Data report on Coding survey

Part 1

OUTLINE

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

EXECUTIVE SUMMARY

•**Talent Landscape**
The developer community is young (25–34), highly educated, and concentrated in regions like the US, Germany, and India. Full-stack and back-end roles dominate the market.
•**Technology Trends**
Python, JavaScript, SQL, and TypeScript are the most widely used and admired programming languages.
PostgreSQL leads as the most desired database, highlighting a shift toward open-source and cloud-ready tools.
React and Node.js remain the most sought-after web frameworks.
•**AI Integration is Widespread**
Over 29,000 developers use AI tools for coding, debugging, and documentation — signaling a transformation in how software is built and maintained.
•**Compensation & Industry Outlook**
Salaries concentrate around $100K, with Fintech, Internet Services, and Banking generating the highest compensation volumes.
•**Strategic Takeaway**
To stay competitive, organizations must:

- Invest in in-demand technologies (e.g., Python, PostgreSQL, React)
- Adopt AI-assisted workflows
- Modernize legacy systems
- Engage global talent with flexible, future-facing tech stacks

INTRODUCTION - Project Overview

- **Goal:** Extract, clean, analyze, and visualize data to uncover insights from the Stack Overflow Developer Survey and job-related APIs
- **Focus Areas:**
    - Real-world data collection using APIs and web scraping
    - Data cleaning and wrangling for analysis readiness
    - Exploratory Data Analysis (EDA) to identify patterns and trends
    - Visual storytelling through charts, dashboards, and final presentation
- **Tools & Technologies:**
    - Python (pandas, requests, BeautifulSoup, matplotlib, seaborn)
    - IBM Cognos Analytics for dashboarding
    - Microsoft PowerPoint for presenting final findings

INTRODUCTION - Objectives

- **Data Acquisition:**
  - Understand and apply HTTP requests
  - Retrieve structured data via GitHub APIs
  - Perform web scraping to gather HTML-based data
- **Data Processing & Exploration:**
  - Clean and normalize datasets (remove duplicates, handle missing values)
  - Analyze distributions, detect outliers, and explore feature relationships
- **Communication & Visualization:**
  - Create clear, insightful charts and dashboards
  - Present findings effectively using best practices in data storytelling

METHODOLOGY - Data Collection and Preparation

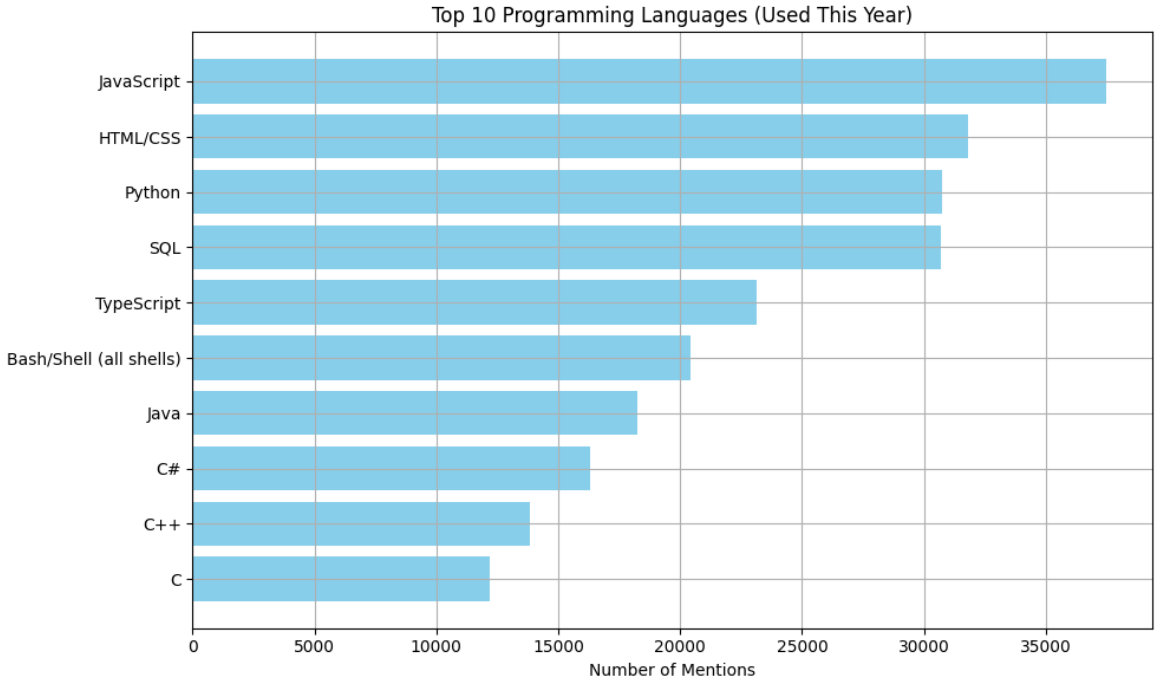
- **Collected Data via APIs & Web Scraping**
- Analyzed HTTP requests to interact with the GitHub REST API
- Retrieved and paginated job postings using the GitHub Jobs API
- Performed web scraping: downloaded pages, extracted links, images, and HTML table data
- **Cleaned & Structured Data (Data Wrangling)**
- Identified and removed duplicate entries
- Handled missing values using imputation strategies
- Normalized key numerical columns for accurate comparisons
- **Used Python (pandas, requests, BeautifulSoup) for all data processing steps**

METHODOLOGY - Analysis, Visualization & Reporting

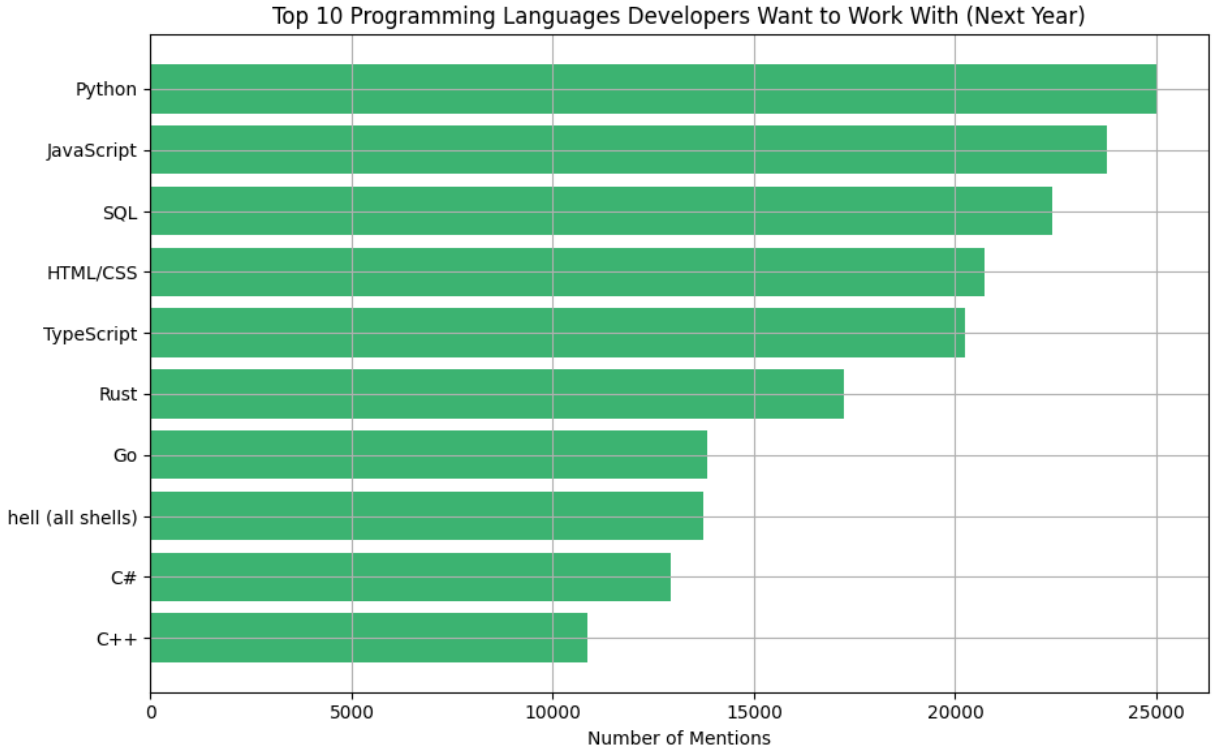
- **Conducted Exploratory Data Analysis (EDA)**
  - Analyzed distributions, correlations, and outliers using plots and statistics
  - Created summary visualizations: histograms, box plots, scatter plots, heatmaps
- **Visualized Trends and Relationships**
  - Compared categories with pie, bar, stacked, and line charts
  - Analyzed technology preferences, job satisfaction, and compensation patterns
- **Built Dashboards and Presented Findings**
  - Created dashboards with IBM Cognos / Looker Studio (current & future tech usage, demographics)
  - Summarized insights in a structured PowerPoint presentation using data storytelling best practices

PROGRAMMING LANGUAGE TRENDS

Current Year



Next Year



PROGRAMMING LANGUAGE TRENDS - FINDINGS & IMPLICATIONS

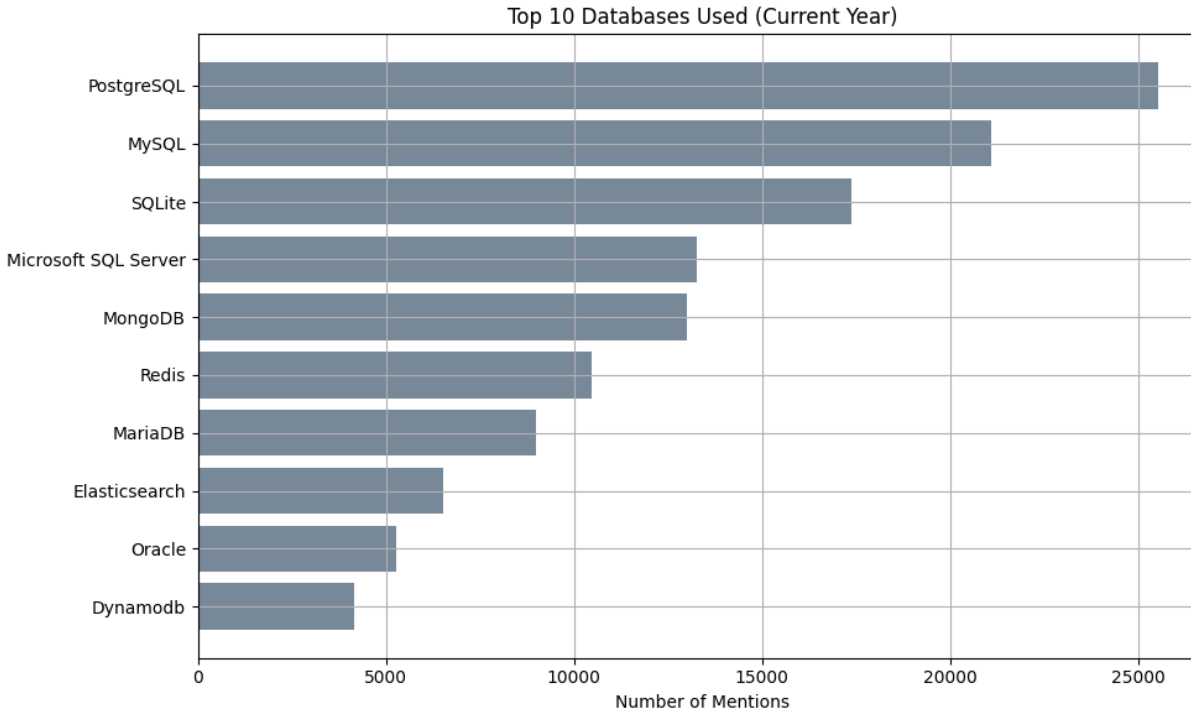
Findings

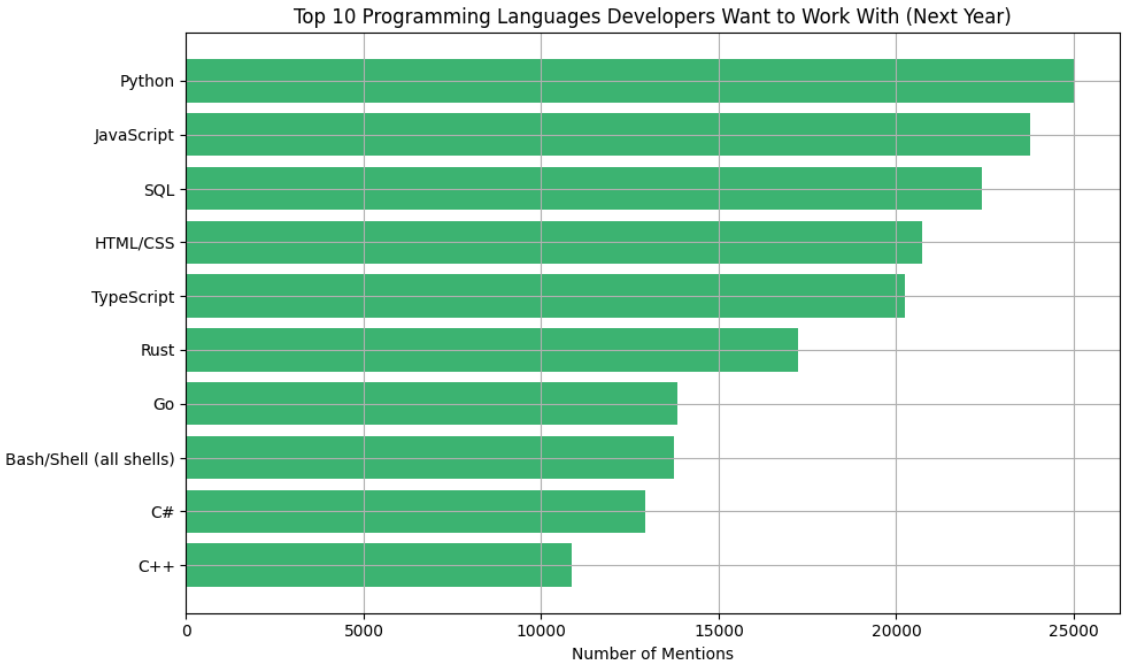
- Python is the most desired language, overtaking JavaScript.While JavaScript is the most used, Python leads in interest for future use, indicating a strong shift in developer preference.
- Rust and Go are rising stars.Despite not appearing in the current top 10 used languages, Rust and Go rank 6th and 7th in the “want to work with” list — showing emerging developer enthusiasm.
- Traditional languages like Java and C are losing appeal.Java and C are used frequently (7th and 10th), but neither appears in the top 10 most wanted list, suggesting waning interest among developers.

Implications

- Organizations should invest more in Python and TypeScript.These languages are both popular now and in future demand — ensuring a healthy talent pipeline and long-term viability.
- Tech stacks should consider adopting Rust or Go for new projects.Since many developers want to work with these languages, offering projects in them could attract top talent and modernize systems.
- Legacy systems in Java or C may face future staffing challenges.As fewer developers express interest in these languages, maintaining systems built on them could become increasingly difficult and costly.

DATABASE TRENDS





DATABASE TRENDS - FINDINGS & IMPLICATIONS

Findings

- PostgreSQL is the dominant and most desired database.It ranks #1 both in current usage and future interest — showing its continued momentum as the preferred open-source RDBMS.
- Modern, developer-friendly databases like Supabase are gaining traction.While not in the current top 10, Supabase appears in the top 10 wanted list — signaling rising popularity in serverless and Jamstack ecosystems.
- Traditional enterprise databases like Oracle and Microsoft SQL Server are declining in appeal.While still widely used (e.g., Oracle #9 in usage), they rank lower or are absent in developer preference for future use.

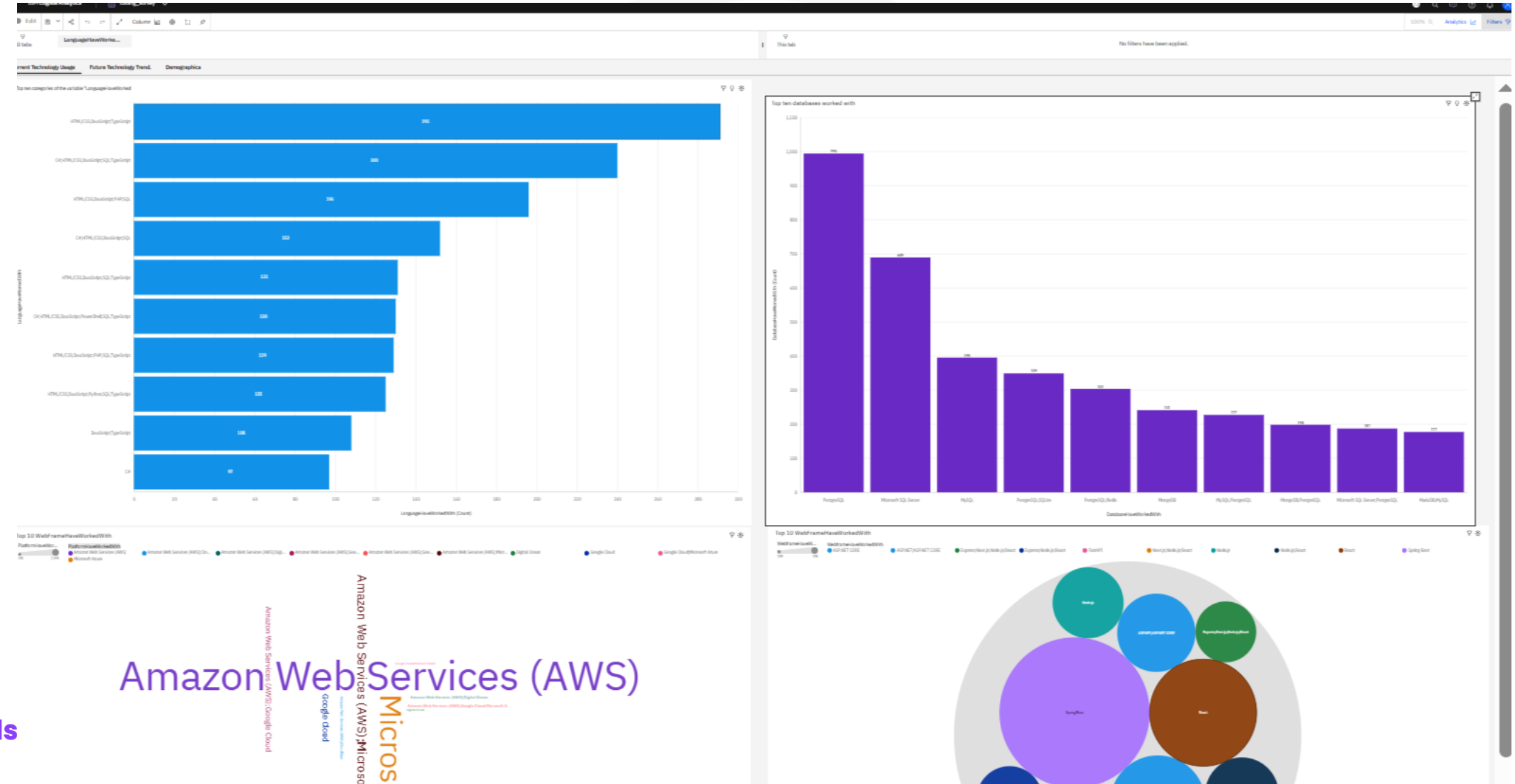
Implications

- PostgreSQL should be a strategic default for new projects.Its strong community support and wide adoption ensure both talent availability and ecosystem stability.
- Tech teams should explore integrating newer tools like Supabase.These tools align with modern developer workflows (e.g., serverless, real-time apps) and may help attract talent looking to work with cutting-edge stacks.
- Organizations relying heavily on legacy systems (like Oracle) should plan for modernization or risk future hiring and maintenance challenges, as developer interest continues to decline.

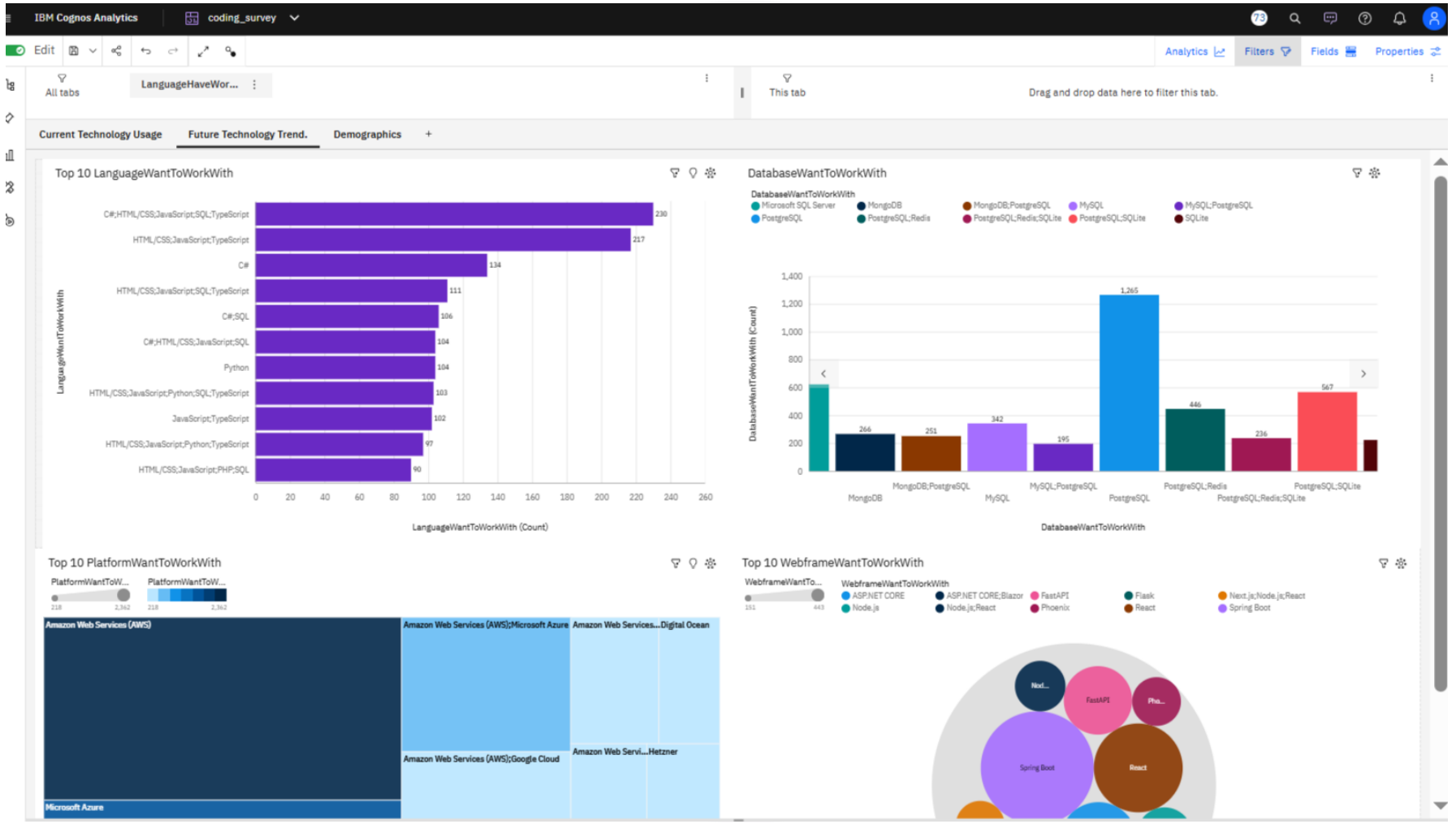
DASHBOARD

< Please present your dashboard in the following slides.>

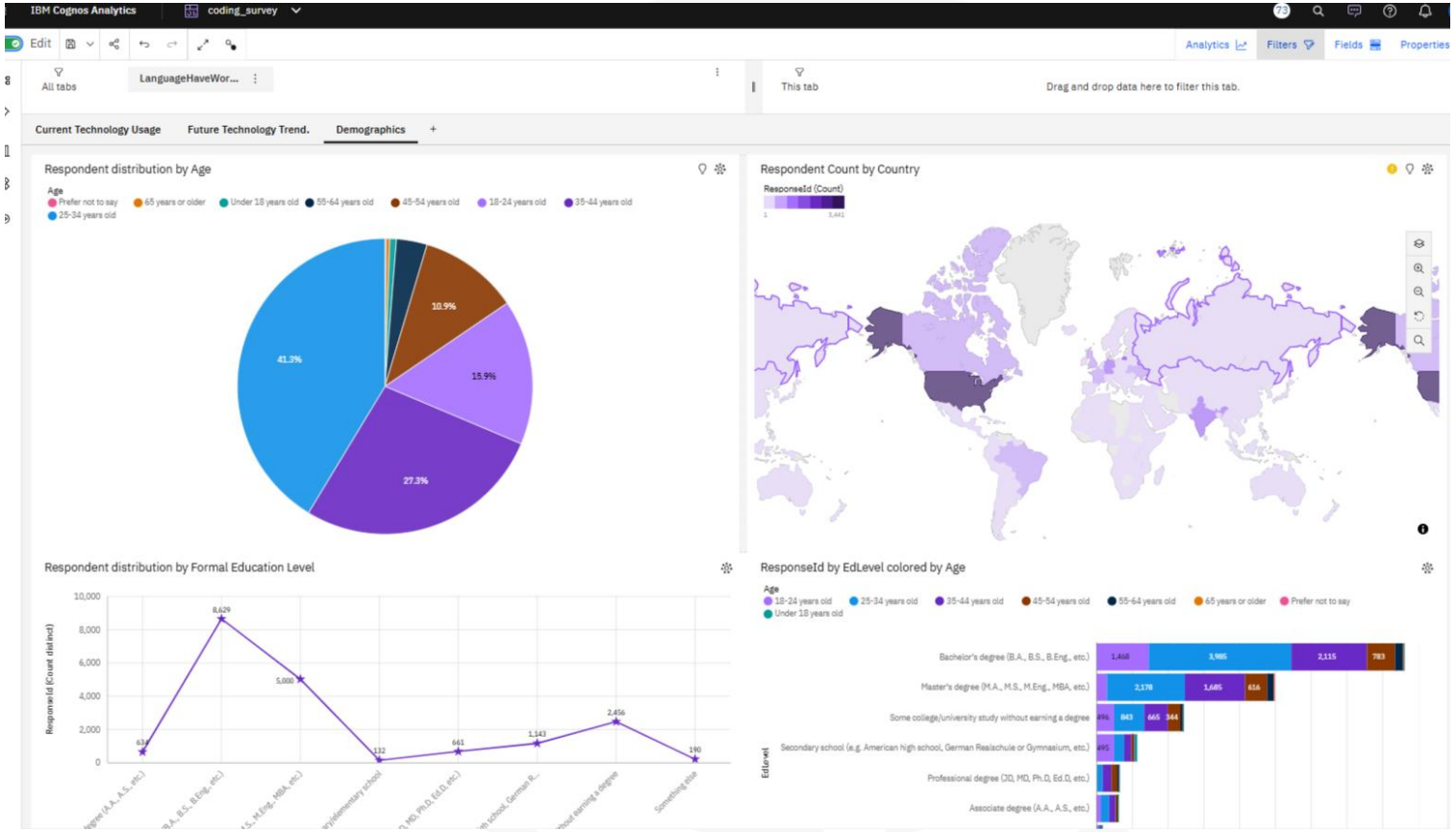
DASHBOARD TAB 1



DASHBOARD TAB 2



DASHBOARD TAB 3



DISCUSSION - Programming Language Trends – Usage vs. Satisfaction

- Most Used Languages
  - JavaScript, HTML/CSS, Python, SQL, and TypeScript top the usage charts
  - These languages also lead in admiration, showing alignment between use and developer preference
- Satisfaction Outliers
  - Groovy, Solidity, Zephyr, and Dart show highest average satisfaction
  - Despite lower adoption, these niche languages may offer high developer enjoyment in specific contexts
  - Low Satisfaction
  - Languages like Julia, Fortran, and R score lowest on satisfaction
  - May indicate limited use cases, outdated tooling, or learning curve issues
- Balanced Stars
  - TypeScript and Python combine high usage, high admiration, and good satisfaction — ideal for modern development stacks

Let me know if you’

DISCUSSION - Strategic Insights Recommendations

- Growth Languages: Python, TypeScript, Rust, Go
  - Python leads in future interest despite being only 3rd in current use
  - Rust and Go show rising appeal — attractive for developers, good fit for new projects
- Declining Developer Interest
  - Java and C: still widely used, but missing from top “wanted” languages
  - Indicates potential long-term resourcing and maintenance challenges
- Strategic Recommendations
  - Invest in Python & TypeScript: combine broad adoption with strong developer interest
  - Adopt Rust or Go for greenfield projects: attract talent and future-proof systems
  - Reassess reliance on legacy tech (Java, C): plan transitions or upskilling
- AI Tools Influence Usage
  - Widespread use of AI tools (e.g., for writing, debugging, documentation)
  - may increase adoption of languages with better tool support like Python and JavaScript

DISCUSSION - Key Insights – Top Databases Among Developers

- PostgreSQL Leads in Popularity
  - Most used and most wanted database
  - Valued for reliability, advanced features, and open-source support
  - Preferred in modern, cloud-native, and analytics applications
- Relational Databases Remain Core Technologies
  - MySQL: Widely used in web dev (e.g., LAMP stack)
  - SQLite: Popular for mobile and embedded systems
  - SQL Server & Oracle: Still entrenched in enterprise environments
- Rise of NoSQL Options
  - MongoDB: Flexible, JSON-based, great for web apps
  - Redis: Specialized in high-speed caching and real-time systems

DISCUSSION - Key Insights – Top Databases Among Developers

- **Growth in Cloud-Native & Scalable Solutions**
  - **Elasticsearch**: Favored for search, logging, observability
  - **DynamoDB**: Gaining traction via AWS ecosystem
- **Oracle Losing Developer Interest**
  - Present in usage, absent in "most wanted"
  - Perceived as outdated, expensive, or overly complex
- **Open-Source Momentum**
  - Clear shift toward **community-driven** and **cloud-friendly** tools
  - PostgreSQL, MongoDB, and Redis benefit from Docker/K8s support
- **Strategic Recommendation**
  - Align with technologies that combine strong developer interest and modern deployment compatibility

OVERALL FINDINGS

- Demographics
  - Majority of respondents aged 25–34, followed by 35–44
  - Top countries: USA, Germany, India, UK, Ukraine
  - Most common roles: Full-stack and Back-end developers
  - Education: Majority hold Bachelor’s or Master’s degrees
- Most Used Tools & Technologies
  - JavaScript, HTML/CSS, and Python top the usage charts
  - PostgreSQL is the most wanted database, surpassing MySQL and MongoDB
  - Web frameworks in demand: React, Node.js, Next.js
- AI Integration is Widespread
  - Developers heavily use AI tools for writing code, searching, and debugging
  - Over 29,000 use AI tools for coding tasks alone
- Compensation Patterns
  - Common salaries cluster around $100,000
  - Highest compensation sums in Fintech, Internet/Telecom, and Banking industries
- Operating System Preferences
  - Windows leads for professional use, followed by macOS and Ubuntu
  - WSL adoption is strong, showing cross-platform development growth

Let me know if you

OVERALL IMPLICATIONS for developers and organizations

- **Invest in In-Demand Skills**
  - Focus on **Python**, **TypeScript**, and **PostgreSQL** for long-term career and project viability
  - Master popular frameworks like **React** and **Node.js** to stay competitive
- **Align with Industry Trends**
  - **Fintech**, **Internet Services**, and **Healthcare** offer the highest earning potential
  - Consider industry context when planning career moves or tech offerings
- **Embrace AI Coding Tools**
  - With widespread adoption, AI tools are reshaping developer workflows
  - Upskilling in AI-assisted development is now essential
- **Rethink Legacy Systems**
  - Languages like **Java** and **C** show declining interest despite heavy usage
  - Organizations should plan transitions to modern, desirable stacks (e.g., Rust, Go)
- **Support Developer Diversity**
  - Large cohorts of students and junior developers entering the field
  - Opportunities for mentoring, inclusive hiring, and early-career support

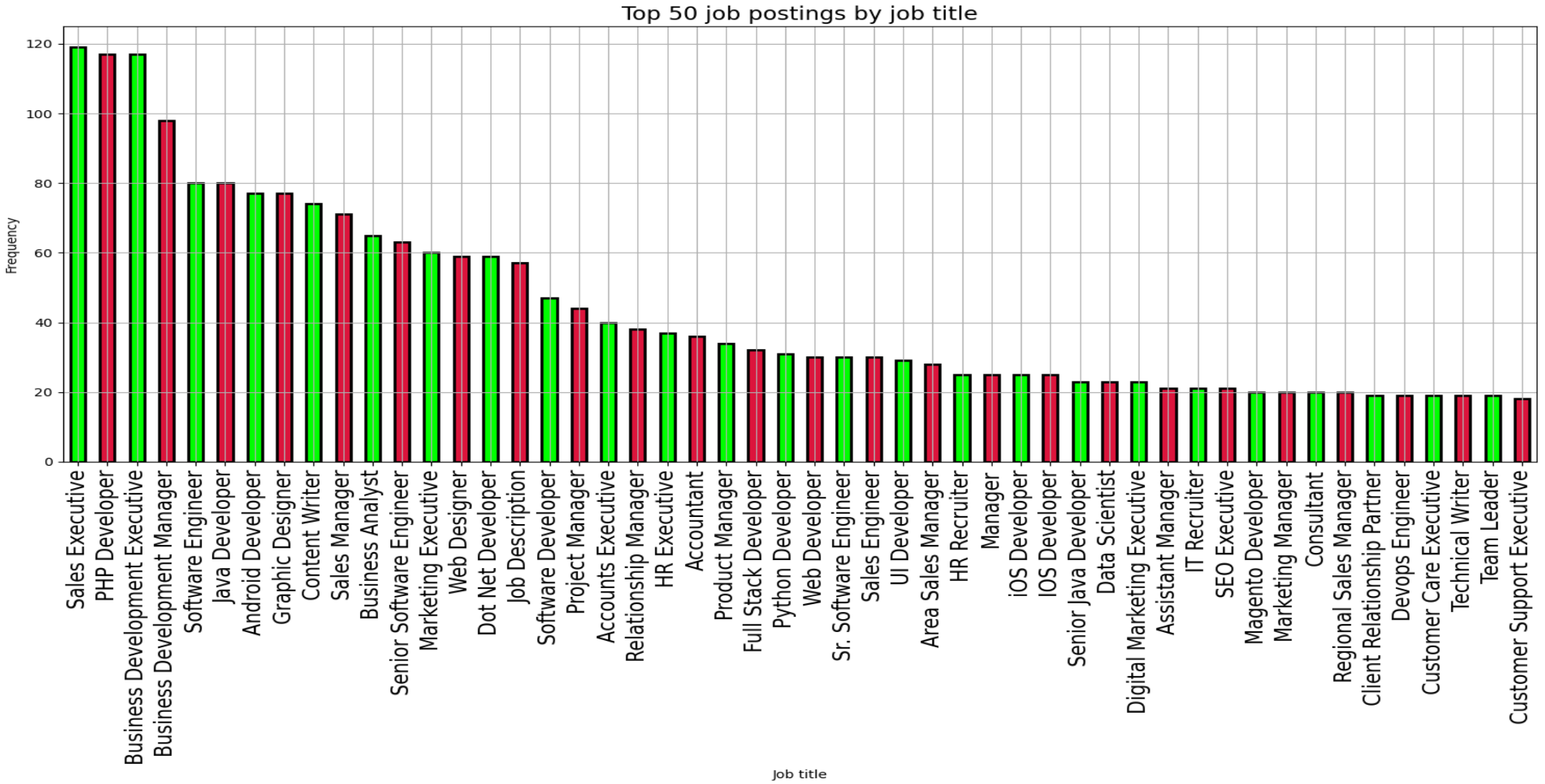
CONCLUSION

- **Modern stacks win developer mindshare**:
PostgreSQL, Python, TypeScript, and React lead both in current use and future interest — ideal technologies for long-term investment.
- **Developer satisfaction matters**:
Languages like Groovy, Dart, and Go score high in satisfaction, suggesting emerging opportunities beyond traditional stacks.
- **AI tools are mainstream**:
AI is now a core part of the developer workflow — especially for writing, debugging, and searching, reshaping productivity expectations.
- **Industry and compensation alignment**:
Fintech, Internet Services, and Banking offer top compensation — key for talent retention and career planning.
- **A diverse, global talent pool**:
Young, highly educated developers from the US, Germany, and India dominate — companies should tailor outreach, tools, and culture accordingly.
- **Strategic imperative**:
Embrace modern, open-source, and AI-enhanced tools to attract talent, future-proof systems, and align with evolving developer expectations.

APPENDIX

- Include any relevant additional charts, or tables that you may have created during the analysis phase.

JOB POSTINGS



POPULAR LANGUAGES

In Module 1 you have collected the job postings data using web scraping in a file named "popular-languages.csv". Present that data using a bar chart here. Order the bar chart in the descending order of salary.