



Amazon Q Developer

The most capable generative AI-powered
assistant for software development

Ayman Ishimwe
Solutions Architect, AWS

Today's agenda

Customer needs for generative AI

Service overview

Key features

Customers

Getting started



The median developer spends less than one hour a day writing code¹

01/08
HOURS PER DAY

05/40
HOURS A WEEK

¹ Software.com, "Global Code Time Report," January 2022.

Key needs for innovating faster



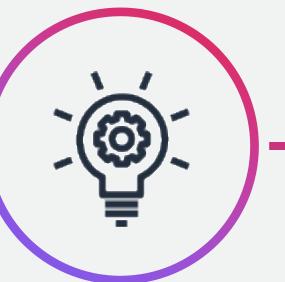
BUILD

Need to increase development velocity, but developers spend the majority of time on undifferentiated tasks instead of coding



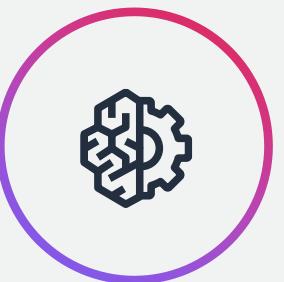
OPERATE

Need an efficient way to manage and optimize AWS cloud environments, resources, configurations, and costs across multiple accounts and services



TRANSFORM

Need to modernize legacy workloads faster and reliably to save costs, improve performance, and drive innovation.



DATA & AI

Need to leverage data to quickly and easily build analytics, AI/ML, and generative AI applications

The Importance of speed across the SDLC

- Rapid innovation
- Accelerating time-to-market
- Scaling with agility
- Reducing costs



Amazon Q Developer
advantage



Quality



Efficiency



Speed



Cost





Amazon Q

Reinvent work with AWS generative AI–powered assistant

Knowledgeable of
your company,
code and systems

Available
wherever you
work

Superior
generative AI
performance on tasks

Generative AI stack



APPLICATIONS THAT USE LLMs AND OTHER FMs



Amazon Q
Business



Amazon Q
Developer



Amazon Q in
QuickSight



Amazon Q in
Connect

TOOLS TO BUILD WITH LLMs AND OTHER FMs



Amazon Bedrock

Guardrails

Agents

Customization Capabilities

INFRASTRUCTURE FOR FM TRAINING AND INFERENCE



GPUs



AWS
Trainium



AWS
Inferentia



Amazon
SageMaker



Amazon EC2
UltraClusters



Elastic Fabric
Adapter (EFA)



Amazon EC2
Capacity Blocks

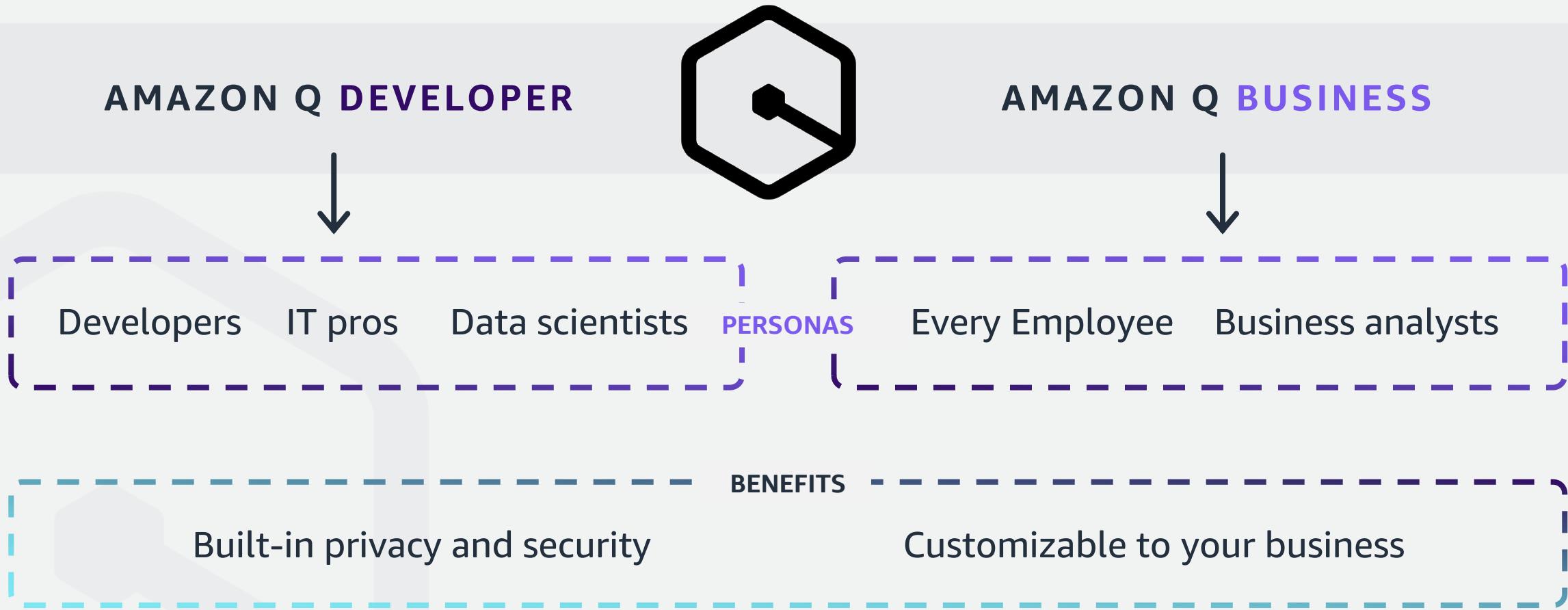


AWS
Nitro



AWS
Neuron

Amazon Q, a generative AI-powered assistant from AWS



Amazon Q Developer



Helps developers and IT professionals build faster across the entire software development lifecycle (SDLC)

Most accurate coding recommendations

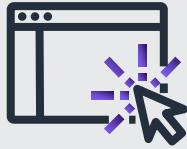
Agents can autonomously help you implement features, refactor code, perform software upgrades and more

Amazon Q is an expert on AWS and optimizing AWS environment

Best-in-class security vulnerability scanning and remediation

Amazon Q is built with security and privacy in mind from the start, making it easier for organizations to use generative AI safely.

Amazon Q is available where you do your work



AWS consoles



IDEs



AWS documentation



AWS Console
Mobile Application



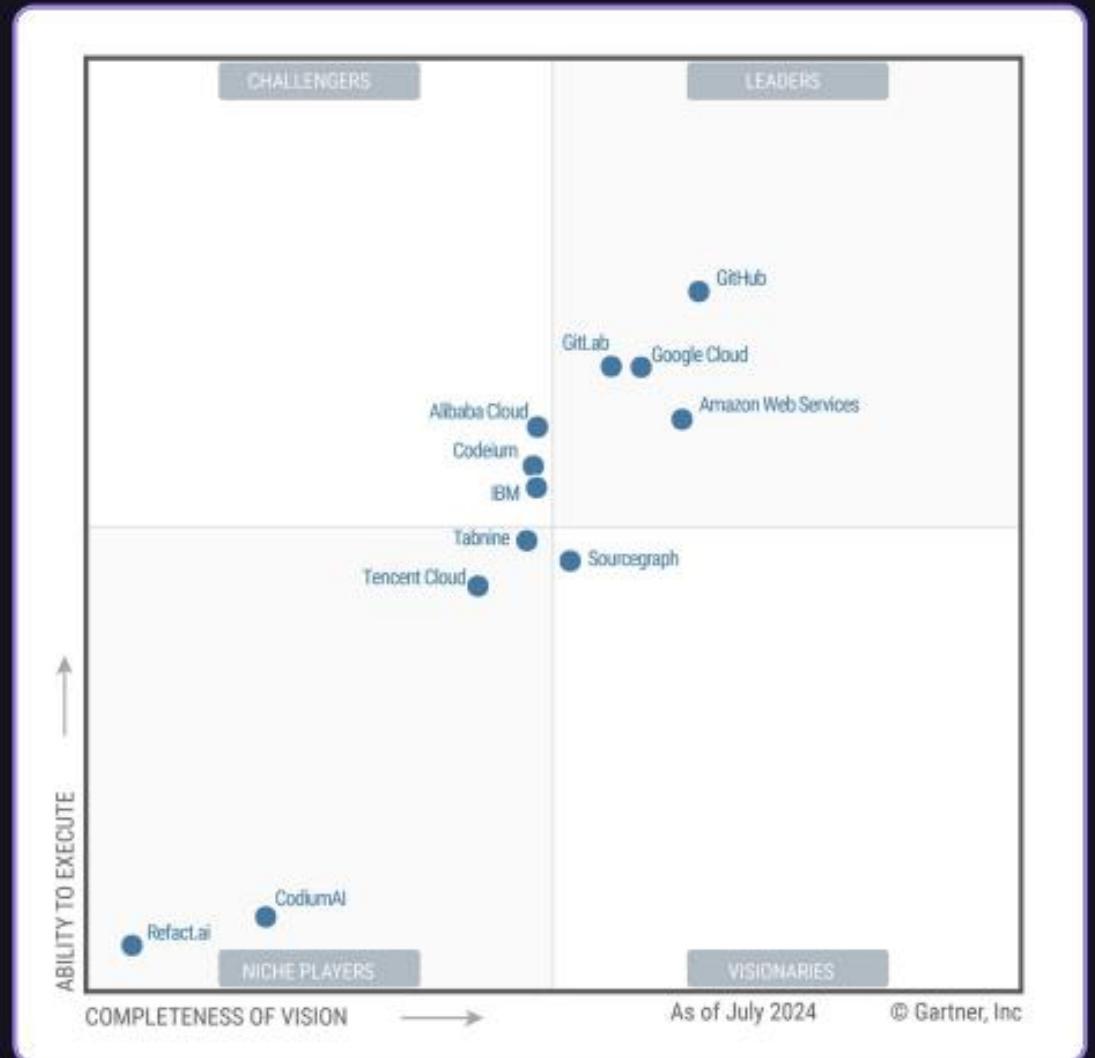
Slack and Teams
(through AWS Chatbot)



GitLab Duo with
Amazon Q

AWS recognized as a leader

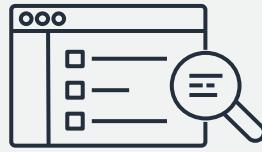
2024 Gartner® Magic Quadrant™ for AI Code Assistants



Source: Gartner, Magic Quadrant for DevOps Platforms, Keith Mann, Thomas Murphy, Bill Holz, George Spafford, September 2024

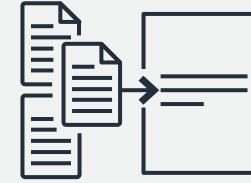


Amazon Q Developer top capabilities



Build faster

Accelerate tasks across the entire SDLC with generative AI-powered capabilities and agents



Operate on AWS

Get expert guidance for building on AWS, managing and optimizing AWS cloud resources, and diagnosing and troubleshooting errors.



Transform workloads

Modernize .NET, mainframe, VMware, and Java workloads at enterprise-scale to optimize processes and reduce costs



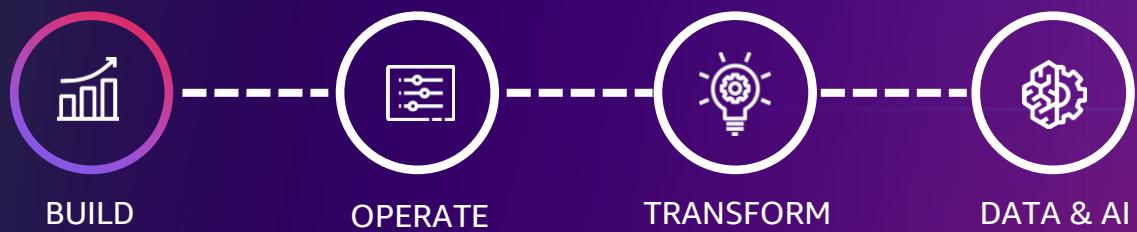
Leverage Data and AI

Get guidance to quickly and easily build analytics, AI/ML, and generative AI applications.

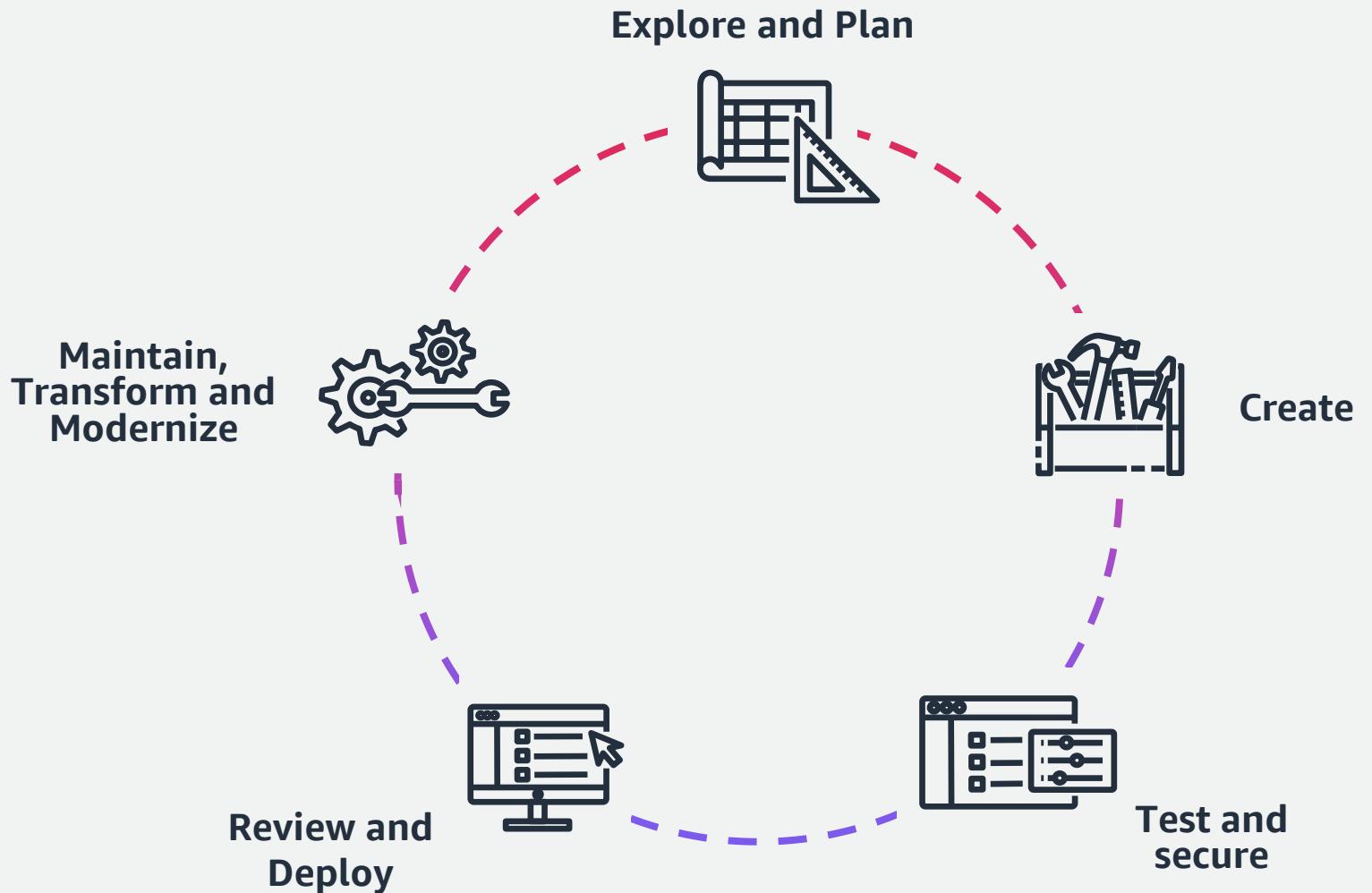


Faster Innovation

Build faster



Where are developers spending time in the SDLC?



Build faster across the SDLC



Explore and plan

- Onboard to new projects faster
- Plan new features
- Understand how to use AWS APIs
- Ask questions about your internal code bases



Create

- Build new apps or features
- Integrate APIs
- Refactor and modernize existing apps
- Generate IaC



Test and Secure

- Improve test coverage with unit test generation
- Scan and remediate security vulnerabilities



Review and Deploy

- Automate code reviews
- Assess deployment risk
- Generate documentation

Explore and plan

DEMO

The screenshot shows the AWS Console Home page with a dark theme. At the top, there's a navigation bar with the AWS logo, a 'Services' dropdown, a search bar containing 'Search', and a keyboard shortcut '[Option+S]'. On the right of the search bar are icons for a window, a bell, a question mark, a gear, and dropdown menus for 'Oregon' and 'Demo'. Below the navigation is a 'Reset to default layout' button and an orange '+ Add widgets' button.

The main content area is divided into several sections:

- Recently visited:** IAM, CloudWatch, Lambda, S3, API Gateway.
- Welcome to AWS:** Getting started with AWS (Learn the fundamentals and find valuable information to get the most out of AWS), What's new with AWS? (Discover new AWS services, features, and Regions).
- Explore AWS:** Build Web and Mobile Apps Faster, Amazon Bedrock is Now Gener... (Amazon Bedrock is a fully managed service that offers leading foundati...), AWS Support (Save time and move faster with).
- Build a solution:** Start building with simple wizards and automated workflows. Options include Launch a virtual machine (With EC2 (2 mins)), Start migrating to AWS (With AWS MGN (2 mins)), Register a domain (With Route 53 (3 mins)), Host a static web app (With AWS Amplify Console (2 mins)), Start a development project, and Build SQL Server on AWS.

At the bottom of the page are links for CloudShell, Feedback, and legal notices: © 2023, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

Ask questions and get guidance from 17 years of AWS knowledge, best practices, and well-architected advice

Create a business-expert application using your data, systems, and more.



Create

DEMO

The screenshot shows the AWS Code IDE interface. On the left is the Explorer sidebar with project files like .mvn, .vscode, src (containing main/java/com/example/project with Calculator.java and test/java/com/example/project with CalculatorTests.java), target, .gitignore, pom.xml, README.md, and summary.md. Below the sidebar are sections for OUTLINE, TIMELINE, JAVA PROJECTS, and MAVEN. The main area displays two tabs: Calculator.java and CalculatorTests.java. The Calculator.java tab shows the following code:

```
26 // Function to multiply numbers
27 public Double multiply(Double a, Double b) {
28     /**
29      * Multiplies two numbers
30      * @param a the first number
31      * @param b the second number
32      * @return the product of a and b
33      */
34     return a * b;
35 }
36
37 // Function to divide numbers
38 public Double divide(Double a, Double b) {
39     /**
40      * Divides two numbers
41      * @param a the first number
42      * @param b the second number
43      * @return the quotient of a and b
44      */
45     return a / b;
46 }
47
48 // Function to square a number
49
50
51 }
```

Below the code editor are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, PORTS, and CODE REFERENCE LOG. A status bar at the bottom says "Your organization controls whether suggestions include code with references. To update these".

Generate code

Explain code

Engage with
Inline chat

Help understand
your code base

Customizable
to your code



Create: Customize code recommendations

The screenshot shows a Java code editor with the file `FoodDeliveryFulfillment.java`. The code processes unassigned deliveries around a driver's current location. A yellow callout bubble labeled "Natural language prompt" points to the first line of code: `// Process a list of unassigned food deliveries around the driver's current location`. Another yellow callout bubble labeled "Generated in-line code based on your company's internal code bases" points to the line `boolean isAssigned = deliveryService.assignDeliveryToDriver(delivery, driver);`.

```
1 // Process a list of unassigned food deliveries around the driver's current location
2
3 package anycompany.fooddelivery.fulfillment;
4 import anycompany.delivery.Delivery;
5 import anycompany.delivery.DeliveryService;
6 import anycompany.driver.Driver;
7 import anycompany.driver.DriverLocationService;
8
9 public class FoodDeliveryFulfillment {
10     private DeliveryService deliveryService;
11     private DriverLocationService driverLocationService;
12
13     //Process all the unassigned deliveries
14     public void processUnassignedDeliveries() {
15         List<Delivery> unassignedDeliveries = deliveryService.getUnassignedDeliveries();
16         //Iterate over all the unassigned deliveries and assign them to a driver
17         for (Delivery delivery : unassignedDeliveries) {
18             //Get the nearest drivers for delivery location
19             List<Driver> drivers = driverLocationService.getDriver(delivery.getLocation());
20             for (Driver driver : drivers) {
21                 //Assign delivery to driver and send notification
22                 boolean isAssigned = deliveryService.assignDeliveryToDriver(delivery, driver);
23                 if (isAssigned) {
24                     deliveryService.notifyDelivery(delivery);
25                     driverLocationService.notifyDriver(driver);
26                     break;
27                 }
28             }
29         }
30     }
31
32     ...
33 }
34 }
```

Generate code based on your private repositories

Use Amazon Q Developer chat in the IDE to ask questions about your internal code base

Help understand your internal code base



Create: Advanced Feature Development: Q Developer agent for software development

```
AMAZON Q (PREVIEW) CHAT
Chat + ...

amazon-q-voting-app
app.py (Working Tree) x
...
52 52 |     return str(votes)
53 53 |
54 54 | @app.route('/')
55 55 | def home():
56 56 |     return '<h1>Welcome to the Voting App</h1><p><b>To vote, you can call the following</b></p>'
57 57 |
58 58 | @app.route('/api/outback')
59 59 | def outback():
60 60 |     string_votes = readvote("outback")
61 61 |     votes = int(string_votes)
62 62 |     votes += 1
63 63 |     string_new_votes = updatevote("outback", votes)
64 64 |     return string_new_votes
65 65 |
66 66 | @app.route('/api/bucadibeppe')
67 67 | def bucadibeppe():
68 68 |     string_votes = readvote("bucadibeppe")
69 69 |     votes = int(string_votes)
70 70 |     votes += 1
71 71 |     string_new_votes = updatevote("bucadibeppe", votes)
72 72 |     return string_new_votes
73 73 |
74 74 | @app.route('/api/ihop')
75 75 | def ihop():
76 76 |     string_votes = readvote("ihop")
77 77 |     votes = int(string_votes)
78 78 |     votes += 1
79 79 |     string_new_votes = updatevote("ihop", votes)
80 80 |     return string_new_votes
81 81 |
82 82 | @app.route('/api/chipotle')
83 83 | def chipotle():
84 84 |     string_votes = readvote("chipotle")
85 85 |     votes = int(string_votes)
86 86 |     votes += 1
87 87 |     string_new_votes = updatevote("chipotle", votes)
88 88 |     return string_new_votes
89 89 |
90 90 | @app.route('/api/getvotes')
91 91 | def getvotes():
92 92 |     string_outback = readvote("outback")
93 93 |     string_ihop = readvote("ihop")
94 94 |     string_bucadibeppe = readvote("bucadibeppe")
95 95 |     string_chipotle = readvote("chipotle")
```

Hi, I'm Amazon Q. I can answer your software development questions. Ask me to explain, debug, or optimize your code. You can enter / to see a list of quick actions.

Try Examples:

Explain selected code How can Amazon Q help me?

/dev refactor the code in the app.py file to remove redundant code

Use of Amazon Q is subject to the [AWS Responsible AI Policy](#).



Q Developer Agent writes and implements entire application features in minutes right from a simple natural language prompt in your IDE

Interactive step-by-step instructions, code, tests, and best practices

Document

```
flaskr
flaskr.py

59     @app.route('/')
60     def show_entries():
61         db = get_db()
62         cur = db.execute('SELECT title, text FROM entries ORDER BY id DESC')
63         entries = cur.fetchall()
64         return render_template('show_entries.html', entries=entries)
65
66
67     @app.route('/add', methods=['POST'])
68     def add_entry():
69         if not session.get('logged_in'):
70             abort(401)
71         db = get_db()
72         db.execute('INSERT INTO entries (title, text) VALUES (?, ?)', [request.form['title'], request.form['text']])
73         db.commit()
74         flash('New entry was successfully posted')
75         return redirect(url_for('show_entries'))
76
77
78     @app.route('/login', methods=['GET', 'POST'])
79     def login():
80         error = None
81         if request.method == 'POST':
82             if request.form['username'] != app.config['USERNAME']:
83                 error = 'Invalid username'
84             elif request.form['password'] != app.config['PASSWORD']:
85                 error = 'Invalid password'
86             else:
87                 session['logged_in'] = True
88                 flash('You were logged in')
89                 return redirect(url_for('show_entries'))
90             return render_template('login.html', error=error)
91
92
93
94     @app.route('/logout')
95     def logout():
96         session.pop('logged_in', None)
97         flash('You were logged out')
98         return redirect(url_for('show_entries'))
```

Agent (/docs) automates documentation

Onboard to new code bases faster

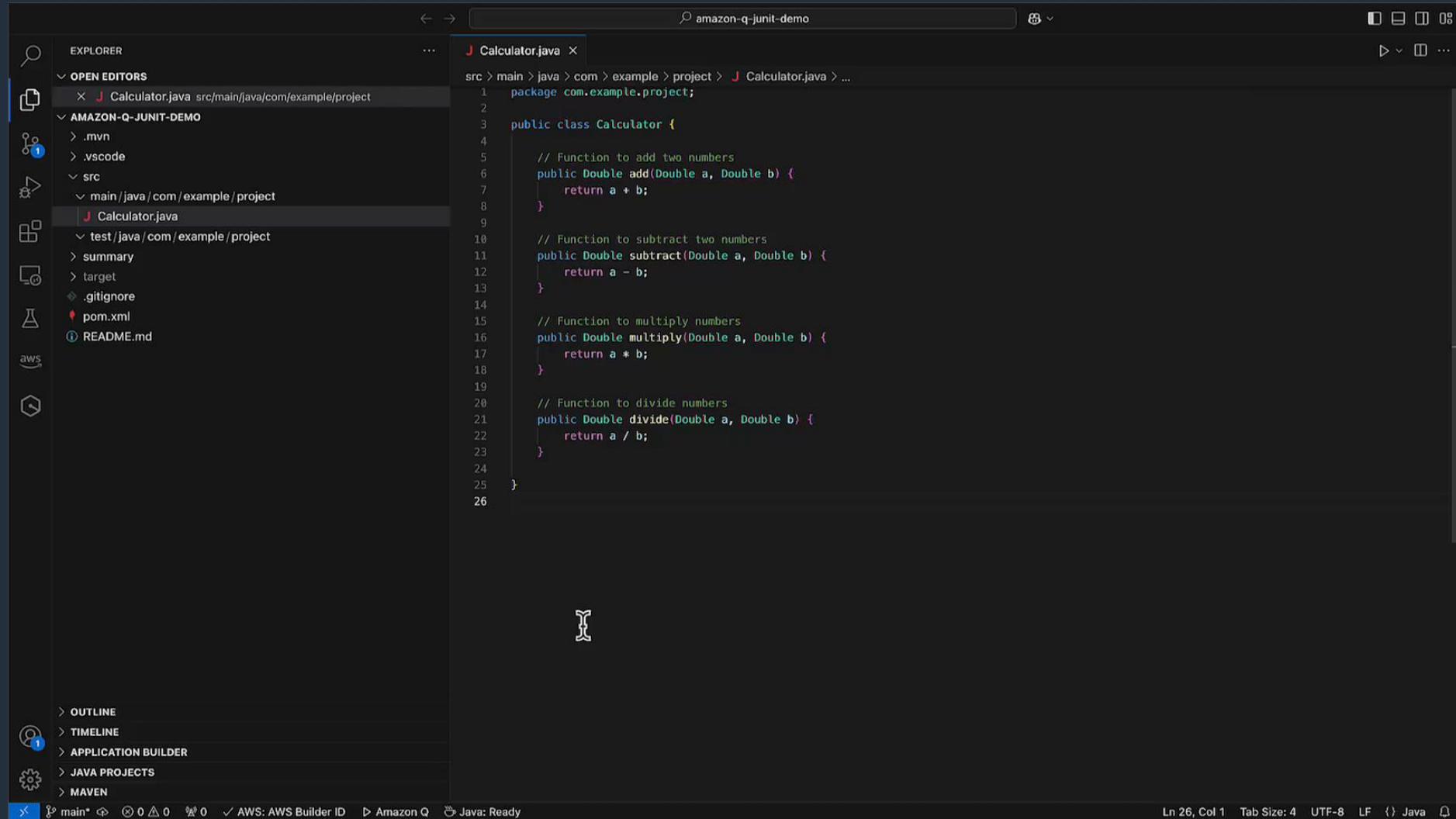
Keep projects well documented for team mates

Save hours while following code quality best practices



Test

DEMO



The screenshot shows the AWS Dev Studio interface with a Java project named "amazon-q-junit-demo". The "Calculator.java" file is open in the editor, displaying four methods: add, subtract, multiply, and divide. The code is as follows:

```
1 package com.example.project;
2
3 public class Calculator {
4
5     // Function to add two numbers
6     public Double add(Double a, Double b) {
7         return a + b;
8     }
9
10    // Function to subtract two numbers
11    public Double subtract(Double a, Double b) {
12        return a - b;
13    }
14
15    // Function to multiply numbers
16    public Double multiply(Double a, Double b) {
17        return a * b;
18    }
19
20    // Function to divide numbers
21    public Double divide(Double a, Double b) {
22        return a / b;
23    }
24
25 }
26
```

The left sidebar shows the project structure with "src/main/java/com/example/project" expanded, showing ".mvn", ".vscode", and "Calculator.java". The bottom status bar indicates "AWS: AWS Builder ID" and "Java: Ready".

Agent (/test) automates end-to-end process of generating unit tests

Ship code more reliably with better test coverage

Stay in control with built in checks



Secure

The screenshot shows the AWS Amazon Q developer interface. On the left, there's a sidebar with various icons and a search bar. The main area has a dark theme with several tabs open. One tab shows a list of code issues, including:

- Critical: 0 issues
- High: 4 issues (CWE-539 - Package Vulnerability in requirements.txt)
- Medium: 2 issues (CWE-352 - Package Vulnerability in requirements.txt, CWE-78 - AWS api logging disabled in cdk_api_logging_disable...)
- Low: 1 issue (CWE-89 - SQL injection in sql_injection.py)
- Info: 0 issues

The requirements.txt file is shown in a code editor window, listing dependencies like alembic, amqp, Flask, Flask-Mail, Flask-Migrate, Flask-RESTful, Flask-Script, Flask-SQLAlchemy, psycopg2-binary, pydantic, SQLAlchemy-i18n, SQLAlchemy-Utils, and Werkzeug.

A modal window titled "Code Issue Details" is open for the "CWE-539 - Package Vulnerability" in requirements.txt. It shows the following details:

- Severity:** High
- Description:** application sets SESSION_PERMANENT = True. The application does not access or modify the session at any point during a request. 4. SESSION_REFRESH_EACH_REQUEST enabled (the default). 5. The application does not set a Cache-Control header to indicate that a page is private or should not be cached.
- This happens because vulnerable versio...**
- Relevant link:** <https://github.com/advisories/GHSA-m2qf-hxjv-5gpq>
- Common Weakness Enumeration (CWE):** CWE-539
- File path:** requirements.txt [Ln 3]
- Suggested code fix preview:** 3. Flask==2.3.2
- Why are we recommending this?** Update installed software package Flask to version 2.3.2
- Action buttons:** Accept Fix, Regenerate Fix, Explain, Ignore, Ignore All

Scan code for hard-to-detect vulnerabilities

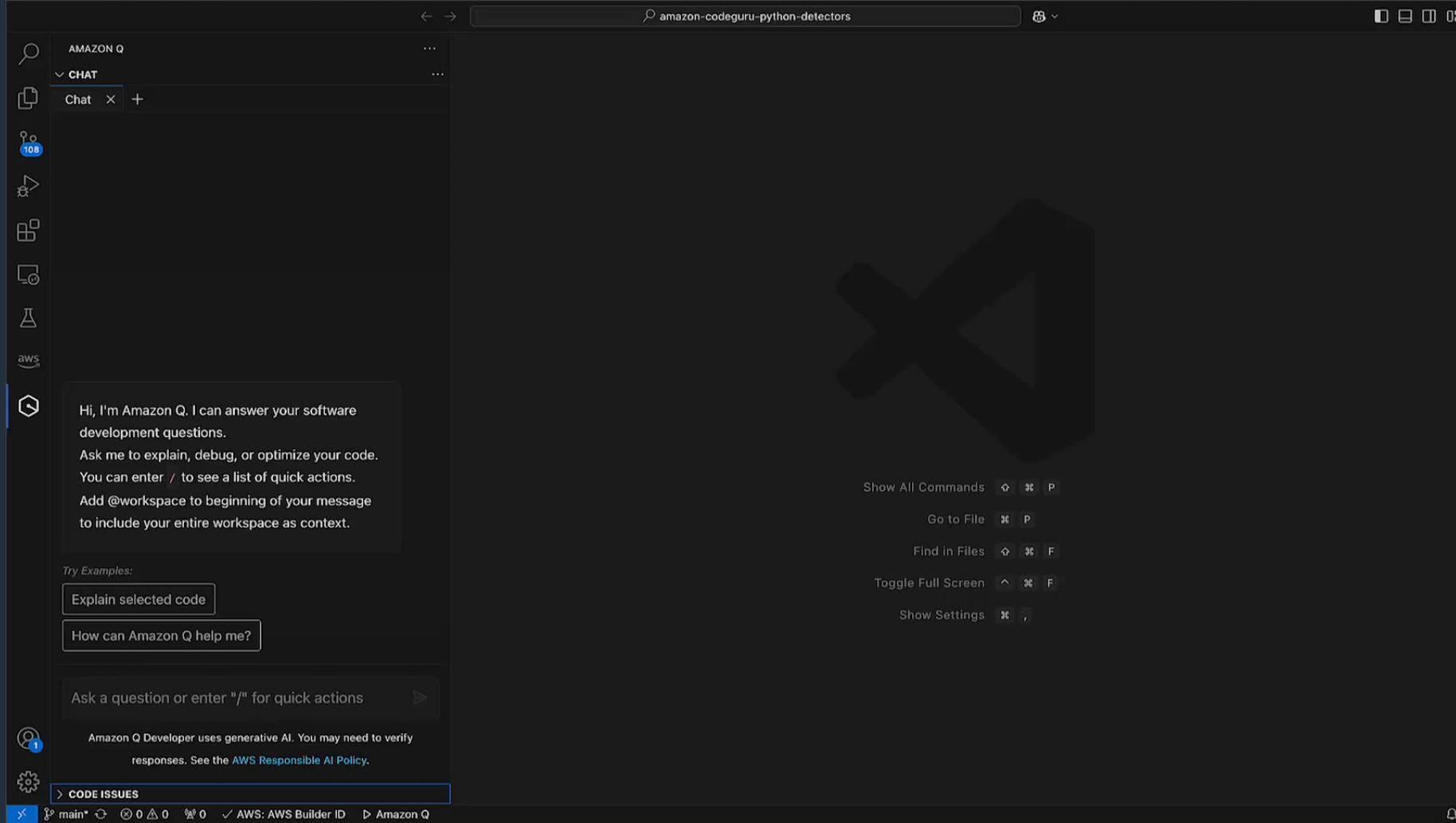
Generate code suggestions and recommendations in near real time

Outperforms leading publicly benchmarkable tools on detection across most of the popular programming languages



Review and deploy

DEMO



Agent (/review)
automates code
reviews

Automatically
detect and resolve
common code
quality issues in
minutes vs hours

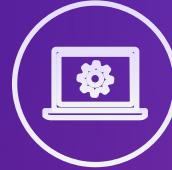
Ensure code
review consistency



Metrics and impact summary



Developers report time savings across a range of tasks



Suggestion acceptance rates up with better models and customization



Increased velocity of code changes and deployments



Reduced context switching with the ability to query code base and general knowledge from IDE



Amazon Q
Developer

UP TO
80%

Acceleration of
development tasks

Based on an internal study

UP TO
60%

Code acceptance
rate

With customization as recently reported by
National Australia Bank

UP TO
40%

Developer
productivity increase

As recently reported by Eviden



Advanced AI agent capabilities embedded into
the familiar DevSecOps workflows developers
use daily.

Better together: GitLab and AWS

We're teaming up to:

- Help customers deliver innovation at scale
- Empower teams from idea to delivery and beyond
- Support the needs of the modern software stack
- Accelerate every step of the software development lifecycle



Friction throughout the SDLC



**Tool
integrations**



**Technology
configurations**



**Scaling app
development**



**Collaboration
at scale**



**Tool
operations**



Security



**Enforcing best
practices across SDLC**





A seamless AI-powered developer experience that unifies software development, security, and deployment workflows to help organizations ship more secure software faster.

Availability in preview

- A bundle of GitLab's Ultimate subscription and GitLab Duo with Amazon Q
- Available for Self-Managed Ultimate customers



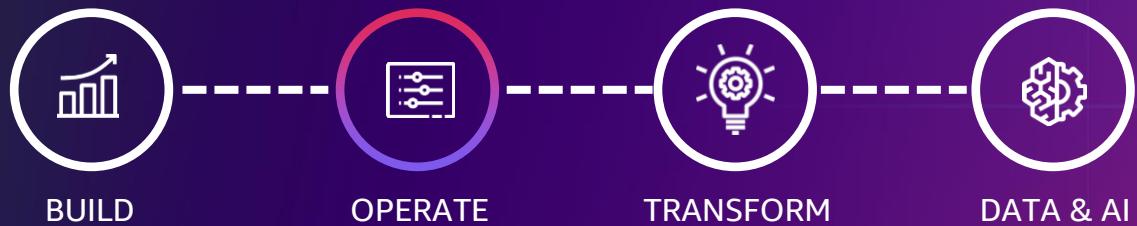
The most comprehensive AI-powered DevSecOps platform



Amazon Q

The most capable generative AI-powered assistant for software development

Operate on AWS



Operate on AWS



Expert guidance for building on AWS

Ask Amazon Q about AWS services, capabilities and architecture solutions — our AWS expert in best practices, documentation, and implementations.



Manage and optimize AWS cloud resources

List and take action on your AWS account resources, configurations, analyze billing information and usage trends through intuitive visualizations and guided actions.



Diagnose and troubleshoot errors

When something is wrong in your operations, minutes matter. Investigate, troubleshoot, and remediate operational issues quickly.

What can you ask Amazon Q?

What is wrong with my Lambda function?



How can I leverage agents?



What are my forecasted costs for the rest of the year?

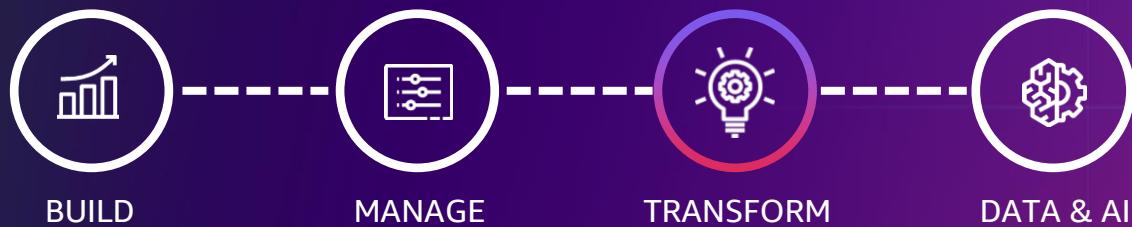
Why is my API gateway seeing errors?

How much am I spending on data transfer?



Show me alarms for this S3 bucket

Transform workloads



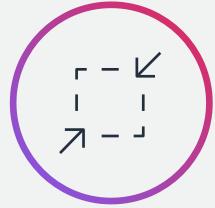
Transformation challenges

CHALLENGES



Slow modernization

Complex legacy systems
Labor-intensive processes



Limited scalability

Sequential processes
Limited collaboration



Suboptimal quality

Scarce modernization expertise
Scarce legacy language expertise

Transformation solutions

SOLUTIONS



Acceleration

Simplification
Generative AI automation



Scale & savings

Parallelized processes
Multifunctional collaboration



High quality

Domain-expert AI agents
Tailored transformation

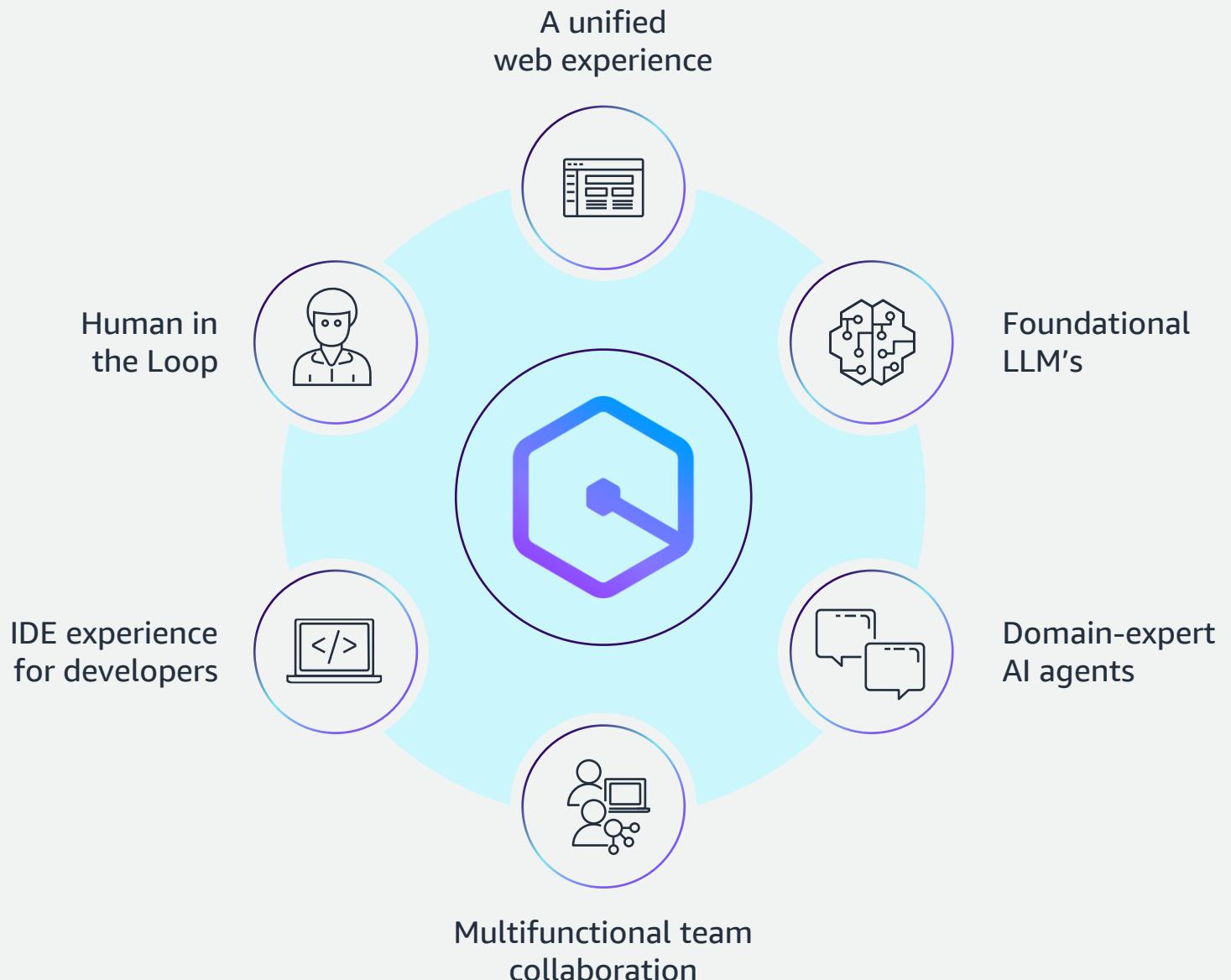
Amazon Q Developer transformation capabilities

The first generative AI-powered assistant for transforming Java, .NET, mainframe, and VMware workloads

The screenshot shows the Amazon Q Developer: Transform interface. On the left, the "Job Plan" section lists tasks: Kick off modernization, Analyze code, Generate documentation, Decompose code (with a completed task "Decompose into doma..."), Plan migration wave, Review planning res..., and Refactor code. On the right, the "View decomposition results" section displays a dependency graph. The graph consists of several nodes: Billing (blue), Accounts (light blue), Credit cards (light blue), Unassigned (orange), ETL and load (orange), Common component (blue), Admin (green), and Transactions (green). Arrows indicate dependencies between these components, such as Billing pointing to Accounts, Credit cards, and Admin; Accounts pointing to Common component; Unassigned pointing to ETL and load; and ETL and load pointing to Common component.

Q Developer: Transform

CAPABILITIES



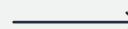
Q Developer: Transform

WORKLOADS



.NET Framework
applications
on Windows

Analysis Planning
Refactoring
Compatibility tests

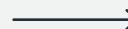


Cross-platform .NET
applications on Linux



COBOL applications
on mainframes

Documentation Wave
planning Decomposition
Refactoring



Java applications
on AWS



VMware workloads
on-premises

Discovery Wave planning
Network automation
Workload migration



VMware workloads
on Amazon EC2



Java applications
in v8, v11

Build/test v8/v11 Apply
knowledge base Build/test
v17Error fixing using genAI



Java applications
in v17

Up to
4 x
faster

Up to
40%
savings

Access to
**a range
of LLMs**





Amazon Q
Developer

1,000

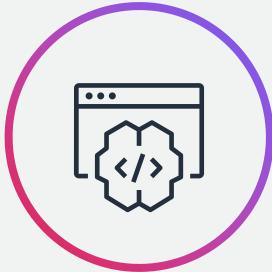
Java applications

2

Days

10

Minutes each,
on average



Amazon Q Developer Java transformation

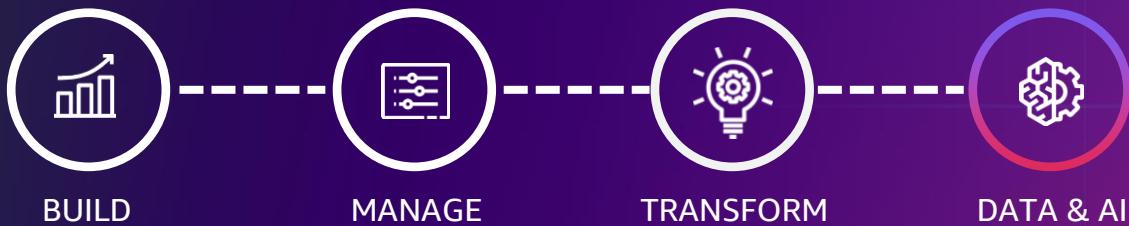
4,500

Years of development work saved

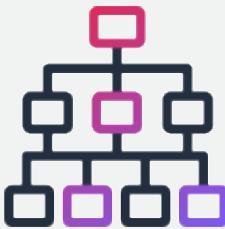
\$260M

Annual cost savings

Leverage data and AI



Build AI and data-driven applications



Analyze and integrate data

Use natural language for guidance as you build and integrate data pipelines, create SQL queries, and derive actionable insights to do more with your data.

Build ML models

Get help with data preparation, model training, and model deployment. Ask for code recommendations and debugging help in IDEs like VSCode, or step-by-step guidance in no-code tooling.

Leverage Data and AI

Ask about available governed data for projects; ramp up quickly on collaborations; and get help authoring code, generating SQL, integrating data, troubleshooting, and more in a unified environment.



FINANCIAL SERVICES

National Australia Bank is one of the largest financial institutions in Australia.

"At NAB, teams across the bank are excited about how generative AI can transform our work, and we've found a great solution in Amazon Q Developer, a powerful generative AI-powered tool for our engineers and developers. The tool has seamlessly integrated advanced generative algorithms and tools into our development process, delivering unparalleled benefits like completing tasks faster, increasing productivity, and minimizing repetitive actions. So far, our developers **have accepted 50% of the code suggestions** made by Amazon Q Developer, and that number continues to increase as we scale. We look forward to seeing how Amazon Q Developer will continue to empower and inspire our engineers to upskill their technical expertise, delivering better service for customers."

Andrew Brydon,
Executive Chief Engineer, National Australia Bank



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.



IT SERVICES

Novacomp provides a diverse range of IT products and services while specializing in quality assurance automation, software testing services, and nearshore outsourcing.

"Modernizing applications at Novacomp has historically been a time-consuming task that is often deprioritized against other development initiatives. Our team turned to Amazon Q Code Transformation agent to help upgrade a project running in Java 8 to Java 17 with over 10,000 lines of code. This is a task that would typically take an expert over two weeks to manually complete, but Amazon Q seamlessly modernized our project in a matter of minutes. Since adopting Amazon Q across our organization, we have realized a **60% decrease** in average in our tech debt."

Gerardo Arroyo, CTO, Novacomp





FINANCIAL SERVICES

BPC helps more than 450 clients in 120 countries deliver innovative solutions in banking, payments, commerce, and mobility industries.

“Amazon Q Developer significantly reduces the amount of time spent on simple code and repetitive tasks. Now we can focus on the more intelligent, challenging, and high-quality code in addition to our security posture.”

Stanislav Krainov, Developer Team Lead, BPC



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

Signaturit Group

DIGITAL TRANSACTIONS

European company which provides complete, secure, and compliant digital transaction solutions, ensuring seamless, trusted processes for businesses users.

"Amazon Q Developer has introduced a new world for modernization at Signaturit and is a service that our developers are excited to use to continue to modernize applications across our portfolio"

- Jordi Martin, CTO, Signaturit



A U T O M O B I L E

Toyota North America drives innovation and sustainability, delivering quality vehicles

“The transformation capabilities of Amazon Q Developer have been a revolutionary solution for us. The powerful documentation, code analysis, and decomposition capabilities provided by AWS give us a clearer understanding of our application landscape. We’re able to generate detailed documentation of hundreds of COBOL modules, including context-aware information and persona-driven insights, in less than a day. Amazon Q Developer has done what many said was impossible.”

- Brian Kursar, CTO, Toyota Motor North America



A U T O M O T I V E

Volkswagen Digital Solutions inspires and transforms the future of IT in the Volkswagen Group

“The integration of GitLab Duo with Amazon Q Developer is a game-changer for our development teams and can help remove friction from their daily tasks, reduce feedback cycles, and improve our code quality at scale. By embedding advanced AI capabilities from autonomous agents in the GitLab workflows they know and love, we expect to see a lift in productivity across teams to help Volkswagen Digital Solutions release new features and applications at the pace our customers expect and deserve.”

- Vivien Kirchgessner, Product Owner at Volkswagen Digital Solutions



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.



Mercedes-Benz

A U T O M O T I V E - S O F T W A R E

MBition GmbH is a subsidiary of the Mercedes-Benz Group and works on developing software for all Mercedes-Benz cars.

"As an enterprise operating in the highly complex and regulated automotive environment, GitLab's new Amazon Q integration has the potential to accelerate our digital transformation journey while strengthening our security and compliance posture.

The seamless integration allows us to stay within the GitLab UI, making it extremely easy to use. The ability to create new branches and merge requests directly from a GitLab issue showcases the significant potential of Amazon Q.

This feature enables our embedded software engineers to concentrate fully on their core competencies, leaving common DevSecOps tasks to Q.

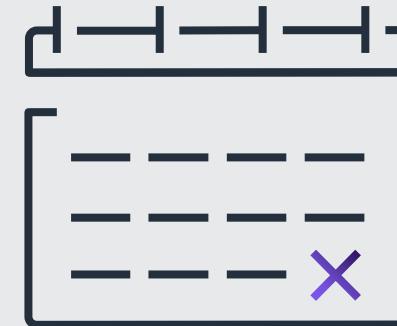
With this integration, we see the opportunity to focus on groundbreaking innovations while streamlining the exceptional work of our Mercedes-Benz and MBition engineers."

- Arne Altrichter, Chief Architect at MBition

Get started with Amazon Q Developer



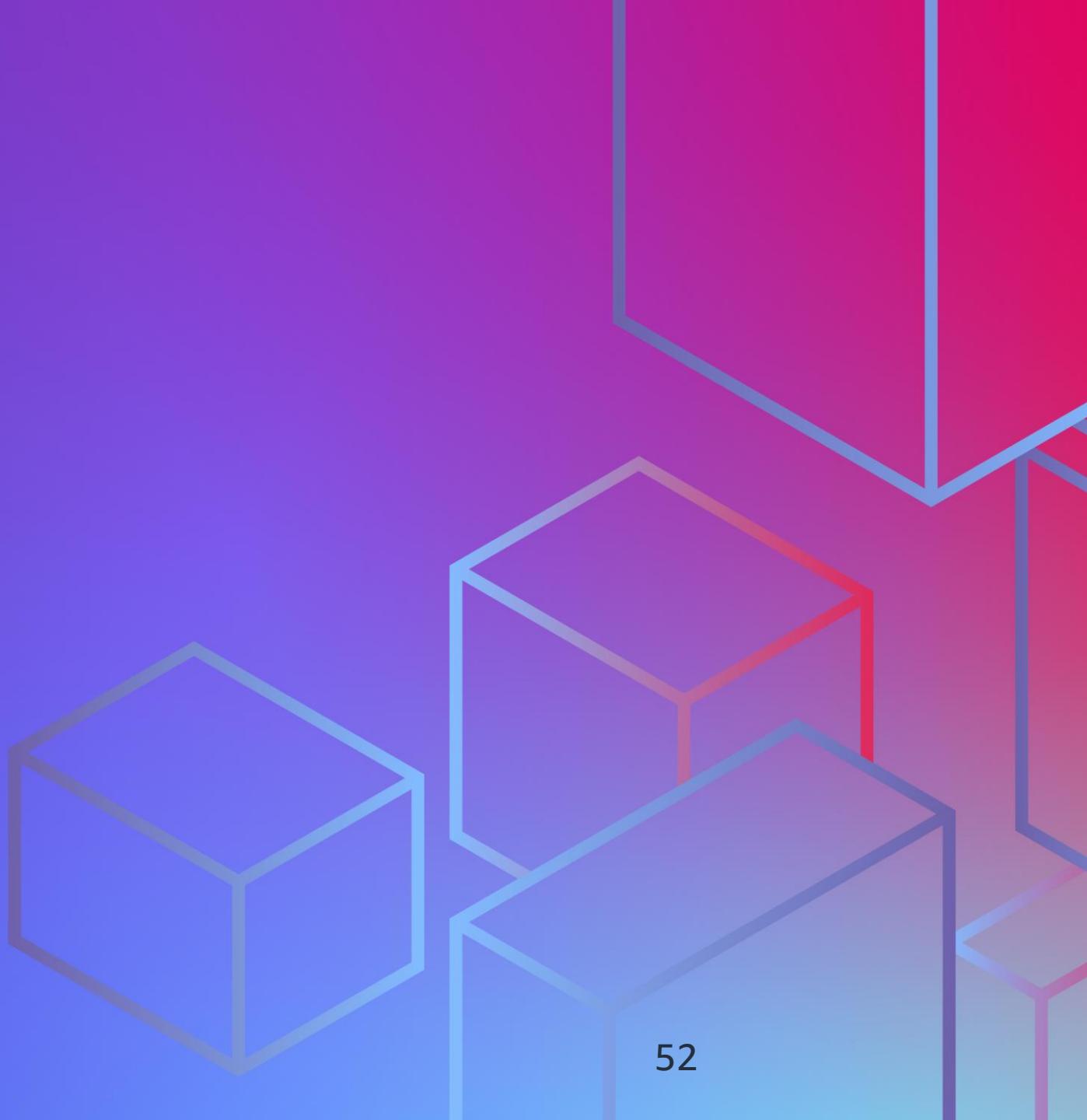
**Try Amazon Q
Developer today**



**Plan your
POC**

Thank you!

Ayman Ishimwe
Solutions Architect, AWS



Appendix



 **GitLab**^{Duo+} with  Amazon Q

AI-driven DevSecOps

Demos

- Faster Feature Development
- Shorter Feedback Loops
- Transform a legacy codebase

Faster feature development

- Go from feature spec in an issue to merge-ready code with a quick action
- Iterate directly from GitLab, using feedback in comments to accelerate development workflows from end-to-end

Idea to Merge Request (MR)

DEMO

The screenshot shows a Git commit history for a project named "micro-frontend". The commits are all from "Create Next App" and were authored by Kyle. Each commit was made 1 minute ago. The files listed are public, src/app, .eslintrc.json, .gitignore, README.md, next.config.ts, package-lock.json, package.json, postcss.config.mjs, tailwind.config.ts, and tsconfig.json. A note at the bottom states: "This is a Next.js project bootstrapped with `create-next-app`".

Name	Last commit	Last update
public	Initial commit from Create Next ...	1 minute ago
src/app	Initial commit from Create Next ...	1 minute ago
.eslintrc.json	Initial commit from Create Next ...	1 minute ago
.gitignore	Initial commit from Create Next ...	1 minute ago
README.md	Initial commit from Create Next ...	1 minute ago
next.config.ts	Initial commit from Create Next ...	1 minute ago
package-lock.json	Initial commit from Create Next ...	1 minute ago
package.json	Initial commit from Create Next ...	1 minute ago
postcss.config.mjs	Initial commit from Create Next ...	1 minute ago
tailwind.config.ts	Initial commit from Create Next ...	1 minute ago
tsconfig.json	Initial commit from Create Next ...	1 minute ago

Project information

- 1 Commit
- 1 Branch
- 0 Tags
- 199 KiB Project Storage

Actions

- README
- Auto DevOps enabled
- Add LICENSE
- Add CHANGELOG
- Add CONTRIBUTING
- Add Kubernetes cluster
- Add Wiki
- Configure Integrations

Created on
November 17, 2024



Shorter feedback loops

- Save developer time with AI-assisted code reviews for new merge requests
- Apply fixes directly from comments
- Maintain code quality based on engineering and security best practices

Code review

DEMO

The screenshot shows a code review interface for a merge request (MR) #116. The top navigation bar indicates the project is 'Initech / download-server' and the merge request is 'Merge requests / !116'. The title of the merge request is 'Update file UserHandler.java' with a branch 'ft-user-handler' being merged into 'main'. The 'Overview' tab is selected, showing '0' assignees, '1' commit, '1' pipeline, and '1' change. A prominent green button at the top left says 'Ready to merge!' with a checkmark icon. Below it are three checkboxes: 'Delete source branch' (checked), 'Squash commits' (unchecked), and 'Edit commit message' (unchecked). A note below states '1 commit and 1 merge commit will be added to main.' A blue button labeled 'Set to auto-merge' is visible. To the right, there's a section titled 'Add a to-do item' with a button. On the far right, there are sections for 'Assignees' (0, None - assign yourself), 'Reviewers' (0, None - assign yourself), 'Labels' (None), 'Milestone' (None), 'Time tracking' (No estimate or time spent), and 'Participants' (1, shown with a small profile icon).

In the main body, there's a 'Activity' section with a rich text editor toolbar above a comment input field. The input field says 'Write a comment or drag your files here...' and has a placeholder 'I'. Below the input field is a link 'Switch to rich text editing'. At the bottom, there's a checkbox 'Make this an internal note' and two buttons: 'Comment' and 'Close merge request'.



Transform a legacy codebase

- Upgrade a Java 8 or 11 codebase in clicks, directly from a GitLab project
- Ensure compatibility with modern toolkits, libraries, and frameworks
- Improve application security, performance, and reduce technical debt

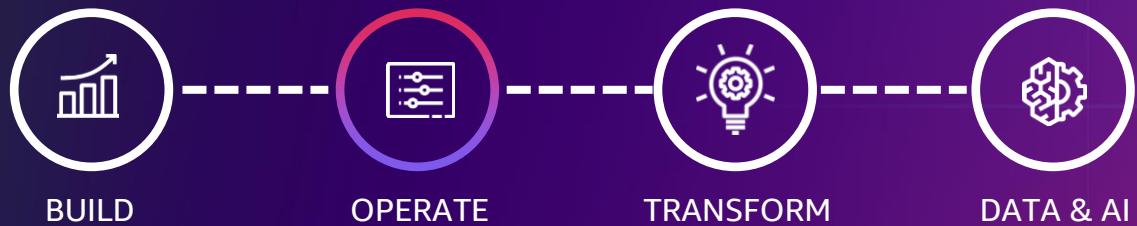
Transform workloads

DEMO

The screenshot shows a web-based application interface for managing workloads, specifically a 'New Issue' creation screen. The left sidebar contains a navigation menu with various project management options like 'Issues', 'Merge requests', 'Manage', 'Plan', and 'Requirements'. The main form is titled 'New Issue' and includes fields for 'Title (required)' (set to 'Upgrade Project to Java 17'), 'Type' (set to 'Issue'), and a rich text editor for 'Description'. Below the description area, there's a note about adding description templates and a checkbox for marking the issue as confidential. The right side of the form includes sections for 'Assignees' (set to 'Unassigned'), 'Weight' (input field 'Enter a number'), 'Epic' (dropdown 'Select epic'), 'Due date' (input field 'Select due date'), 'Milestone' (dropdown 'Select milestone'), 'Iteration' (dropdown 'Select iteration'), and 'Labels' (dropdown 'Select label').



Operate on AWS



Expert guidance for building on AWS

DEMO

The screenshot shows the AWS Console Home page with the Q feature integrated. At the top right, there's a vertical sidebar with three colored sections: purple, red, and pink. The purple section contains the text "Chat with Q seamlessly in the AWS console". The red section contains "Get accurate and actionable guidance to build on AWS". The pink section contains "Research and evaluate options to build on AWS". The main content area includes sections for "Recently visited", "Welcome to AWS", "Explore AWS", and "Build a solution", each with various links and icons.



Manage and optimize AWS resources

DEMO

The screenshot shows the AWS Billing and Cost Management home page on the left and the Amazon Q AI assistant on the right.

Billing and Cost Management Home:

- Cost summary:** Month-to-date cost **\$1,120.14** (↑ 3% compared to last month for same period), Last month's cost for same time period **\$1,088.25** (Jul 1–12).
- Total forecasted cost for current month:** **\$2,150.95** (↑ 10% compared to last month's total costs).
- View bill** button.

Cost breakdown: Bar chart showing monthly costs from March to August 2024. The chart is grouped by service: Amazon QuickSight, Amazon Elastic Container Service for Kubernetes, AWS Support (Business), Amazon Route 53, Amazon Virtual Private Cloud, and Others.

Month	Amazon QuickSight	Amazon Elastic Container Service for Kubernetes	AWS Support (Business)	Amazon Route 53	Amazon Virtual Private Cloud	Others	Total
Mar 2024	~1000	~100	~100	~100	~100	~100	~1800
Apr 2024	~1200	~100	~100	~100	~100	~100	~2500
May 2024	~1000	~100	~100	~100	~100	~100	~1800
Jun 2024	~1000	~100	~100	~100	~100	~100	~1800
Jul 2024	~1000	~100	~100	~100	~100	~100	~1800
Aug 2024	~1000	~100	~100	~100	~100	~100	~1800

Analyze your costs in Cost Explorer button.

Amazon Q AI Assistant:

- Hello!** I'm Amazon Q, your AWS generative AI assistant.
- Ask me anything about AWS services and features or choose a sample question below to start a conversation.
- Sample questions shown:
 - List my S3 buckets
 - What is the CLI command to list all the t3.micro instances in my account?
 - How can I deploy a containerized web application to AWS?
- I'm learning more every day. Help me improve by [providing feedback](#). Visit the [Amazon Q documentation](#) for more information.
- Additional info:** Amazon Q may retain chats to provide and maintain the service. Amazon Q stores and processes your data in the N. Virginia (us-east-1) Region, even if the AWS Management Console is set to a different AWS Region.
- Ask me anything about AWS** input field with character limit of 4000 characters. Use of Amazon Q is subject to the [AWS Responsible AI Policy](#).

Visualize, understand, and manage AWS cloud costs and usage

Forecast and optimize resource usage for managing cloud spend

Chat with Q to understand your spend

CloudShell Feedback



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Diagnose and troubleshoot errors

DEMO

The screenshot shows the AWS Lambda Functions console. At the top, there is a search bar with the placeholder "[Option+S]" and a "Search" button. Below the search bar, the breadcrumb navigation shows "Lambda > Functions". The main area displays a table titled "Functions (3)". A filter bar at the top of the table includes a search input "Filter by tags and attributes or search by keyword" with a placeholder "Runtime = java17" and a clear filters button. The table has columns for "Function name", "Description", "Package type", "Runtime", and "Last modified". A single row is visible, representing a function named "Calculator" which is a "Java function" packaged as a "Zip" file, running on "Java 17", and was last modified "5 minutes ago". The table also includes a header row with sorting and filtering icons. The status bar at the bottom of the browser window shows "Last fetched 7 seconds ago".

Diagnose and
troubleshoot
common AWS
console errors

Accelerate
research and
resolution

Investigate and remediate issues

DEMO



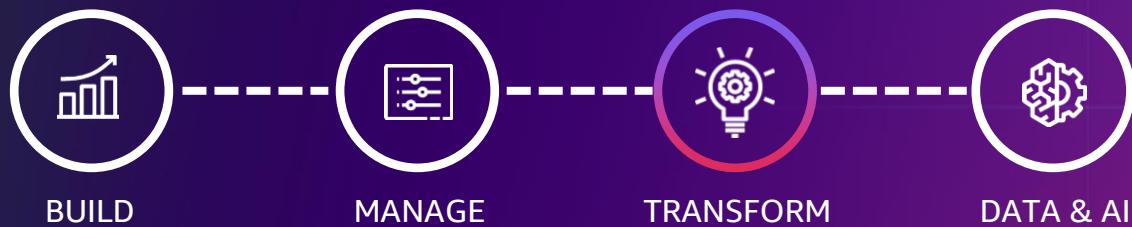
Get meaningful investigation hypotheses and guidance for remediation

Streamline investigation summaries and updates

Trigger investigations from alarms, metrics, or manually chatting with Q



Transform workloads



NEW

Amazon Q Developer transformation capabilities

for .NET porting

Accelerate the porting of hundreds of .NET applications to Linux using generative-AI transformation agents

IN PREVIEW

Accelerate porting of .NET applications by up to 4x

Deliver more projects, and lower costs by up to 40%

Enhance software quality, performance, & security



NEW

Amazon Q Developer transformation capabilities

for mainframe modernization

Accelerate large-scale mainframe application modernization using generative-AI transformation agents

IN PREVIEW

Speed up mainframe application modernization and migration processes

Reduce project complexity & enhance collaboration

Enhance application performance, security, and scalability



NEW

Amazon Q Developer transformation capabilities

for VMware migration and modernization

Accelerate large-scale migration and modernization of VMware workloads using generative-AI transformation agents

IN PREVIEW

Accelerate migration of VMware workloads

Lower costs and license fees

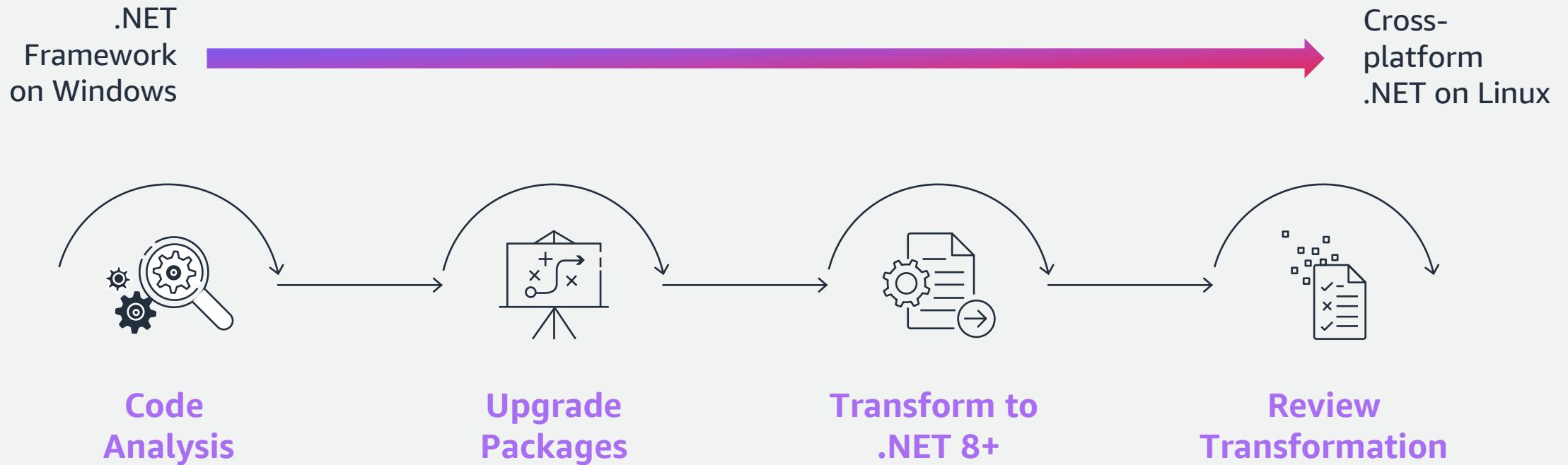
Enhance innovation, security, and resilience



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

Q Developer: Transform for .NET

The first generative AI-powered assistant for large-scale porting of .NET applications



Accelerate .NET porting

DEMO



Note: placeholder demo. The demo video is not final yet.



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

Modernize .NET application from Windows to Linux up to 4x faster and reduce costs by up to 40%

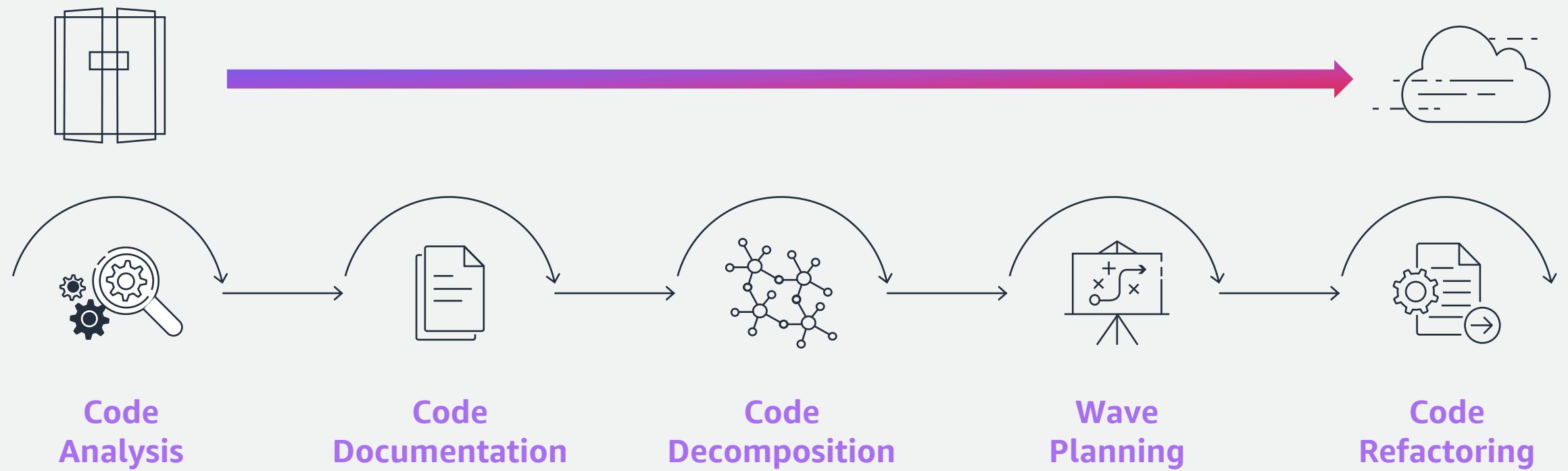
Port hundreds of .NET applications in parallel with generative AI agents

Boost application performance and security by moving off legacy operating systems

Drive more projects with consistency and collaboration

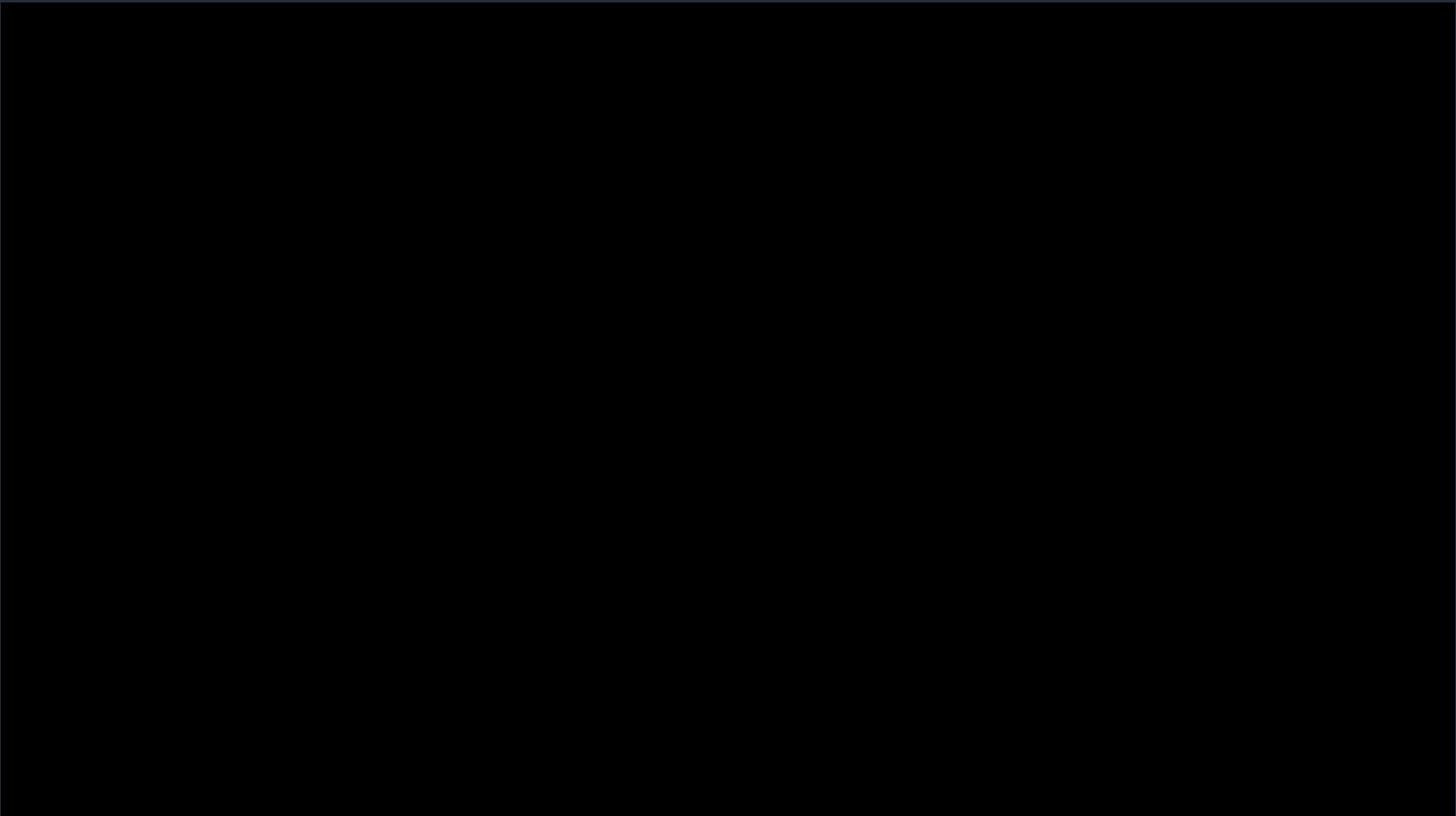
Q Developer: Transform for mainframe

The first generative AI-powered assistant for large-scale modernization and migration of mainframe applications



Accelerate mainframe modernization

DEMO



Note: placeholder demo. The demo video is not final yet.



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

Speed up
mainframe
application
modernization

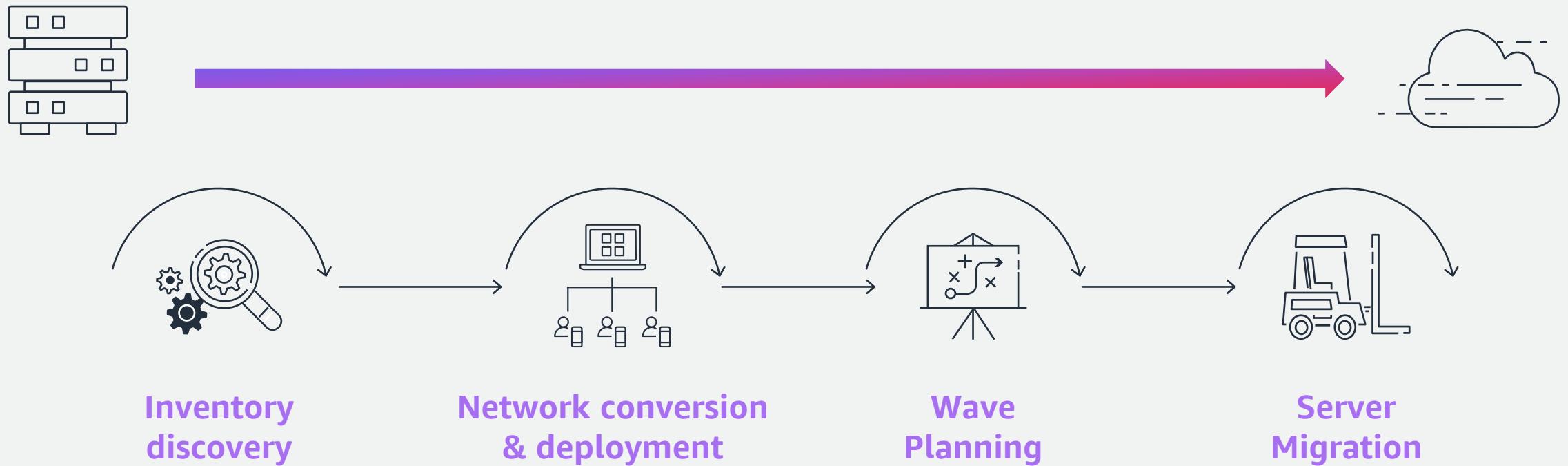
Mass modernization
of complete
applications

Refactor mainframe
code into logical
components faster
than ever before
(not yet approved)

Simplify projects
and improve team
collaboration

Q Developer: Transform for VMware

The first generative AI-powered assistant for large-scale migration and modernization of VMware workloads



Accelerate VMware migration

DEMO

Accelerate
migration of
VMware workloads

Simplify and speed
up migration and
modernization of
VMware workloads
to Amazon EC2
using generative AI
agents

Reduce costs and
license fees to
improve efficiency

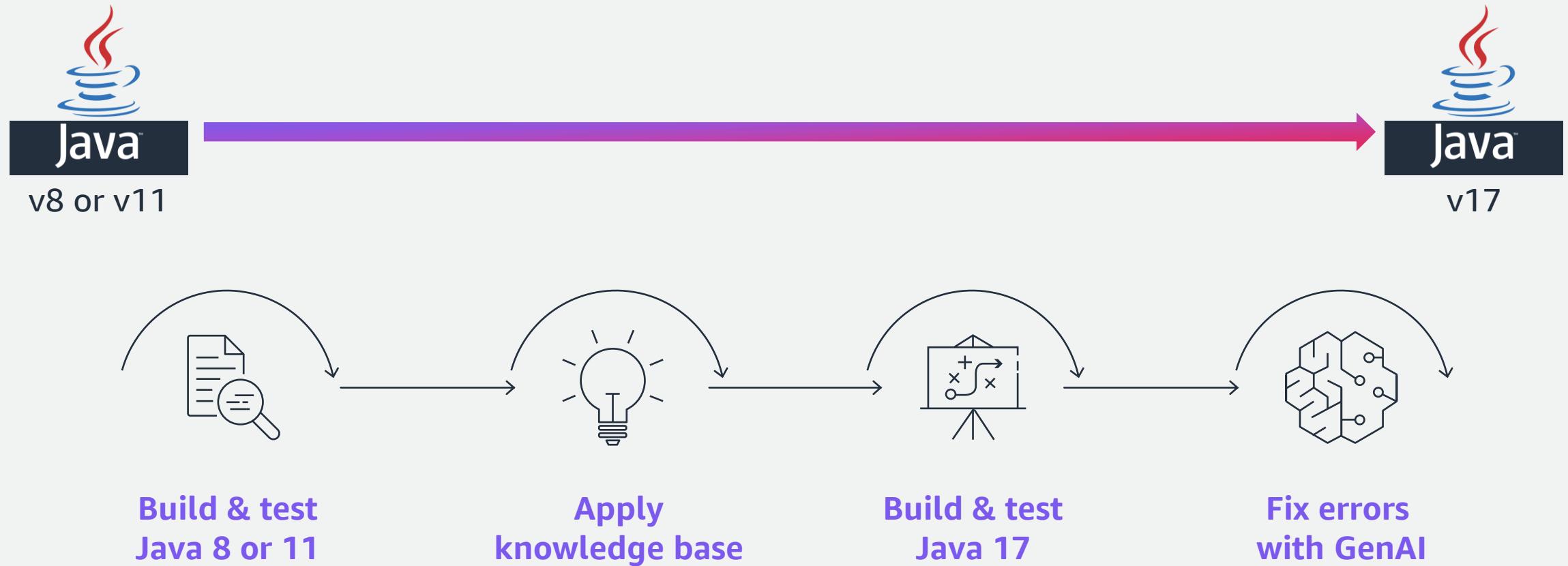
Increase innovation,
security, and
resilience

Note: placeholder demo. The demo video is not final yet.



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

Q Developer: Transform for Java



Accelerate Java upgrades

DEMO

DOP229-NEW

Automate app upgrades & maintenance using Amazon Q Code Transformation

Vishvesh Sahasrabudhe
Principal Product Manager
Amazon Web Services

Jas Chhabra
Senior Software Development Manager
Amazon Web Services



© 2024, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Complete language upgrades in a fraction of the time

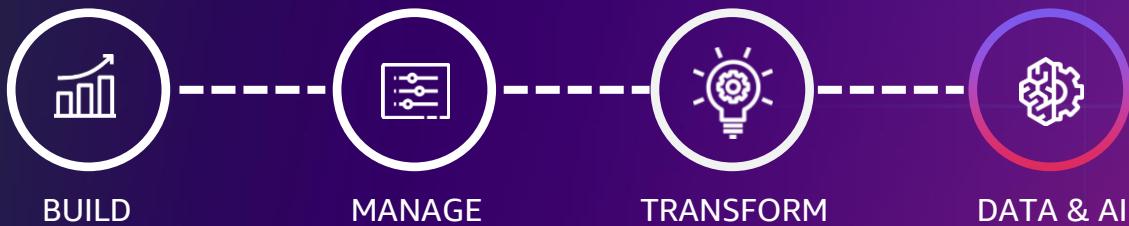
Enhance security posture and performance

Support for large-scale Java transformation projects coming soon



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

Leverage data and AI



Analyze and integrate data

The screenshot shows the Amazon Q generative SQL interface. On the left, there's a code editor with a Markdown header asking if you can filter by the year 1998. Below it is a SQL query for filtering sales data by store name 'able' and year 1998, ordered by total sales price. To the right is a results table with one row showing item ID AAAAAAAACEOCAAA. At the bottom, a status bar indicates Query ID 2033943, Elapsed time: 268 ms, and Total rows: 1. On the right side of the interface, there's a panel titled "Amazon Q generative SQL" with a generated SQL script for the same query, a "Regenerate SQL" button, and a "Add to notebook" button. Below this is a natural language input field asking if you can filter by the year 1998, followed by another copy of the generated SQL script.

```
1  ## Can you filter by the year 1998?  
1  SELECT  
2    i.i_item_id  
3  FROM  
4    tpcds.item i  
5  JOIN tpcds.store_sales ss ON i.i_item_sk = ss.ss_item_sk  
6  JOIN tpcds.store s ON ss.ss_store_sk = s.s_store_sk  
7  JOIN tpcds.date_dim d ON ss.ss_sold_date_sk = d.d_date_sk  
8 WHERE  
9   s.s_store_name = 'able'  
10  AND d.d_year = 1998  
11 GROUP BY  
12   i.i_item_id  
13 ORDER BY  
14   SUM(ss.ss_ext_sales_price) DESC  
15 LIMIT  
16  1;  
  
1  SELECT  
2    i.i_item_id  
3  FROM  
4    tpcds.item i  
5  JOIN tpcds.store_sales ss ON i.i_i  
6  JOIN tpcds.store s ON ss.ss_store_  
7  JOIN tpcds.date_dim d ON ss.ss_sold_  
8 WHERE  
9   s.s_store_name = 'able'  
10  AND d.d_year = 1998  
11 GROUP BY  
12   i.i_item_id  
13 ORDER BY  
14   SUM(ss.ss_ext_sales_price) DESC  
15 LIMIT  
16  1;
```

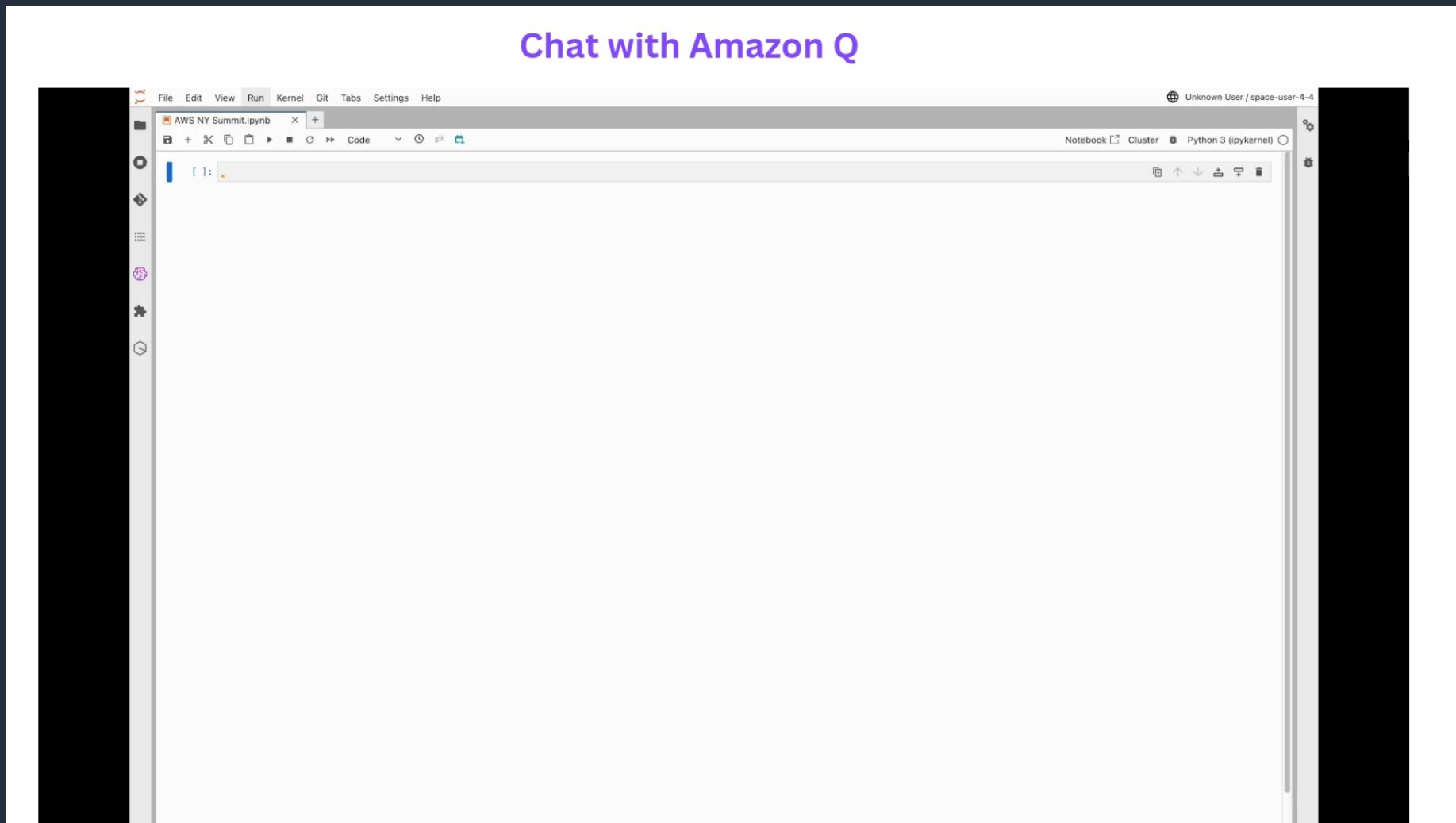
Generate a complete extract, transform, and load (ETL) script or code snippet for individual ETL operations

Express queries in natural language and receive SQL code recommendations



Build ML models

DEMO



Get code suggestions and debug errors within the IDE

Data preparation and training

No- and low-code options to democratize ML development



© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved. Amazon Confidential and Trademark.

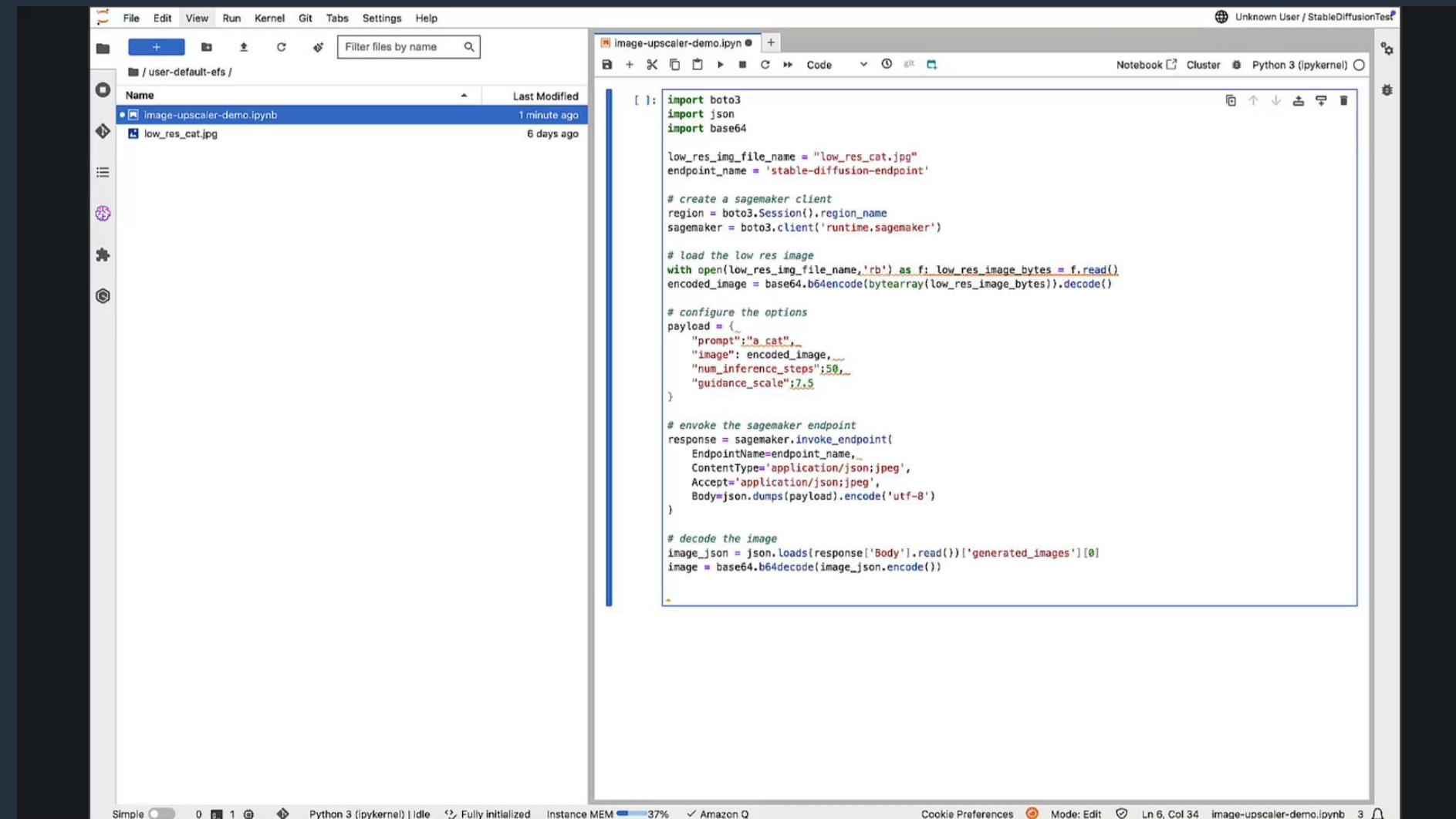
Use ML to solve business problems

DEMO



Leverage data and AI

DEMO



The screenshot shows a Jupyter Notebook interface with a sidebar on the left containing file navigation and search tools. The main area displays a Python script titled "image-upscaler-demo.ipynb". The code uses the boto3 library to interact with an AWS Sagemaker endpoint named "stable-diffusion-endpoint". It reads a low-resolution image file ("low_res_cat.jpg") and encodes it as base64. Then, it creates a payload dictionary with a prompt ("I a cat"), the encoded image, and inference steps (50). This payload is sent to the Sagemaker endpoint. Finally, the response is decoded to get the generated image. The notebook status bar at the bottom indicates "Fully initialized" and "Amazon Q" integration.

```
import boto3
import json
import base64

low_res_img_file_name = "low_res_cat.jpg"
endpoint_name = 'stable-diffusion-endpoint'

# create a sagemaker client
region = boto3.Session().region_name
sagemaker = boto3.client('runtime.sagemaker')

# load the low res image
with open(low_res_img_file_name, 'rb') as f: low_res_image_bytes = f.read()
encoded_image = base64.b64encode(bytearray(low_res_image_bytes)).decode()

# configure the options
payload = {
    "prompt": "I a cat",
    "image": encoded_image,
    "num_inference_steps": 50,
    "guidance_scale": 7.5
}

# invoke the sagemaker endpoint
response = sagemaker.invoke_endpoint(
    EndpointName=endpoint_name,
    ContentType='application/json; charset=UTF-8',
    Accept='application/json; charset=UTF-8',
    Body=json.dumps(payload).encode('utf-8')
)

# decode the image
image_json = json.loads(response['Body'].read())['generated_images'][0]
image = base64.b64decode(image_json.encode())
```

Discover and best utilize available enterprise data according to your projects

Use Amazon Q Developer across a governed and unified data and AI development environment