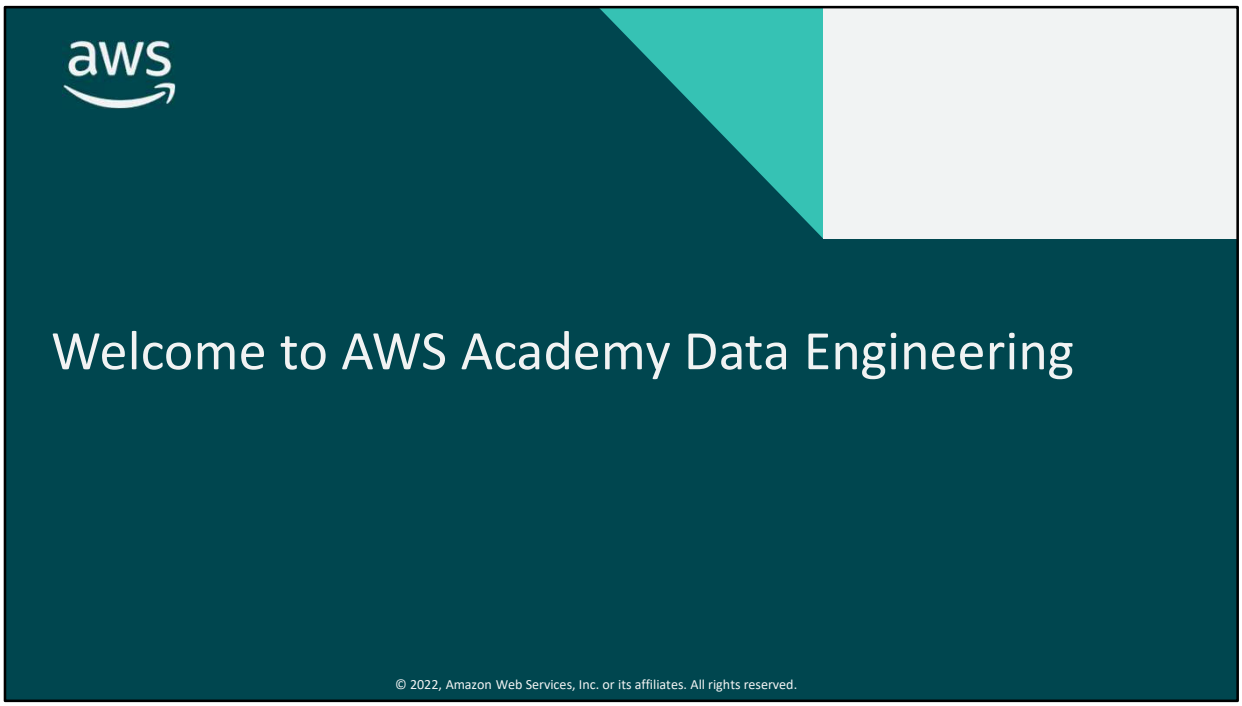


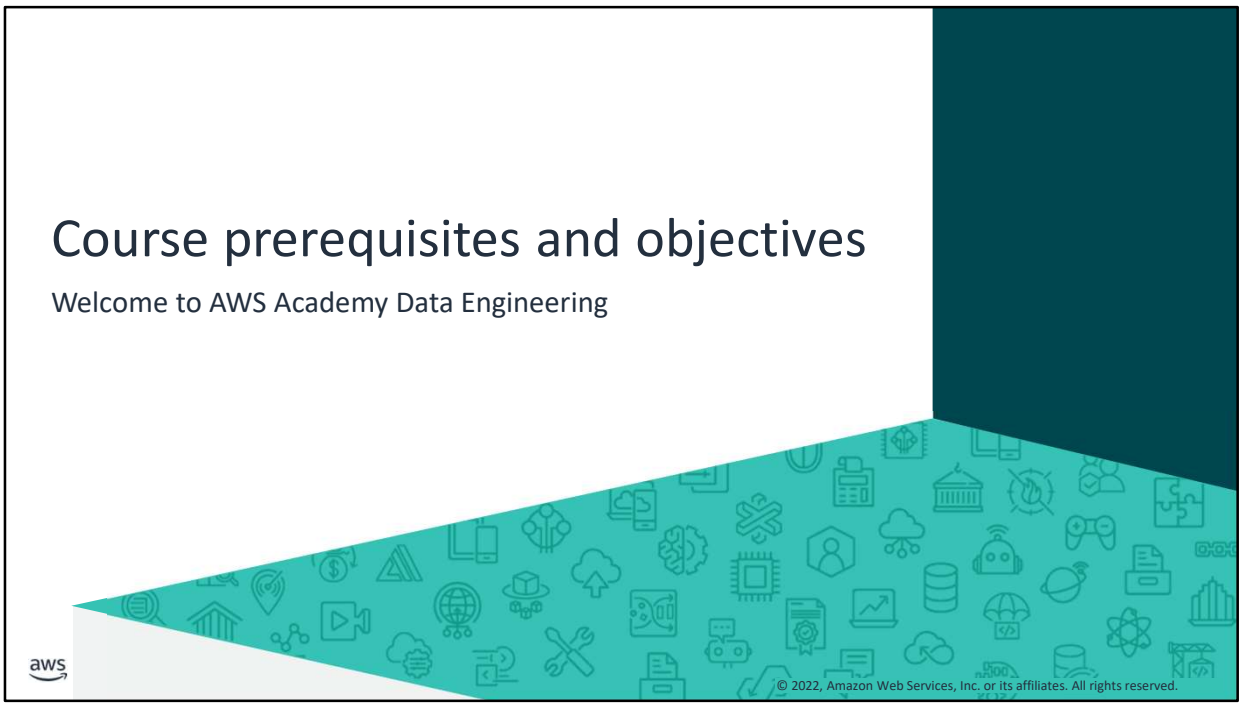
|Slide number 1
|Instructor notes
|
|Student notes
Welcome to the AWS Academy Data Engineering course.

Module overview

Sections

- Course prerequisites and objectives
- Course overview
- What's new at AWS

|Slide number 2
|Instructor notes
|
|Student notes
This module introduces you to the course.



|Slide number 3
|Instructor notes
|
|Student notes
This section provides information about the prerequisites and objectives of this course.

Prerequisites

- Completed the AWS Academy Cloud Foundations course or have equivalent experience
- Worked with Structured Query Language (SQL)
- Worked with databases
- Introduced to general networking concepts
- Understanding of decision-making knowledge in math, probability, and statistics

|Slide number 4
|Instructor notes
|
|Student notes
This slide provides an overview of the knowledge and skills that you are expected to have prior to taking this course.

Related job roles

- Data engineer
- Data analyst
- Data scientist
- Extract, transform, and load (ETL) developer
- Machine learning (ML) practitioners

|Slide number 5
|Instructor notes
|
|Student notes
This course is most aligned to a data engineer role. However, this course would also be appropriate for data analysts; data scientists; extract, transform, and load (ETL) developers; or machine learning (ML) practitioners who want to understand how the data that they use in their analysis and predictions is prepared for analysis using AWS.

Course objectives (Slide 1 of 2)

This course prepares you to do the following:

- Summarize the role and value of data science in a data-driven organization.
- Recognize how the elements of data influence decisions about the infrastructure of a data pipeline.
- Illustrate a data pipeline by using AWS services to meet a generalized use case.
- Identify the risks and approaches to secure and govern data at each step and each transition of the data pipeline.
- Identify scaling considerations and best practices for building pipelines that handle large-scale datasets.
- Design and build a data collection process while considering constraints such as scalability, cost, fault tolerance, and latency.

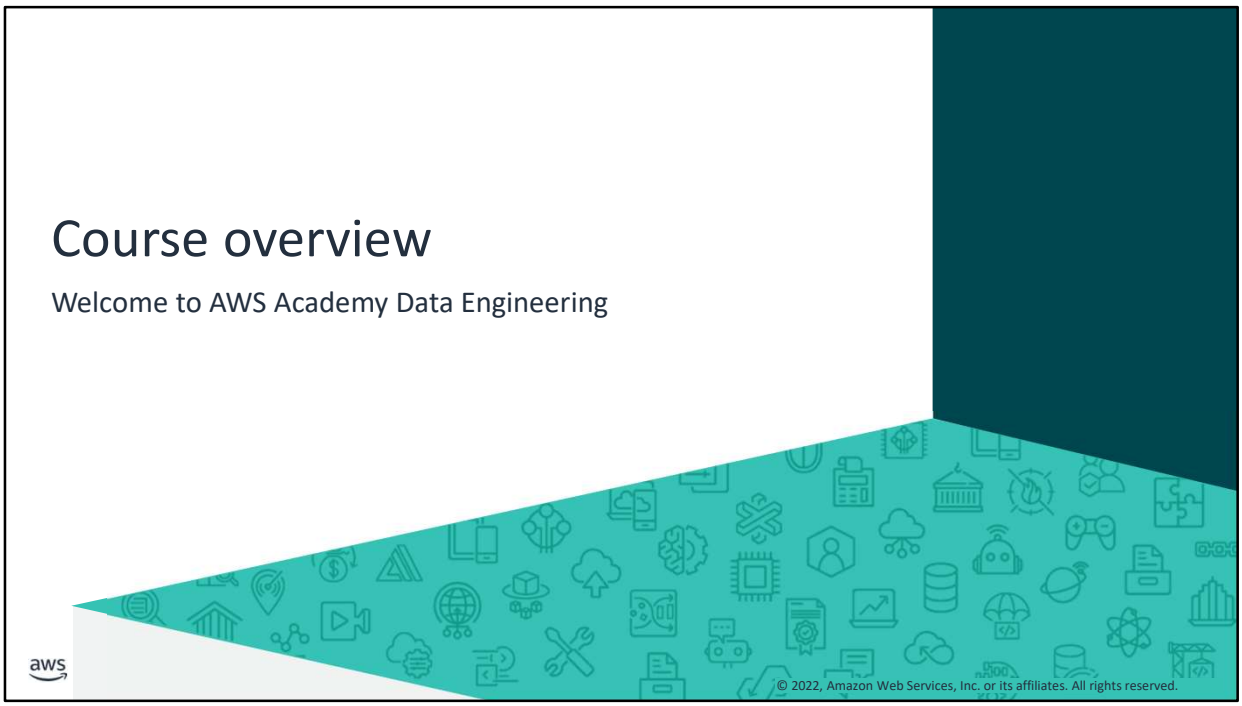
|Slide number 6
|Instructor notes
|
|Student notes

Course objectives (Slide 2 of 2)

- Select a data storage option that matches the requirements and constraints of a given data analytics use case.
- Implement the steps to process both structured, semistructured, and unstructured data formats in a data pipeline that is built with AWS.
- Explain the concept of MapReduce and how Amazon EMR is used in big data pipelines.
- Differentiate the characteristics of an ML pipeline and its specific processing steps.
- Analyze data by using AWS tools that are appropriate to a given use case.
- Implement a data visualization solution that is aligned to an audience and data type.

aws

|Slide number 7
|Instructor notes
|
|Student notes



|Slide number 8
|Instructor notes
|
|Student notes
This section provides an overview of the modules and learning modalities in this course.

Course outline

- Module 1: Welcome to AWS Academy Data Engineering
- Module 2: Data-Driven Organizations
- Module 3: The Elements of Data
- Module 4: Design Principles and Patterns for Data Pipelines
- Module 5: Securing and Scaling the Data Pipeline
- Module 6: Ingesting and Preparing Data
- Module 7: Ingesting by Batch or by Stream
- Module 8: Storing and Organizing Data
- Module 9: Processing Big Data
- Module 10: Processing Data for ML
- Module 11: Analyzing and Visualizing Data
- Module 12: Automating the Pipeline
- Module 13: Bridging to Certification
- IoT Use Case Slides

|Slide number 9
|Instructor notes
|
|Student notes
To achieve the course objectives, the course consists of the modules that are listed on the slide.

The course also includes a set of slides called the IoT Use Case Slides. Your educator might use them to discuss an example of a data pipeline to ingest and process Internet of Things (IoT) data.

Course modalities (Slide 1 of 2)

Slides

- All modules

Activities

- Module 3: Planning Your Pipeline
- Module 4: Using the Well-Architected Framework
- Module 10: Labeling with SageMaker Ground Truth

Demonstrations

- Module 10:
  - Preparing Data and Training a Model with SageMaker
  - Preparing Data and Training a Model with SageMaker Canvas
  - Creating a Linear Regression Model Using Amazon CodeWhisperer
- Module 11: Analyzing and Visualizing Data with AWS IoT Analytics and QuickSight

aws

|Slide number 10
|Instructor notes
|
|Student notes
This course is presented by using a combination of slides, activities, demonstrations, and hands-on labs. Slides are the primary delivery method and are available for every module. The remaining modalities are distributed as shown on the slide.

Course modalities (Slide 2 of 2)

Labs

- Module 2: Accessing and Analyzing Data by Using Amazon S3
- Module 4: Querying Data by Using Athena
- Module 7: Performing ETL on a Dataset by Using AWS Glue
- Module 8: Storing and Analyzing Data by Using Amazon Redshift
- Module 9:
  - Processing Logs by Using Amazon EMR
  - Updating Dynamic Data in Place
- Module 11: Analyzing and Visualizing Streaming Data with Kinesis Data Firehose, OpenSearch Service, and OpenSearch Dashboards
- Module 12: Building and Orchestrating ETL Pipelines by Using Athena and Step Functions

Capstone Project

|Slide number 11
|Instructor notes
|
|Student notes

This course is presented by using a combination of slides, activities, demonstrations, and hands-on labs. Slides are the primary delivery method and are available for every module. The remaining modalities are distributed as shown on the slide.

The Capstone Project provides an integrative project-based learning experience that reinforces technical skills that are taught in this course. The capstone offers you an opportunity to demonstrate critical thinking, problem solving, the software development lifecycle, and communication skills.

Learning assessments

Knowledge checks

- Provided for modules 2 through 12
- Drawn from the slides and student guide notes

Course assessment

- 25 randomized questions
- Drawn from the slides and student guide notes

|Slide number 12
|Instructor notes
|
|Student notes
You can assess your progress throughout this course by completing the knowledge checks and the course assessment that are included with the course materials. Each module (except the first and last modules) includes a knowledge check with questions that are specific to that module. The course assessment includes questions from all the modules to test your overall course knowledge.

Both the knowledge checks and the course assessment questions are drawn from material found in the slides and student guide notes.

Preparing for certification

- This course is not intended to fully prepare you for the AWS Certified Data Analytics – Specialty and AWS Certified Machine Learning – Specialty certifications.
- Review the Exam Guides for information regarding the certifications.
- The Bridging to Certification module provides information about additional exam preparation resources.

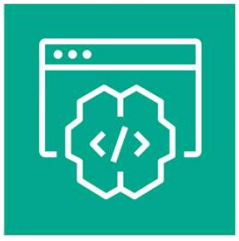
|Slide number 13
|Instructor notes
|This course will not—and is not intended to—fully prepare students for either of the related AWS Certification exams. Advise students to thoroughly review the Exam Guides for guidance in preparing for the exams.
|
|Student notes
While this course will introduce you to many aspects of data analytics and machine learning, the course will not fully prepare you for the related AWS Certification exams.

For additional information regarding the certifications, see the Exam Guides. Links are available in the Content Resources section of the course.

|Slide number 14
|Instructor notes
The "What’s new at AWS” section introduces new and emerging services and features. In order to experiment and innovate more quickly, this section provides an introduction to the latest technologies.

|Student notes
The “What’s new at AWS” section introduces new and emerging technologies at AWS.

What is Amazon CodeWhisperer?



CodeWhisperer

AI-powered code generator for IDEs and code editors

- AI coding companion:
  - Generates code suggestions based on comments and existing code
  - Offers real-time support for code authoring directly within your integrated development environment (IDE)
- AI security scanner:
  - Helps identify hard-to-find vulnerabilities
  - References multiple standards and best practices

Content processed by CodeWhisperer Professional is not stored or used for service improvement.

|Slide number 15
|Instructor notes

- | **Code suggestions**
- | Highlight the limitations in terms of IDEs and programming languages supported.
- | When talking about security scans and standards, mention Open Worldwide Application Security Project (OWASP) specifically.

|Student notes

Amazon CodeWhisperer analyzes your comments and code as you write them in your integrated development environment (IDE). It goes beyond code completion by using natural language processing to comprehend the comments in your code. By understanding English comments, CodeWhisperer generates complete functions and code blocks that align with your descriptions. CodeWhisperer also analyzes the surrounding code, ensuring the generated code matches your style and naming conventions and seamlessly integrates into the existing context.

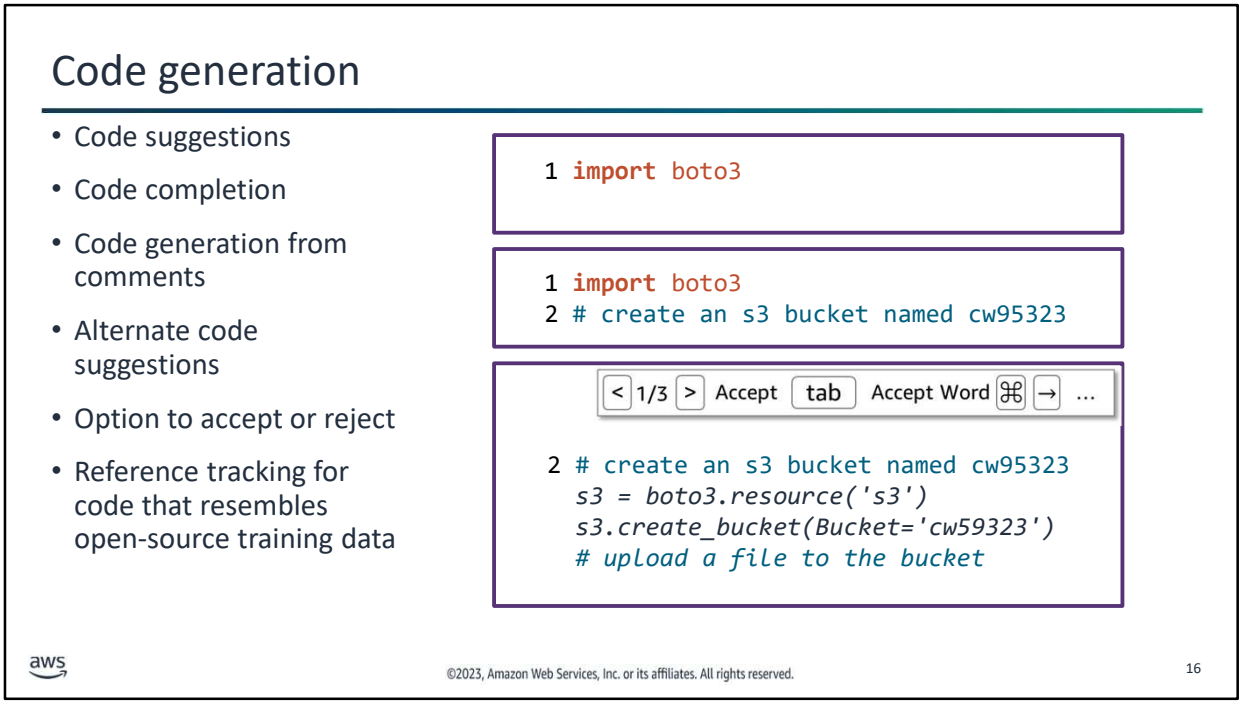
When scanning for security vulnerabilities, CodeWhisperer assesses your code against multiple sets of standards and best practices. This includes the following:

- Open Worldwide Application Security Project (OWASP) standards
- Crypto library best practices
- AWS security standards

The security scan feature is continuously updated to help keep applications free from new security vulnerabilities.

Compatibility: CodeWhisperer integrates with popular tools such as Visual Studio Code, JetBrains IDEs (IntelliJ IDEA, PyCharm, etc.), Amazon SageMaker Studio, JupyterLab, AWS Cloud9, and AWS Lambda console.
Support: CodeWhisperer supports a wide range of programming languages and development environments, including Python, Java, JavaScript, TypeScript, C#, Go, Rust, PHP, Ruby, Kotlin, C, C++, shell scripting, structured query language (SQL), and Scala.
Installation: You can access CodeWhisperer by downloading and installing the AWS Toolkit IDE extension or plugin. You can also activate CodeWhisperer from directly within the AWS Lambda and AWS Cloud9 console code editors.

Installation instructions vary depending on the environment. For more information, see the content resources page of your online course.



|Slide number 16

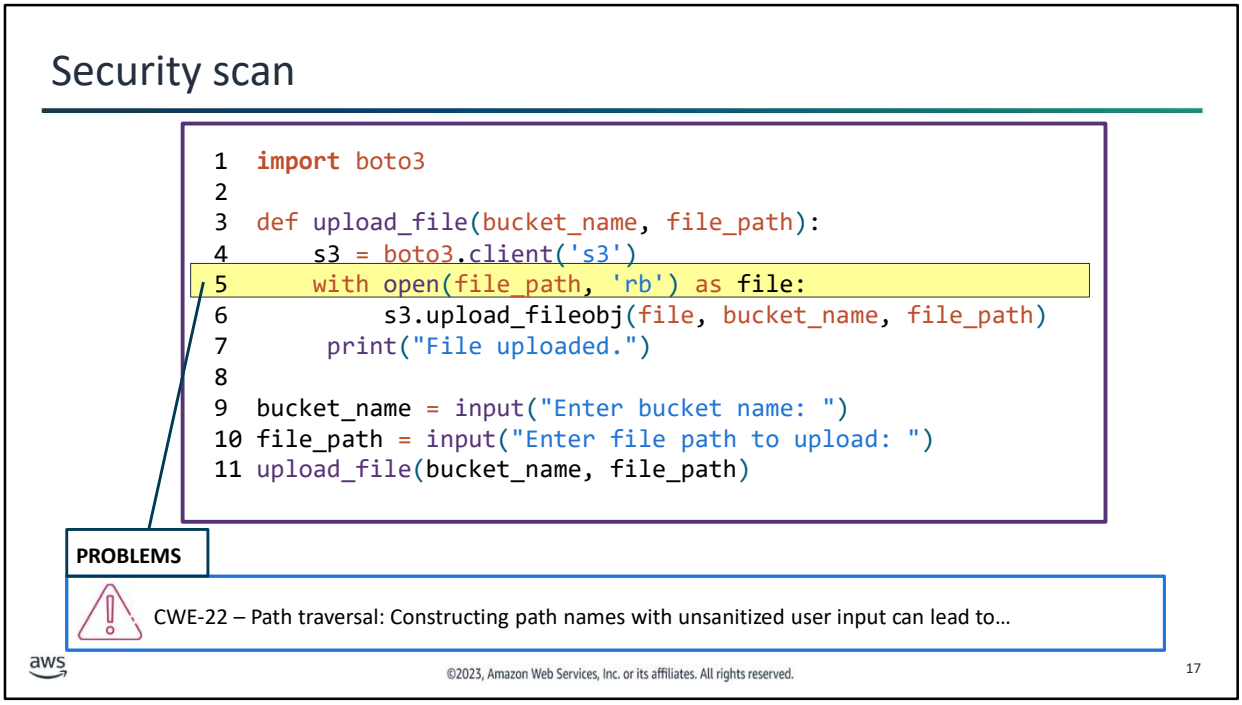
| Instructor notes

- | **Code Generation**
- | Code example shows a progression over time in each of the three boxes.
- | Large Language Model (LLM) based code suggestions
- | Can generate code suggestions based on user comments and current code context
- | Can generate complete functions and code blocks (more than just 'autocomplete')
- | Learns based on your prior code
- | Available in many popular IDEs and works with many popular programming languages
- | **Open Code Reference Tracking**
- | CodeWhisperer is trained on billions of lines of Amazon and open-source code to enhance its capabilities.
- | The code generated by CodeWhisperer is independent of its training data, ensuring originality.
- | Although rare, there might be instances where the generated code resembles open-source code from the training data, which CodeWhisperer detects.
- | If you accept the suggested code in such cases, CodeWhisperer adds a reference to the Open Code Reference Log.
- | The referenced code is marked with repository and licensing information, so you can provide appropriate attribution as necessary.
- | It's important to note that your company might have specific policies regarding the use of open-source code.
- | Enterprise controls can deactivate or filter code suggestions that resemble open-source training data, providing flexibility in aligning with your organization's guidelines.

|**Student notes**
The code generation feature of CodeWhisperer offers code suggestions in real time in your development environment. It automatically offers code completion and code generation suggestions. It uses natural language processing of English comments in your code and an understanding of surrounding code to suggest whole lines of code, complete functions, and logical blocks of code. The generated code is aligned with your coding style and naming conventions. CodeWhisperer prioritizes secure coding and responsible artificial intelligence (responsible AI) practices. It's optimized for Amazon APIs and trained extensively on Amazon and open-source code. You have the option to accept the first suggestion, explore more suggestions, or continue writing your own code. It's important to review each code suggestion before accepting it because you might need to make edits to ensure that the suggestion aligns with your intended functionality.

User actions
- Previous and next suggestion: Use the left arrow and right arrow.
- Accept a suggestion: Press Tab.
- Reject a suggestion: Press Esc.
- Manually start code generation when typing a comment: On MacOS, press Option+C, and on Windows, press Alt+C.

Open Code Reference Log
CodeWhisperer learns from open-source projects and the code it suggests might occasionally resemble code samples from the training data. With the reference log, you can view references to code suggestions that are similar to the training data. When such occurrences happen, CodeWhisperer notifies you and provides repository and licensing information. Use this information to make decisions about whether to use the code in your project and properly attribute the source code as desired.



|Slide number 17

| Instructor notes

- | The Amazon CodeWhisperer security scan is integrated with Amazon CodeGuru.
- | Scan generated and developer-written code to detect security vulnerabilities
- | Receive vulnerability remediation suggestions.
- | Scan for hard-to-find security vulnerabilities.
- | It supports VS Code and JetBrains IDEs for Python, Java, and JavaScript.
- | Scanning for security vulnerabilities includes Open Worldwide Application Security Project (OWASP) standards, enforcement of crypto library and AWS security standards and best practices, and more. The detector library in CodeGuru is continuously updated to help keep applications free from new security vulnerabilities.

| **Student notes**

The security scanning feature of CodeWhisperer detects security vulnerabilities in both CodeWhisperer-generated code and developer-written code. It scans the code to identify potential vulnerabilities and provides suggestions for remediation. This includes scanning for hard-to-find vulnerabilities that might be overlooked. The security scan is compatible with popular IDEs such as VS Code and JetBrains. It supports Python, Java, and JavaScript.

Benefits of Amazon CodeWhisperer

Value to developers	Value to organizations
• Increase velocity.	• Use at all experience levels.
• Spend less time writing code.	• Support open-source attribution.
• Receive help directly within your IDE.	• Reduce the risk of security vulnerabilities.
• Find security vulnerabilities in your code.	• Increase code quality and developer productivity.

|Slide number 18

| **Instructor notes**

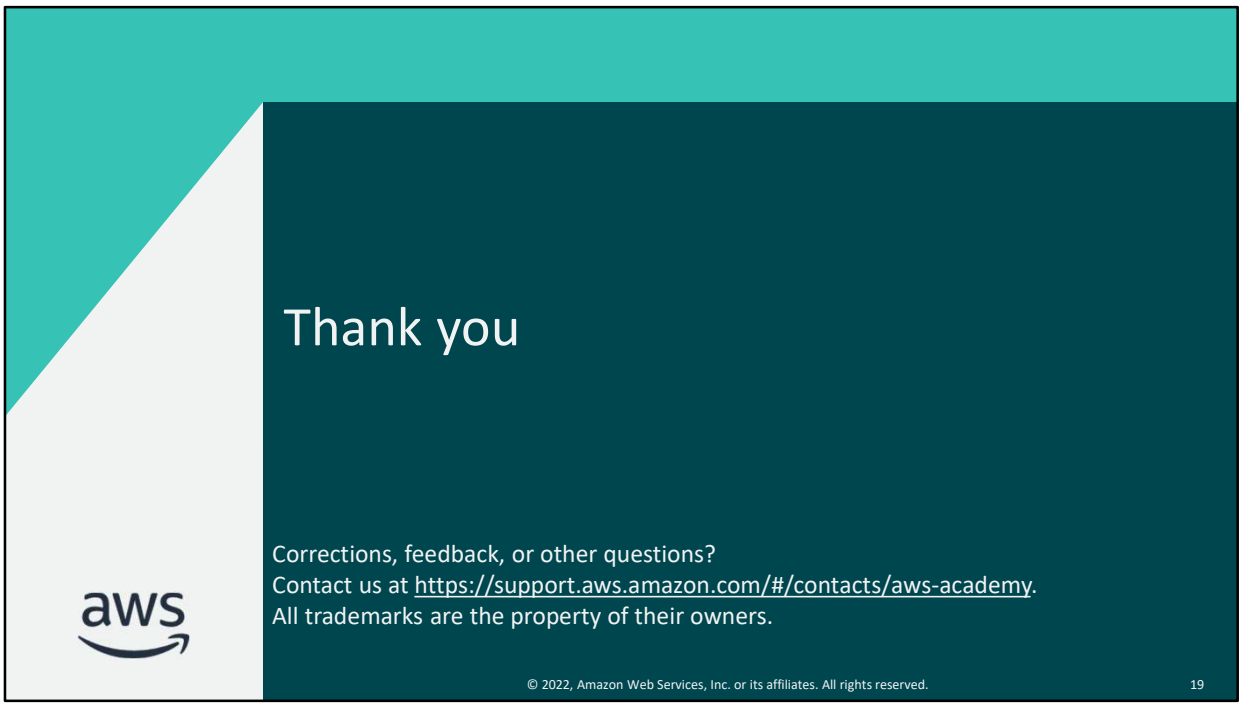
- | **Value to Coders:**
- | Automated code generation for straightforward and repetitive tasks, such as unit tests, string manipulation, and list processing
- | Reduced time spent on exploring and learning new programming languages, SDKs, APIs, and frameworks
- | Access to high-quality code suggestions tailored to their coding style and patterns
- | Improved productivity and focus on important business problems
- | Avoidance of unintentional code violations and potential legal issues related to open-source software
- | **Value to Organizations:**
- | Faster development of applications due to automated code generation and reduced manual coding time
- | Mitigation of the developer shortage by optimizing the use of available developers' time
- | Minimization of security vulnerabilities through code suggestions that follow best practices and identification of potential risks
- | Protection of intellectual property by avoiding unintentional code violations and copyright or license infringements
- | Enhanced code quality and reliability by reducing the reliance on code reuse from external sources
- | Improved efficiency in addressing continuously evolving software threats and maintaining a secure codebase

|**Student notes**
Automated code generation automates repetitive tasks and saves you time. It eliminates the need for you to invest excessive hours in exploring and learning new technologies. Instead, you can rely on high-quality code suggestions that match your coding style. This approach enhances your productivity so you can focus on critical tasks, which encourages innovation and progress in software development. With automated code generation, you can streamline your workflows and achieve significant time savings while ensuring the delivery of code that meets your standards.

CodeWhisperer code generation offers many benefits for software development organizations. It accelerates application development for faster delivery of software solutions. By automating repetitive tasks, it optimizes the use of developer time, so developers can focus on more critical aspects of the project. Additionally, code generation helps mitigate security vulnerabilities, safeguarding the integrity of the codebase.

CodeWhisperer also protects open source intellectual property by providing the open source reference tracker. CodeWhisperer enhances code quality and reliability, leading to robust and efficient applications. And it supports an efficient response to evolving software threats, keeping the codebase up to date with the latest security practices. CodeWhisperer has the potential to increase development speed, security, and the quality of software.

For more information on the benefits of CodeWhisperer, see the content resources page of your online course.



|Slide number 19
|Instructor notes

Student notes
Thank you for completing this module.