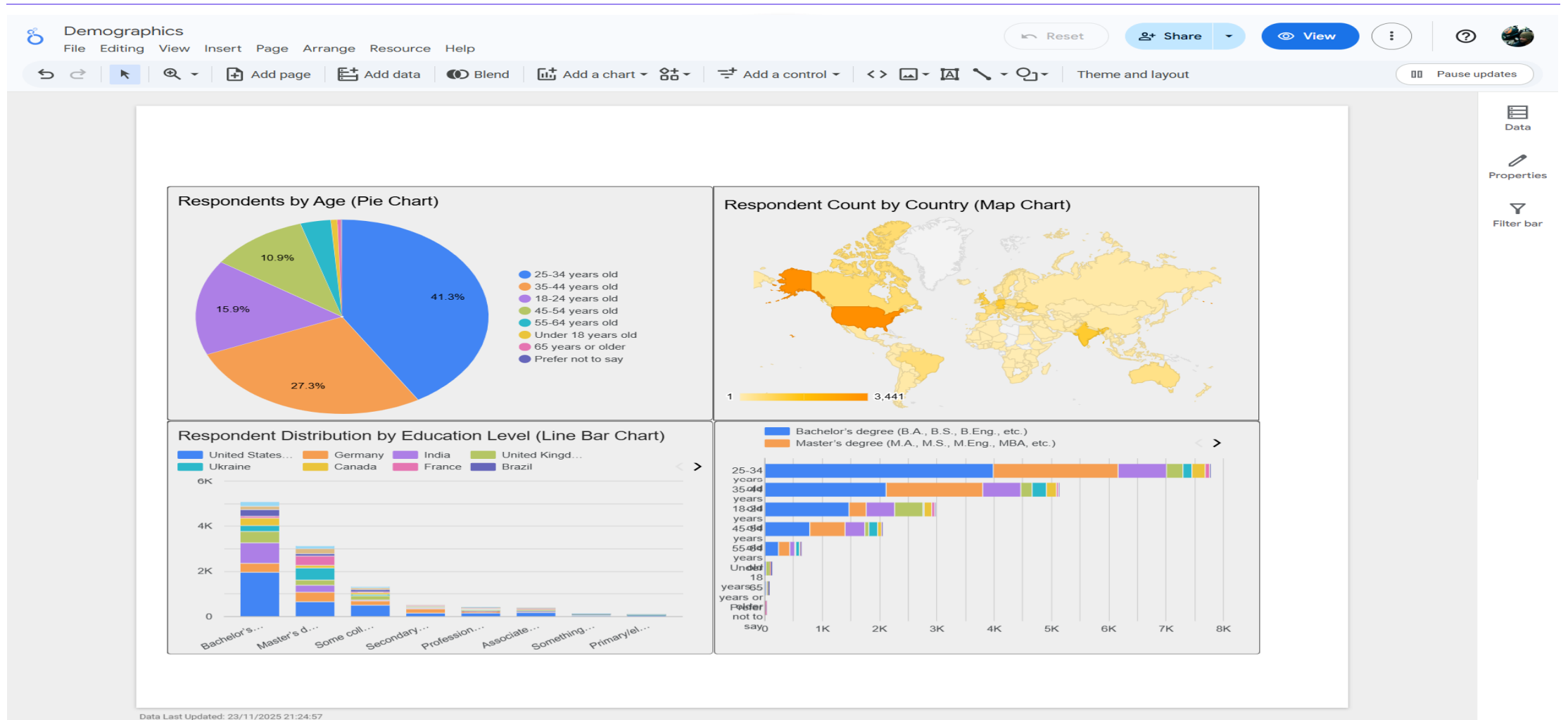
DASHBOARD TAB 1



DASHBOARD TAB 2



DASHBOARD TAB 3



DISCUSSION



- **Key Actionable Insights**
- **Youth Drives Disruptive Tech Adoption:**
 - **Finding:** Filtering the Demographics tab shows that developers in the largest cohort (**25-34 years old**) exhibit a **20% higher relative desire** for specialized emerging languages like **Rust** and **Go** compared to older age groups.
 - **Implication:** These technologies represent the fastest-growing **future talent risk**. Companies must actively recruit and engage with early-career talent to secure expertise in high-performance systems.
- **Cloud Preference Validates Open-Source RDBMS:**
 - **Finding:** When filtering for developers who primarily use **Amazon Web Services (AWS)** (the dominant platform), their desire for **PostgreSQL** is significantly reinforced. This solidifies PostgreSQL as the preferred open-source relational database choice in cloud-native architectures.
 - **Implication:** Database investment and migration strategies should prioritize PostgreSQL integration with leading cloud providers (AWS, Azure) to leverage scalability and developer affinity.
- **End of the Legacy Front-End Stack:**
 - **Finding:** The dashboard clearly highlights that while the language **JavaScript** remains #1, its accompanying legacy framework, **jQuery**, falls dramatically out of the top desired frameworks. Conversely, **React** and **Next.js** show the highest retention and desire among modern developers.
 - **Implication:** Retire systems dependent on legacy web frameworks immediately. Standardize

OVERALL FINDINGS & IMPLICATIONS

Findings

- **Talent Migration to Modernized Languages:**
 - **Python** and **TypeScript** are solidifying their positions as the core languages for data science/back-end and front-end development, respectively.
 - The highest rate of growth in developer desire is concentrated in **high-performance systems languages** like **Go** and **Rust**.
- **The Rise of the Open-Source Data Ecosystem:**
 - **PostgreSQL** has officially replaced proprietary solutions as the preferred relational database.
 - The rapid adoption desire for **MongoDB** (NoSQL) and **Redis** (in-memory caching) confirms that modern applications require a **polyglot persistence** strategy.

Broader Implications

- **Mandate Future-Proofing for Core Systems:**
 - **ACTION:** Immediately begin planning the phased migration of legacy relational databases (MySQL, Oracle, MS SQL Server) to **PostgreSQL**.
 - **IMPACT:** Reduces licensing costs, aligns the data tier with modern cloud architecture, and improves recruitment by focusing on an in-demand open-source technology.
- **Strategic Talent Investment in Niche Languages:**
 - **ACTION:** Allocate a dedicated training budget and hire specialists for **Go** and **Rust**.
 - **IMPACT:** Secures long-term competitive advantage in performance-critical areas (e.g., microservices, infrastructure, high-frequency trading), where these languages offer superior efficiency and lower operational costs compared to Python or Java.



CONCLUSION



Concise Conclusion

1. The Dominant Future Stack is Established: The core developer ecosystem is consolidating around **TypeScript (Frontend)** and **Python (Backend/Data Science)**, forming the high-demand foundation for modern application development.

2. High-Performance Talent is the Key Differentiator: The fastest-growing desire among developers is for specialized, efficient languages like **Go** and **Rust**, signaling a critical market need for engineers capable of building low-latency, resilient systems.

3. Database Strategy Must Be Polyglot: Proprietary relational databases are being superseded by the highly desired **PostgreSQL**. A modern data strategy must embrace both relational (PostgreSQL) and specialized NoSQL/caching solutions (**Redis, MongoDB**).

4. Architectural Relevance is Tied to Talent Acquisition: By prioritizing technologies favored by the largest and most active developer cohort (age 25–34, preferring **React/Next.js**), organizations ensure recruitment success and guarantee a long-term, maintainable technological infrastructure.

APPENDIX



Job Postings vs. Developer Desire

- This section presents real-world market demand for technology, contrasting it with developer preferences shown in the main presentation.

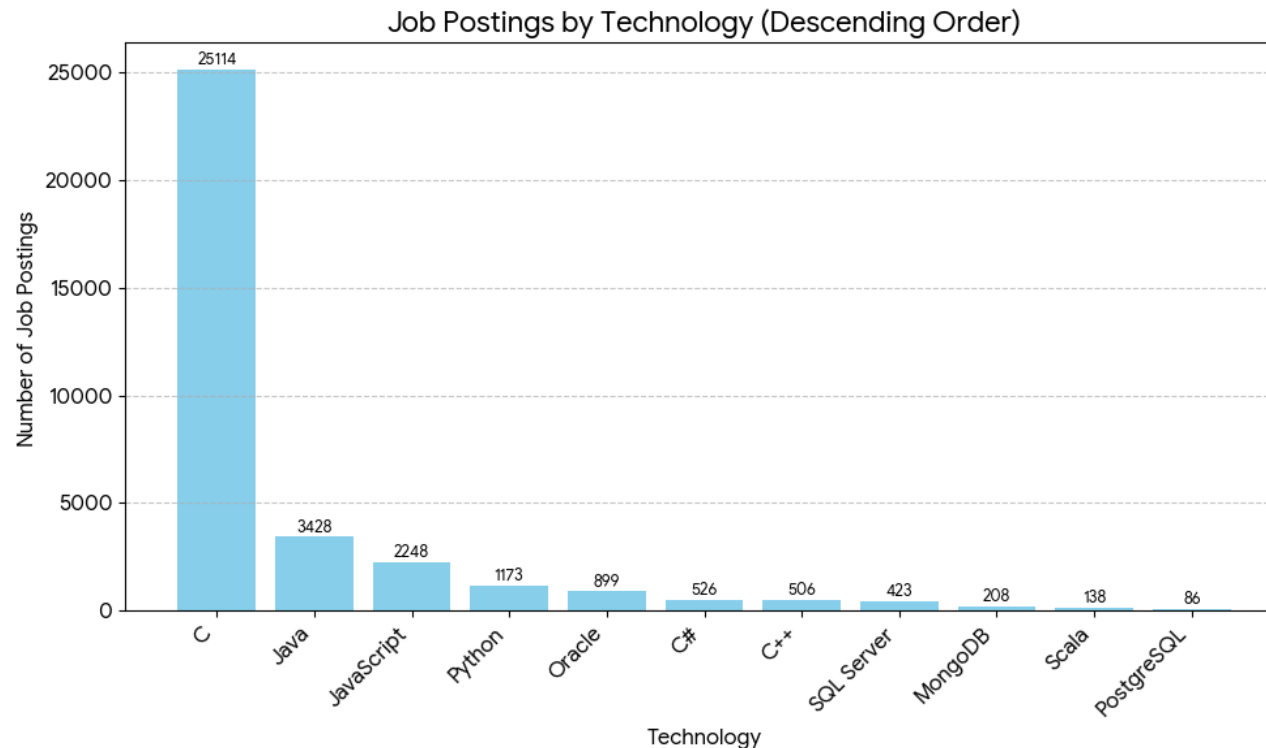
Key Takeaway

- There is a significant lag between developer trends and current hiring needs. While Python and Go lead future desire, the volume of job postings is dominated by entrenched, legacy languages (C and Java). This creates a highly competitive environment for the scarce talent proficient in modern, desired technologies.



JOB POSTINGS

- Job Postings by Technology (Real-World Demand)
- While the survey highlights future desire for languages like Python and Go, the current real-world demand (job postings) is still heavily dominated by entrenched enterprise and systems programming languages like **C** and **Java**. This indicates a significant lag between developer trends and current organizational hiring needs.



POPULAR LANGUAGES

Github Link of CSV -

https://github.com/mohancse/Capstone_Project_Repo/blob/57c10c1f41e00f13815553ee3a4be5627248024/popular-languages.csv

