About .NET on AWS:
AWS 도구 및 서비스 소개

Dongcheol Ryu
AWS | Partner Solutions Architect

© 2024, Amazon Web Services, Inc. or its Affiliates.

아젠다

- Why Windows on AWS
- . NET Tools Overview
  - IDE Integration
  - Programmable SDKs
  - Command-line tools
  - CI/CD integration
- Migrating .NET applications to AWS
  - Application Modernization Tools
- How to monitor .NET Applications

© 2024, Amazon Web Services, Inc. or its Affiliates.

Why Windows on AWS

© 2024, Amazon Web Services, Inc. or its Affiliates.

운영 안정성 경험

"경험에 대한 압축 알고리즘은 없습니다."
- 앤디 재시

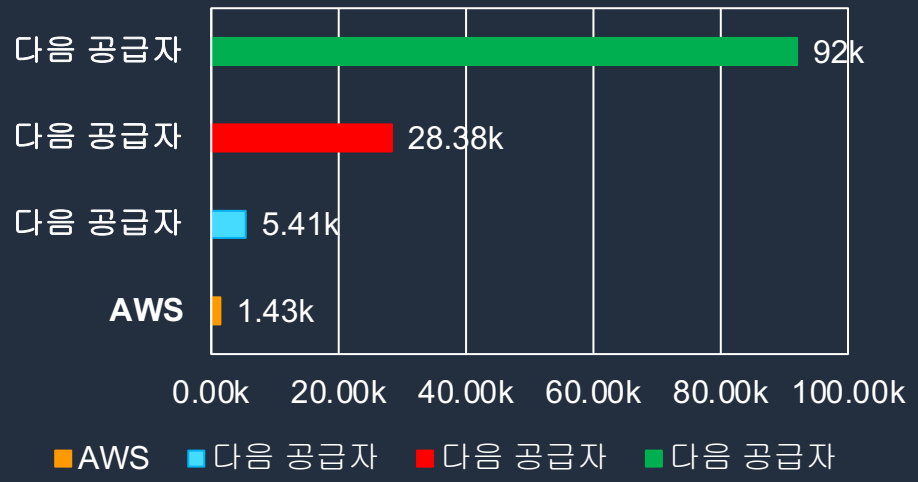
© 2024, Amazon Web Services, Inc. or its Affiliates.

고가용성으로 확장하도록 설계된 AWS 클라우드 인프라

복원력이 높은 기본 인프라에 대한 AWS의 투자는 고객 비용 절감으로 이어집니다.

2023년 공급업체별 글로벌 서비스 중단 시간(퍼블릭)

서비스 중단 시간 = 게시된 다운타임 시간 * 영향을 받은 지역 * 영향을 받은 서비스



높은 다운타임 및 보안 사고는 고객 비용에 더 큰 영향을 미칩니다.

$1.25 to $2.5B	$500K to $1M	$3.6M	$141	$474K
Annual Fortune 1000 application downtime costs (IDC)	Cost/hr of a critical application failure (IDC)	Average cost of a data breach (Ponemon Institute)	Average cost per lost or stolen record (Ponemon Institute)	Average cost/hr of downtime (Ponemon Institute)

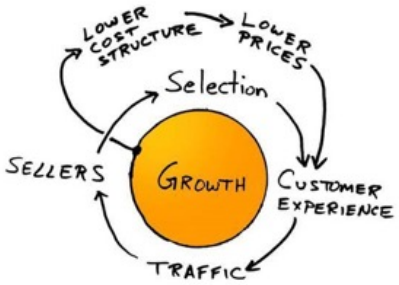
Source: IDC: Annual Fortune 1000 application downtime costs Study
Ponemon Institute: Cost of Data Center Outages, ; updates and stats

* Sources: Public Status Pages for Cloud Providers

© 2024, Amazon Web Services, Inc. or its Affiliates.

클라우드 서비스 가격 책정에 대한 AWS의 철학

규모 확장을 통해 얻은 경제적 효율성을 고객에게 돌려주는 문화...



aws went through Price Reductions since 2006

가격 인하 관련 AWS 블로그

고객에게 비용 효율성을 가져다주는 제품을 개발하기 위한 끊임없는 노력...

Since the launch of S3 Intelligent Tiering, customers saved over $2 Billion

Since 2018, AWS has released 3 generations of custom ARM chips, named Graviton.

The latest Graviton 3 CPUs offer up to 40% better price/performance ratios over conventional CPUs, for most workloads.

Examples[1] of cost savings enabled by Graviton (over x86 CPUs):

1. **ML Inference** (SageMaker) – 70% lower price
2. **ML Training** (SageMaker) – 50% lower price
3. **Databases** (Aurora, RDS) – 50% lower price
4. **General purpose compute** (EC2) – 50% lower price
5. **Containers** (Kubernetes, Docker) – 40% lower price
6. **Serverless compute** (Lambda) – 34% lower price
7. **Big Data** (EMR) – 30% lower price
8. **NoSQL** (Document, Graph) – 30% lower price

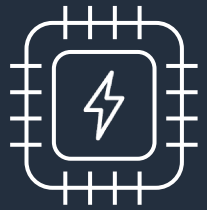
© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS compute is powered by unique, extra hardware

Nitro Hypervisor

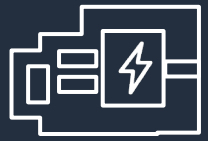
- 경량 하이퍼바이저
- 메모리 및 CPU 할당
- 베어 메탈과 같은 성능

Nitro Security Chip



- 마더보드에 통합
- 하드웨어 리소스 보호

Nitro Card



- Amazon Elastic 블록 스토어
- Elastic 네트워크 어댑터
- 모니터링 및 보안

Graviton Processors

- 탁월한 성능
- 비용 절감
- 광범위한 ARM 에코시스템 (OS 및 서비스)

전용 목적에 맞게 설계된 모듈식 빌딩 블록

© 2024, Amazon Web Services, Inc. or its Affiliates.

Why do customers choose AWS for Windows?

BREADTH AND DEPTH

신뢰성 및 보안

BEST PERFORMANCE

LOWER TCO 및 유연한 라이선싱

혁신과 경험

2x
higher performance for a SQL Server workload than the next largest cloud provider*

2x
more regions with multiple availability zones than the next largest cloud provider

5x
more services offering encryption than the next largest cloud provider

36%
savings using AWS over three years, by right-sizing instances with Migration Evaluator

© 2024, Amazon Web Services, Inc. or its Affiliates.

.NET Tools in AWS
Overview

© 2024, Amazon Web Services, Inc. or its Affiliates.

.NET 개발자에게 필요한 것은 무엇인가요?

풍부한 IDE 및 편집기 지원

Intellisense

“Don’t make us leave the IDE”

“Getting started” project templates

모든 기능을 쉽게 사용할 수 있는 소모성 SDK

간편한 디버깅 및 테스트

명령줄 지원

PowerShell and Windows cmd.exe

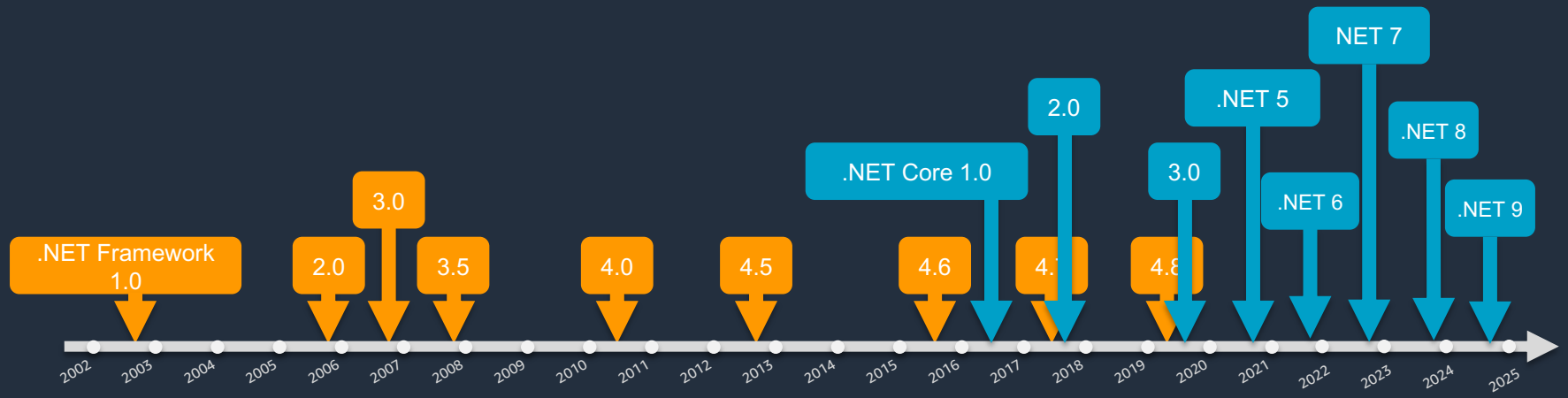
‘dotnet CLI’ integration for .NET Core

CI/CD 통합

.NET Core를 통한 크로스 플랫폼 개발 지원

© 2024, Amazon Web Services, Inc. or its Affiliates.

.NET 여정



레거시(.NET 프레임워크)

최신(.NET Core 이상)

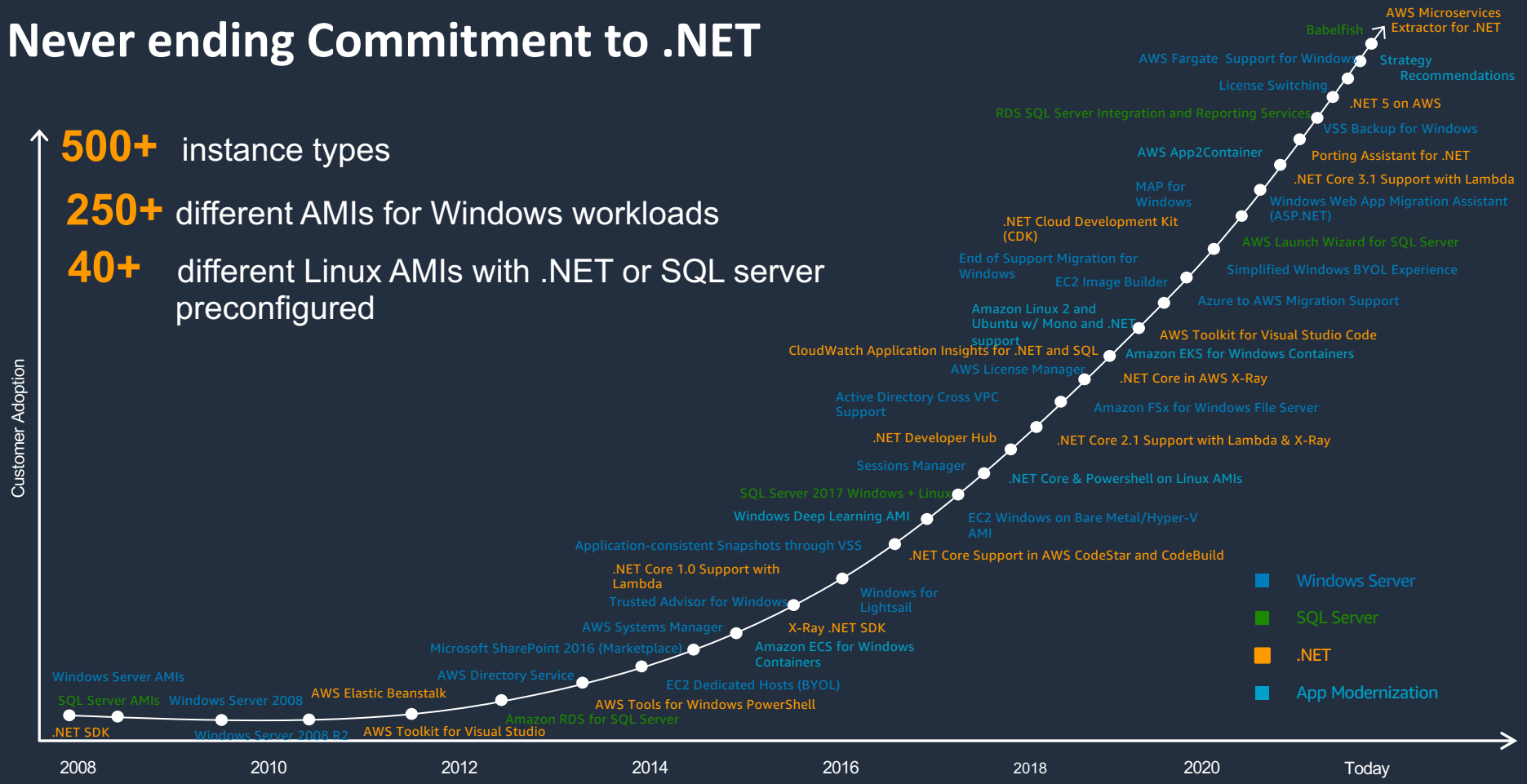
© 2024, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Never ending Commitment to .NET

500+ instance types

250+ different AMIs for Windows workloads

40+ different Linux AMIs with .NET or SQL server preconfigured



© 2024, Amazon Web Services, Inc. or its Affiliates.

Development Tools

IDE Integration

- AWS Toolkit for Visual Studio
- AWS Toolkit for Visual Studio Code
- AWS Toolkit for Rider

Programmable SDK

- AWS SDK for .NET
- AWS CDK for .NET

Command Line Tools

- AWS CLI
- 'dotnet' CLI extensions
- AWS Tools for PowerShell
- AWS SAM

CI/CD Integration

- AWS Tools for Azure DevOps
- Amazon CodeCatalyst
- AWS CodePipeline/ CodeBuild

Migration / Modernization Tools

- Porting Assistant for .NET and AWS Toolkit for .NET Refactoring
- App2Container
- Microservice Extractor for .NET

© 2024, Amazon Web Services, Inc. or its Affiliates.

.NET Tools in AWS
IDE Integration

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Toolkit for Visual Studio

AWS Toolkit for Visual Studio

AWS 탐색기 - AWS Resource Tree view

- 개발자가 관심 있는 AWS 서비스 및 리소스에 대한 트리뷰를 제공합니다.
- S3 버킷 관리
- Amazon DynamoDB 테이블 보기, 생성, 편집 및 삭제
- EC2 컴퓨팅 인스턴스 시작 및 관리

... 그리고 훨씬 더!

Amazon Q통합

- Use AI to generate code recommendations based on prior code and comments in your ID
- Reference Tracking
- Quickly accomplish coding tasks

손쉬운 배포를 위한 마법사

- AWS Elastic Beanstalk
- Amazon Elastic Container Service
- AWS Lambda
- AWS CloudFormation

"Getting started" project templates ("blueprints") for serverless functions and applications

Code-editing Intellisense for AWS CloudFormation json templates

https://aws.amazon.com/visualstudio

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Toolkit for Visual Studio

AWS Toolkit for Visual Studio

File Edit View Git Project Debug Test Analyze Tools Extensions Window Help Search HelloWorld

Select Startup Item...

AWS Explorer

Credentials: Profile:d

Region: Asia Pacific (Seoul)

- Amazon CloudFront
- Amazon CloudWatch
- Amazon DynamoDB
- Amazon EC2
  - AMIs
  - Elastic IPs
  - Instances
  - Key Pairs
  - Security Groups
  - Volumes
- Amazon Elastic Container Service
- Amazon RDS
- Amazon S3
- Amazon SNS
- Amazon SQS
- Amazon VPC
- AWS CloudFormation
- AWS Elastic Beanstalk
- AWS Identity and Access Manageme
  - Groups
  - Roles
  - Users
- AWS Lambda

AWS Getting Started

Getting Started: AWS Toolkit with Amazon Q

Documentation | GitHub

Features

You can return to this page at any time (Extensions > AWS Toolkit > Getting Started)

Amazon Q

Connected with IAM Identity Center

Sign out

Disable Amazon Q

AWS Toolkit

Connected with IAM Credentials

Sign out

Disable AWS Toolkit

Documentation and Tutorials

- Amazon Q Developer User Guide
- Try Amazon Q with examples
- Using the Toolkit for Visual Studio
- Deploying C# Functions to AWS Lambda
- Deploying .NET Applications on AWS Elastic Beanstalk
- AWS .NET Developer Blog

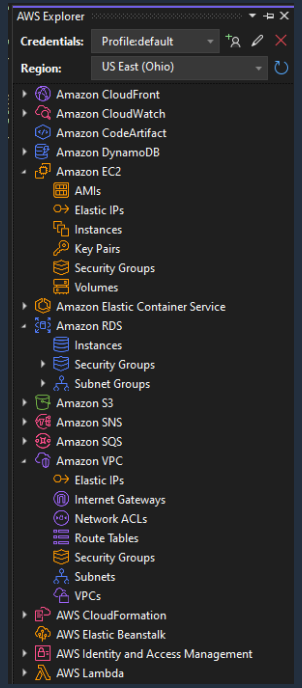
☑ Allow AWS Toolkit with Amazon Q to collect usage information

By letting us know which Toolkit features are used, this data helps improve the AWS Toolkit and other Amazon product data in accordance with the AWS Privacy Policy. When available, an AWS Account ID is associated with this informatior

© 2024, Amazon Web Services, Inc. or its Affiliates.

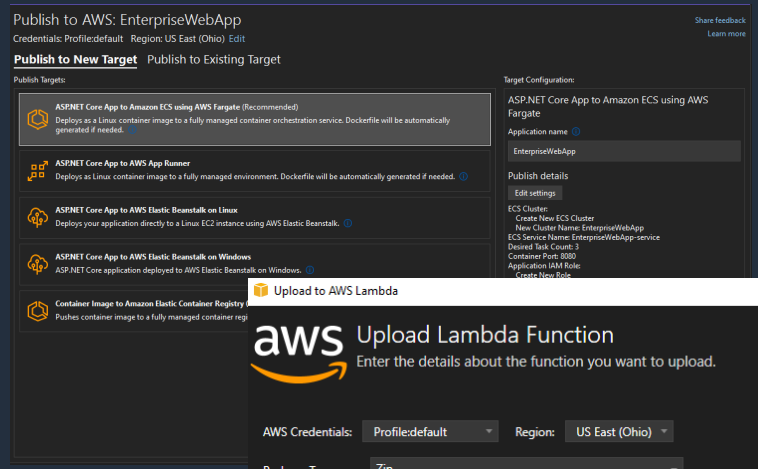
AWS Toolkit for Visual Studio

AWS Toolkit for Visual Studio



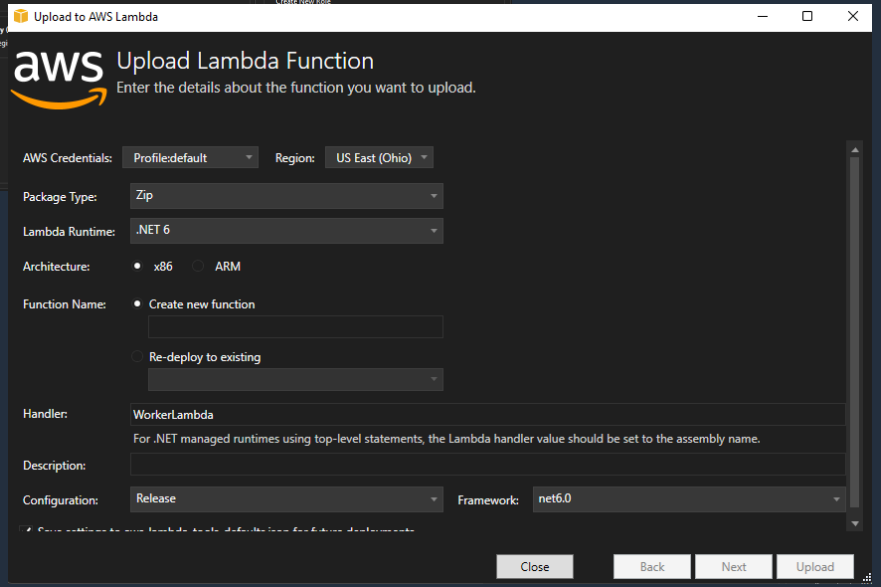
AWS 서비스 및 리소스 살펴보기

사용하기 쉬운 마법사...



웹 애플리케이션으로:

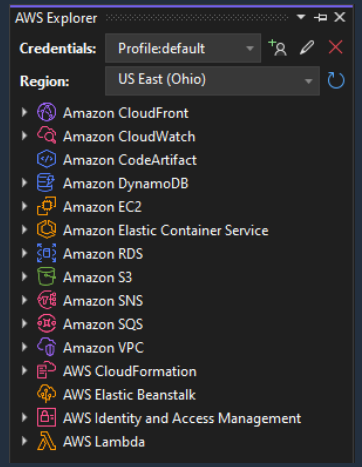
Elastic beanstalk
Amazon ECS with Fargate
AWS App Runner
Amazon ECR



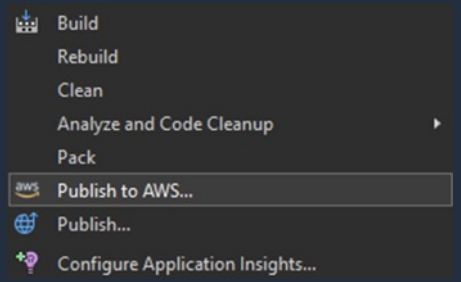
Serverless functions and applications to AWS Lambda

© 2024, Amazon Web Services, Inc. or its Affiliates.

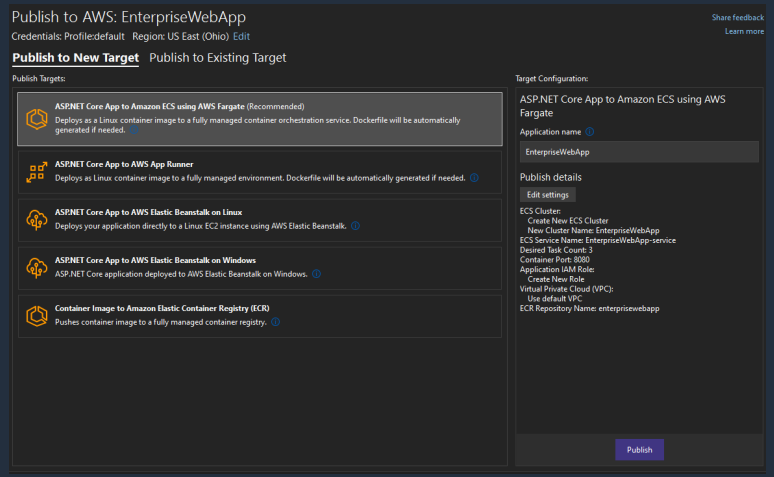
.NET Deployment Experience



새로워진 AWS 탐색기



간소화된 퍼블리싱 경험



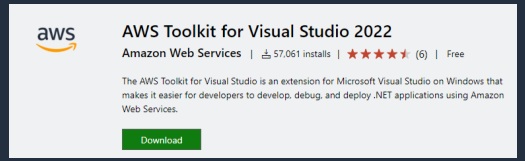
지능형 추천

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Toolkit for Visual Studio

AWS Toolkit for Visual Studio

- Visual Studio 마켓플레이스에서 설치할 수 있습니다:



- 또는 https://aws.amazon.com/visualstudio/ 에서 다운로드

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Toolkit for Visual Studio Code

AWS Toolkit for Visual Studio Code

- **AWS Toolkit for Visual Studio Code** 은 비주얼 스튜디오 코드 마켓플레이스에서 **사용 할 수 있음**
- Node.js, Python, **.NET** 및 **PowerShell에서** 서버리스 애플리케이션 개발을 목표로 하는 통합 경험을 제공

"시작하기" 프로젝트 템플릿

단계별 디버깅

Deployment from the IDE

Amazon Q (CodeWhispere) Integration

https://aws.amazon.com/visualstudiocode/
https://github.com/aws/aws-toolkit-vscode
https://docs.aws.amazon.com/toolkit-for-vscode

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Toolkit for JetBrains Rider

AWS Toolkit for Rider

Get quickly started using serverless application templates

서버리스 애플리케이션을 로컬에서 디버그

Deploy from the IDE

IDE에서 CloudWatch 액세스

Amazon Q (CodeWhisperer) Integration

https://aws.amazon.com/rider/

© 2024, Amazon Web Services, Inc. or its Affiliates.

Amazon Q Developer .NET transformation

AI 지원

.NET Framework에서 플랫폼 간 .NET 업그레이드를 수행하여 Windows에서 Linux로 애플리케이션을 더 빠르게 마이그레이션

- 현재 Java 지원
- NET 지원 출시 예정(Preview)
- 마이그레이션 시간 단축
- 보안 취약성 감소
- Linux로의 마이그레이션을 가속화하여 비용 절감

© 2024, Amazon Web Services, Inc. or its Affiliates.

Amazon Q Developer .NET transformation

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help

Debug | Any CPU | Bookstore.Cdk

Data Sources

Solution Explorer

Search Solution Explorer (Ctrl+;)

Solution 'BobsBookstoreClassic' (5 of 5 projects)

- Solution Files
- Bookstore.Cdk
- Bookstore.Common
- Bookstore.Data
  - Properties
  - References
  - FileServices
  - ImageResizeService
  - ImageValidationServices
  - Repositories
  - App.config
  - ApplicationDbContext.cs
  - BookstoreConfiguration.cs
  - BookstoreDbInitializer.cs

Solution Explorer | Git Changes

- Build Solution Ctrl+Shift+B
- Rebuild Solution
- Clean Solution
- Analyze and Code Cleanup
- Batch Build...
- Configuration Manager...
- Manage NuGet Packages for Solution...
- Restore NuGet Packages
- Port solution with Amazon Q Developer
- Collapse All Descendants Ctrl+Left Arrow
- New Solution Explorer View
- File Nesting
- Project Dependencies...
- Project Build Order...
- Add

Port a solution with Q Developer

The entire solution has been selected for porting.

Choose a .NET target

net8.0

☑ Port .NET Standard projects to .NET

Start | Cancel

Code Transformation Plan

Amazon Q Developer Code Transformation Hub

Transformation job is in progress. Elapsed time 00:00:13 View code transformation plan

View Transformation progress

- Building locally to validate project(s)
- Awaiting transformation job startup
- Generating code transformation plan
- Transforming code

© 2024, Amazon Web Services, Inc. or its Affiliates.

.NET Tools in AWS
Programmable SDKs

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS SDK for .NET and extension libraries

AWS SDK
for .NET

Central home for all SDK and extension libraries

Programmatic access to all AWS services

- 거의 매일 업데이트 (in sync with service updates and launches)
- 간단한 프로그래밍 모델

다양한 확장 라이브러리가 '상위 수준' 기능을 지원합니다.

.NET 크로스 플랫폼 지원

- .NET 프레임워크, 최신 .NET(.NET Core → .NET 8.0)을 지원합니다, 및 Xamarin 개발
- 오픈 소스

Distributed via NuGet

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS SDK for .NET - Consistent SDK coding pattern

AWS SDK for .NET

```
using servicename;
using servicename.Model;

var client = new Amazonservicenameclient();
operationnameResponse response = await client.operationnameAsync(new
operationnameRequest
{
    RequestProperty1 = "some data",
    RequestProperty2 = [ "someother", "data" ]
});

foreach (var element in response.OutputDataMember)
{
   // do something
}
```

API Client

Operation

Request Object

Process Result

© 2024, Amazon Web Services, Inc. or its Affiliates.

NET용 AWS SDK - 실제 사례

AWS SDK for .NET

```
using Amazon.S3;
using Amazon.S3.Model;

namespace ConsoleApp;

class Program
  {
    static async Task ListS3BucketObjects()
    {
      using var s3Client = new AmazonS3CIient(Amazon.RegionEndpoint.USWest2);
      string nextpagemarker = null;

      do
      {
        var listObjectsResponse await s3Client.ListObjectsAsync(new ListObjectsRequest
            {
                BucketName "my-bucket-name",
                Marker = nextPageMarker
            });

        nextpagemarker = listObjectsResponse.NextMarker;


        foreach (var s3Object in listObjectsResponse.S3Objects)
        {
            Console.WriteLine($"{s3Object.Key}, last modified {s3Object.LastModified.ToString("g")}");
        }

      } while ( ! string.IsNuUOrEmpty(nextPageMarker));
    }
}
```

예시:
S3 버킷에 있는 모든 개체 나열

© 2024, Amazon Web Services, Inc. or its Affiliates.

NET용 AWS SDK - 실제 사례

AWS SDK for .NET

```
using Amazon.S3;
using Amazon.S3.Model;

namespace ConsoleApp;

class Program
  {
    static async Task ListS3BucketObjects()
    {
      using var s3Client = new AmazonS3CIient(Amazon.RegionEndpoint.USWest2);
      string nextpagemarker = null;

      do
      {
        var listObjectsResponse await s3Client.ListObjectsAsync(new ListObjectsRequest
            {
                BucketName "my-bucket-name",
                Marker = nextPageMarker
            });

        nextpagemarker = listObjectsResponse.NextMarker;


        foreach (var s3Object in listObjectsResponse.S3Objects)
        {
            Console.WriteLine($"{s3Object.Key}, last modified {s3Object.LastModified.ToString("g")}");
        }

      } while ( ! string.IsNuUOrEmpty(nextPageMarker));
    }
}
```

Client

Operation

Request Object

Process Result

© 2024, Amazon Web Services, Inc. or its Affiliates.

Amazon S3를 사용한 프로그래밍 - 기본 작업

AWS SDK for .NET

버킷 만들기

개체 가져오기

버킷 삭제

```
using (var s3CIient = new AmazonS3Client());

// Create a bucket
var bucketResponse = s3Client.PutPutBucket(new PutBucketRequest
{
    BucketName = "<bucketname>"
});

// Add Object to bucket
var putObjectResponse = s3CIient.PutObject(new PutObjectRequest
{
    Key = "<objectkey>",
    ContentBOdy = "<Object-Content>",
    BucketName = "<bucketname>"
});

// Get Object
var getObjectResponse = s3CIient.GetObject(new GetObjectRequest
{
    Key = "<objectkey>",
    BucketName = "<bucketname>"
});

// Delete Object
var deleteObjectResponse = s3CIient.DeleteObject(new DeleteObjectRequest
{
    Key = "<objectkey>",
    BucketName = "<bucketname>"
});

// Delete Bucket
var deleteBucketResponse = s3CIient.DeleteBucket(new DeleteBucketRequest
{
    BucketName = "<bucketname>"
});
```

개체 만들기

개체 삭제

© 2024, Amazon Web Services, Inc. or its Affiliates.

Using ASP.NET Core Dependency Injection

AWS SDK for .NET

NuGet ***AWSSDK.Extensions.NETCore.Setup*** adds extension methods to **IServiceCollection** that can be used to add AWS services to the .NET Dependency Injection container

```
public void ConfigureServices(IServiceCollection services)
{

  services.AddControllers();

  // Add Amazon S3 to  ASP.NET Core Dependency Injection
  services.AddAWSService<Amazon.S3.IAmazonS3>();

}
```

Configure DI

Use Service Client

```
public class HomeController (IAmazonS3Client s3Client)
  : Controller
{

    private readonly IAmazonS3 _s3Client  = s3Client;

    // Use "_s3Client" in your code

}
```

© 2024, Amazon Web Services, Inc. or its Affiliates.

aws

.NET Tools in AWS
Command Line Tools

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS 명령줄 인터페이스(CLI)

Python 기반 CLI

Install via PIP or
.msi from AWS site

"원라이너" 또는 셸 스크립트를
사용하여
AWS 리소스 관리

```
$ aws ec2 describe-instances

$ aws ec2 start-instances --instance-ids i-1348636c

$ aws sns publish --topic-arn arn:aws:sns:us-east-1:546419318123:OperationsError --message "Script Failure"

$ aws sqs receive-message --queue-url https://queue.amazonaws.com/546419318123/Test
```

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Tools for PowerShell

AWS Tools for PowerShell

- 윈도우 **파워셸** 및 **파워셸 코어에서** 사용 가능
- AWS 리소스 및 서비스 관리에 사용
  - **150개 이상의 서비스에서 5,000개 이상의 cmdlet 사용**
- PowerShell 갤러리에서 배포: https://www.powershellgallery.com/packages?q=AWS

AWSPowerShell

- Windows PowerShell v2-v5.1
- 아마존에서 제공하는 EC2 Windows 이미지

AWSPowerShell.NetCore

- Windows PowerShell v3+
- PowerShell Core 6+ (Windows, macOS 및 Linux)

모듈형 변형도 사용할 수 있습니다. 별도의 작은 모듈(예: **AWS.Tools.EC2**, **AWS.Tools.S3**...)을 통해 각 AWS 서비스를 관리할 수 있습니다.

SDK에서 무언가를 코딩할 수 있다면 PowerShell에서 스크립팅할 수 있습니다!

```
PS C:\> Start-EC2Instance -InstanceIds i-10a567548
```

© 2024, Amazon Web Services, Inc. or its Affiliates.

‘dotnet’ CLI extensions

‘dotnet’ CLI extensions

닷넷 CLI 확장을 통해 .NET Core 애플리케이션을 빌드하고 **AWS 서비스에 게시하는** 프로세스를 **간소화합니다**.

크로스 플랫폼 '닷넷 CLI' 확장

- Replicate the deployment wizards inside Visual Studio
- Tools for Lambda, Elastic Beanstalk, and ECS
- Use on developer workstation or in CI/CD scenarios

Round-trip capable with Visual Studio integration

- ‘defaults’ file (json) carries deployment settings
- Code shared between CLI tools and IDE

Simple install: **dotnet tool install –g *<toolname>***

```
dotnet tool install -g Amazon.Lambda.Tools

dotnet tool update -g Amazon.Lambda.Tools
```

https://github.com/aws/dotnet

© 2024, Amazon Web Services, Inc. or its Affiliates.

‘dotnet’ CLI tools—Lambda, ECS, and Elastic Beanstalk

‘dotnet’ CLI extensions

```
steve-dev@us-west-2 C:\codebase
>dotnet lambda help
Amazon Lambda Tools for .NET Core applications (3.2.2)
Project Home: https://github.com/aws/aws-extensions-for-dotnet-cli, https://github.com/aws/aws-lambda-dotnet


Commands to deploy and manage AWS Lambda functions

        deploy-function          Command to deploy the project to AWS Lambda
        invoke-function          Command to invoke a function in Lambda with an optional input
        list-functions           Command to list all your Lambda functions
        delete-function          Command to delete a Lambda function
        get-function-config      Command to get the current runtime configuration for a Lambda function
        update-function-config   Command to update the runtime configuration for a Lambda function

Commands to deploy and manage AWS Serverless applications using AWS CloudFormation

        deploy-serverless        Command to deploy an AWS Serverless application
        list-serverless          Command to list all your AWS Serverless applications
        delete-serverless        Command to delete an AWS Serverless application

Commands to publish and manage AWS Lambda Layers:

        publish-layer            Command to publish a Layer that can be associated with a Lambda function
        list-layers              Command to list Layers
        list-layer-versions      Command to list versions for a Layer
        get-layer-version        Command to get the details of a Layer version
        delete-layer-version     Command to delete a version of a Layer

Other Commands:

        package                  Command to package a Lambda project into a zip file ready for deployment
        package-ci               Command to use as part of a continuous integration system.

To get help on individual commands execute:
        dotnet lambda help <command>

steve-dev@us-west-2 C:\codebase
>
```

```
steve-dev@us-west-2 C:\codebase
>dotnet ecs help
Amazon EC2 Container Service Tools for .NET Core applications (3.0.0)
Project Home: https://github.com/aws/aws-extensions-for-dotnet-cli


Commands to deploy to Amazon EC2 Container Service:

        deploy-service          Push the application to ECR and runs the application as a long lived service on the ECS Cluster.
        deploy-task             Push the application to ECR and then runs it as a task on the ECS Cluster.
        deploy-scheduled-task   Push the application to ECR and then sets up CloudWatch Event Schedule rule to run the application.

Commands to manage docker images to Amazon EC2 Container Registry:

        push-image              Execute "dotnet publish", "docker build" and then push the image to Amazon ECR.

To get help on individual commands execute:
        dotnet ecs help <command>

steve-dev@us-west-2 C:\codebase
>
```

```
steve-dev@us-west-2 C:\codebase
>dotnet eb help
Amazon Elastic Beanstalk Tools for .NET Core applications (3.1.0)
Project Home: https://github.com/aws/aws-extensions-for-dotnet-cli


Commands to manage environments:

        deploy-environment      Deploy the application to an AWS Elastic Beanstalk environment.
        list-environments       List the AWS Elastic Beanstalk environments.
        delete-environment      Delete an AWS Elastic Beanstalk environment.

Other Commands:

        package                 Package the application to a zip file to be deployed later to an Elastic Beanstalk environment

To get help on individual commands execute:
        dotnet eb help <command>

steve-dev@us-west-2 C:\codebase
>
```

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS 서버리스 애플리케이션 모델

AWS Serverless Application Model

- 서버리스에 최적화된 **CloudFormation 확장 기능**
- **새로운 서버리스 리소스 유형**: 함수, API, 테이블
- **PowerShell Core** 및 **C#/.NET 6.0으로** 작성된 **서버리스** 애플리케이션 지원
- **Open specification** (Apache 2.0)

https://github.com/awslabs/serverless-application-model

© 2024, Amazon Web Services, Inc. or its Affiliates.

.NET Tools in AWS CI/CD Integration

© 2024, Amazon Web Services, Inc. or its Affiliates.

Amazon CodeCatalyst

Amazon CodeCatalyst

UNIFIED PLATFORM FOR TEAMS TO GO FROM PLANNING TO PRODUCTION FASTER, ON AWS

- 소프트웨어 개발 팀을 위한 클라우드 기반의 통합 협업 서비스
- 제공된 .NET용 애플리케이션 블루프린트를 사용하여 프로젝트를 만들거나 직접 빌드
- 통합 소스 제어를 사용하거나 기존 GitHub 리포지토리에 연결
- Amazon Q 와 통합 포함
- 포함된 무료 티어로 시작하기

... 그리고 더!

https://codecatalyst.aws

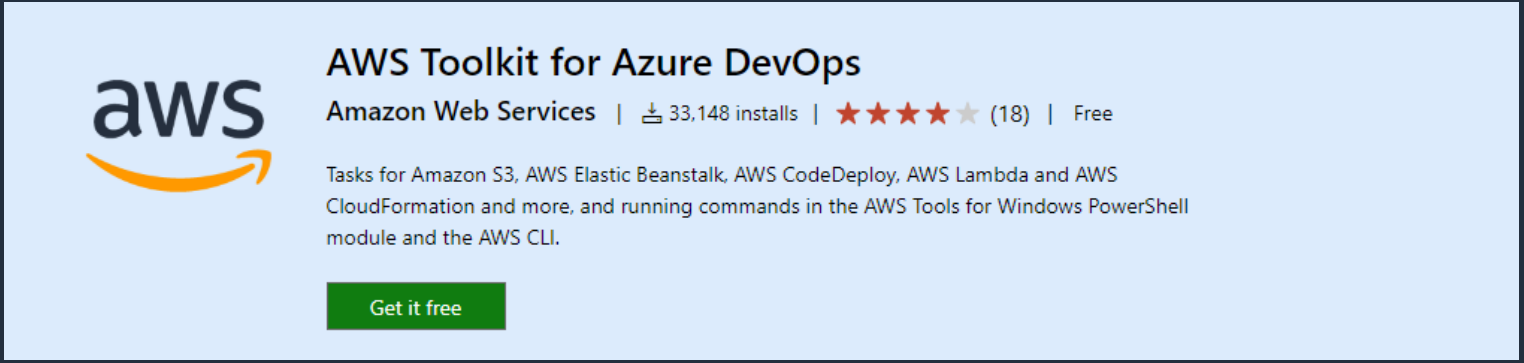
© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Tools for Azure DevOps

AWS Tools for Azure DevOps

온라인 Azure DevOps(이전: Visual Studio 팀 서비스) 및
온-프레미스 Azure DevOps Server(TFS 2017 업데이트 1 이상)

https://marketplace.visualstudio.com/items?itemName=AmazonWebServices.aws-vsts-tools

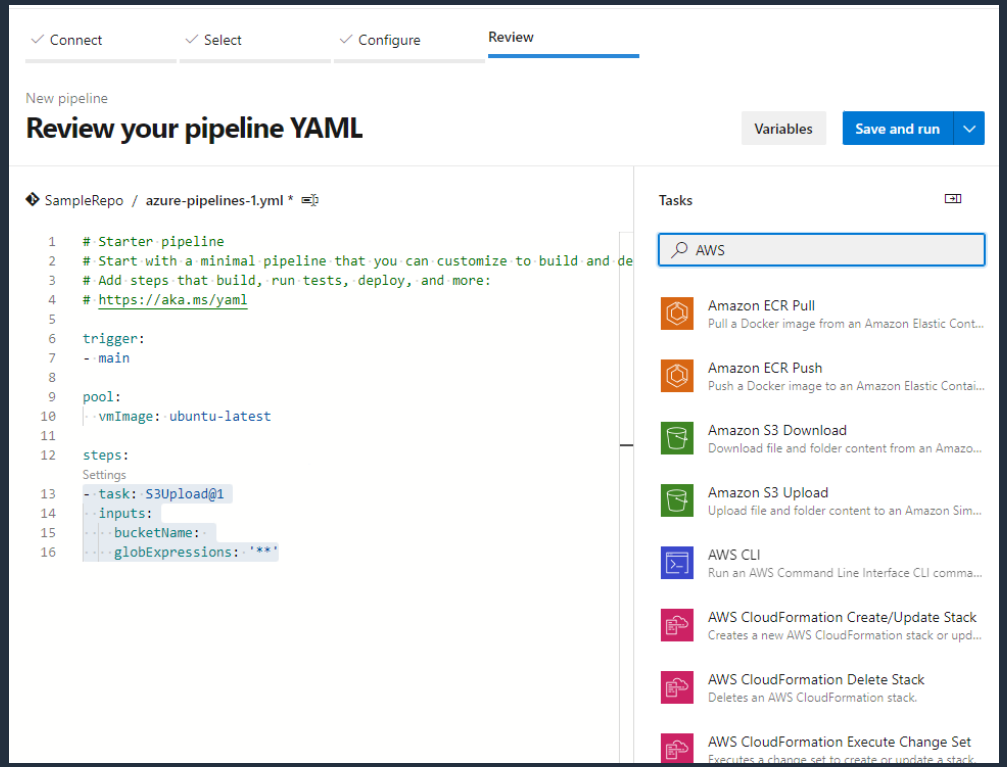


https://aws.amazon.com/vsts/

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Tools for Azure DevOps (VSTS & TFS)

AWS Tools for Azure DevOps



Collection of tasks for working with:
Elastic Beanstalk
CloudFormation
CodeDeploy
Elastic Container Registry
S3 (upload and download)
Secrets Manager
Parameter Store
Lambda (all languages and .NET Core specific task)
AWS CLI / AWS Tools for PowerShell / AWS bash shell

© 2024, Amazon Web Services, Inc. or its Affiliates.

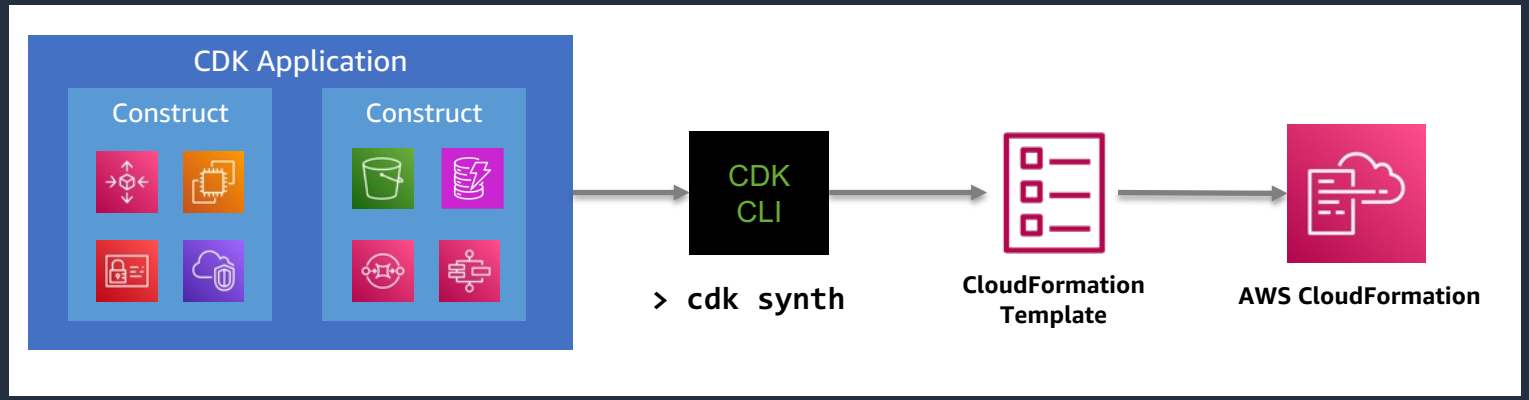
AWS Cloud Development Kit (CDK)

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Cloud Development Kit (CDK)

AWS CDK for .NET

- AWS CDK는 **클라우드 인프라를 코드로** 정의하고 CloudFormation을 통해 프로비저닝하기 위한 소프트웨어 개발 프레임워크입니다.
- CDK는 AWS 서비스와 완벽하게 통합되며, **AWS 리소스를** 필수적으로 **정의하기** 위해 더 높은 수준의 **객체 지향** 추상화를 제공합니다.

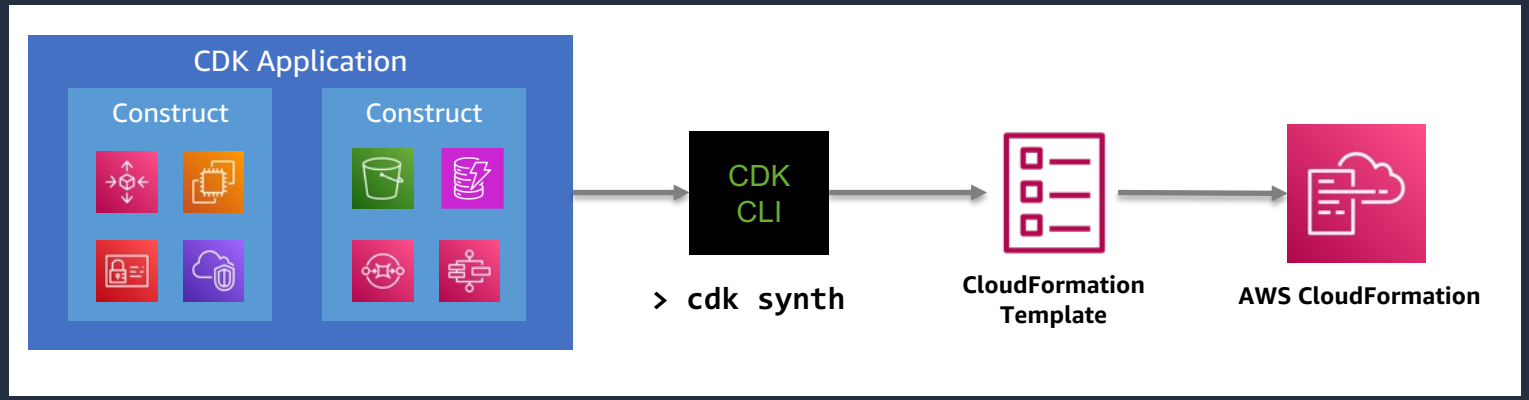


© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS Cloud Development Kit (CDK)

AWS CDK for .NET

- C#을 사용하여 클라우드 인프라 및 애플리케이션 정의 및 프로비저닝
- AWS 리소스를 필수적으로 정의하고 공유하기 위한 상위 수준의 객체 지향 추상화
- 또한 타입스크립트, 자바스크립트, 자바, 파이썬, Go(개발자 프리뷰에서)를 지원합니다.



© 2024, Amazon Web Services, Inc. or its Affiliates.

CDK 예제 코드

AWS CDK for .NET

예: 애플리케이션에서 커뮤니케이션을 위한 Amazon SQS 대기열 및 Amazon SNS 토픽을 만듭니다.

```
Queue queue = new (this, "MyFirstQueue", new QueueProps
{
    VisibilityTimeoutSec = 300,

});

Topic topic = new (this, "MyFirstTopic", new TopicProps
{
    DisplayName = "My first topic"
});
```

Example: Custom construct deploys a set of buckets, and grants read access to a user.

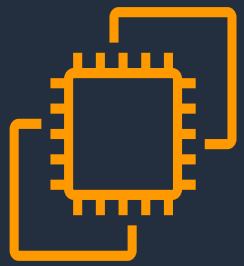
```
const int NUM_BUCKETS = 5;

...

// Custom construct creates an arbitrary number of buckets
CustomBucketConstruct custom = new(this, "buckets", new CustomBucketConstructProps
{
    BucketCount = NUM_BUCKETS
});


// New user that needs access
User user = new(this, "bucketUser", new userProps());

// Grant read access to the user to all the created buckets in aggregate
custom.GrantRead(user);
```

© 2024, Amazon Web Services, Inc. or its Affiliates.

Migrating .NET applications to AWS

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS에서 .NET 애플리케이션 호스팅



Amazon EC2

- 리호스팅
- 가상 머신 (Linux/Windows)
- 보안
- 크기 조정 가능

Containers

- 리플랫폼
- 완전 관리형, DIY 및 서버리스 컨테이너 옵션의 광범위한 범위

AWS Lambda

- 리팩터링
- 서버 프로비저닝 또는 관리 없이 코드 실행
- 동기식(요청/응답) 또는 비동기식 실행 모델

Level of abstraction

H/W → OS → Runtime

Unit of Consumption

VM → Task → Function

© 2024, Amazon Web Services, Inc. or its Affiliates.

Compute Resources: Where to Run .NET Code

가상 머신 - EC2 인스턴스

- Windows: .NET Framework 또는 .NET
- Linux: .NET
- 인스턴스 유형/크기, 메모리, 디스크, vCPU 선택
- 관리 및 패치

컨테이너 - 도커/쿠버네티스

- Windows: .NET Framework 또는 .NET
- Linux: .NET
- 간편한 패키징, 배포, 재사용 및 확장
- 인프라가 아닌 개발에 집중

Lambda

- .NET - 3.1 또는 2.1(Long Term Support)
- 컨테이너 이미지로 기능 지원 - .NET Core 2.1 , 3.1 및 .NET 5
- 단기 실행 작업(15분 미만)을 위한 서버리스 플랫폼
- 코드가 실행되는 시간에 대해서만 비용을 지불

추상화

© 2024, Amazon Web Services, Inc. or its Affiliates.

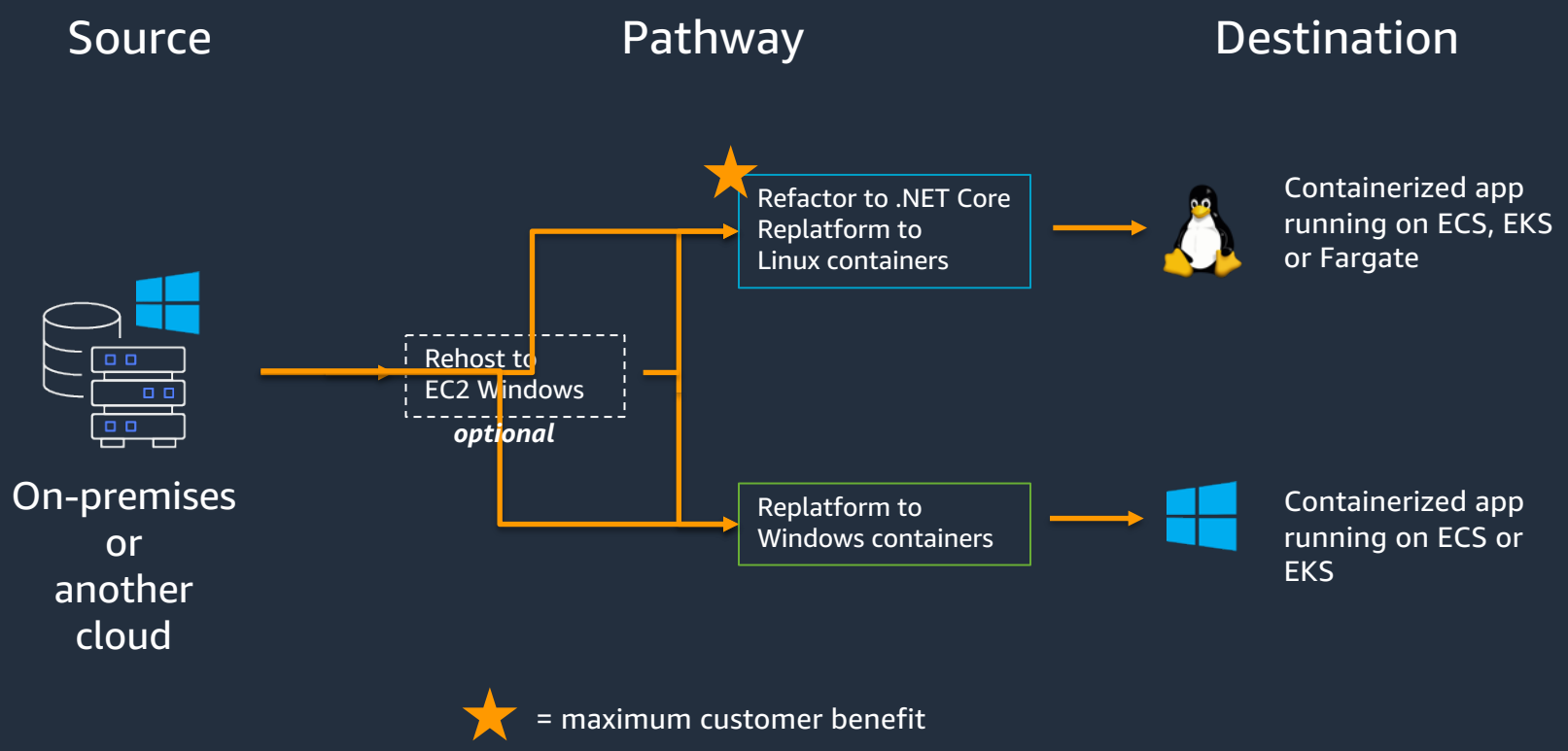
.NET Framework app migration strategies

리호스팅	Amazon EC2 가상 머신 또는 Elastic Beanstalk에 "있는 그대로" .NET Framework 앱 배포
리플랫폼	Windows 컨테이너에서 컨테이너화하고 Amazon ECS 또는 Amazon EKS로 오케스트레이션하기
리팩터링	NET Core로 포팅하고, Amazon ECS, Fargate 또는 Amazon EKS에서 Linux 컨테이너 또는 Lambda

낮음

© 2024, Amazon Web Services, Inc. or its Affiliates.

What is “Modernizing .NET Workloads”?



© 2024, Amazon Web Services, Inc. or its Affiliates.

Application Modernization Tools

© 2024, Amazon Web Services, Inc. or its Affiliates.

Application modernization tools

ASSESS → MODERNIZE

AWS Migration Hub
Strategy recommendations

Central tool across all phases of migration and modernization

Assess your modernization at scale

Modernization at scale

Application
 Recommendations

Porting Assistant for .NET

Easier and assisted refactoring to .NET Core

Minimize manual effort

Modernization pathway

.NET Framework
 .NET Core

AWS App2Container

Faster containerization for existing applications

Accelerate modernization

Modernization pathway

.NET/Java on Win/Linux
 Win/Linux containers

AWS Microservice Extractor for .NET

Reduced effort to transform monolithic applications into microservices

Accelerate refactoring

Modernization pathway

Monolith
 Microservices

© 2024, Amazon Web Services, Inc. or its Affiliates.

AWS App2Container

https://aws.amazon.com/app2container

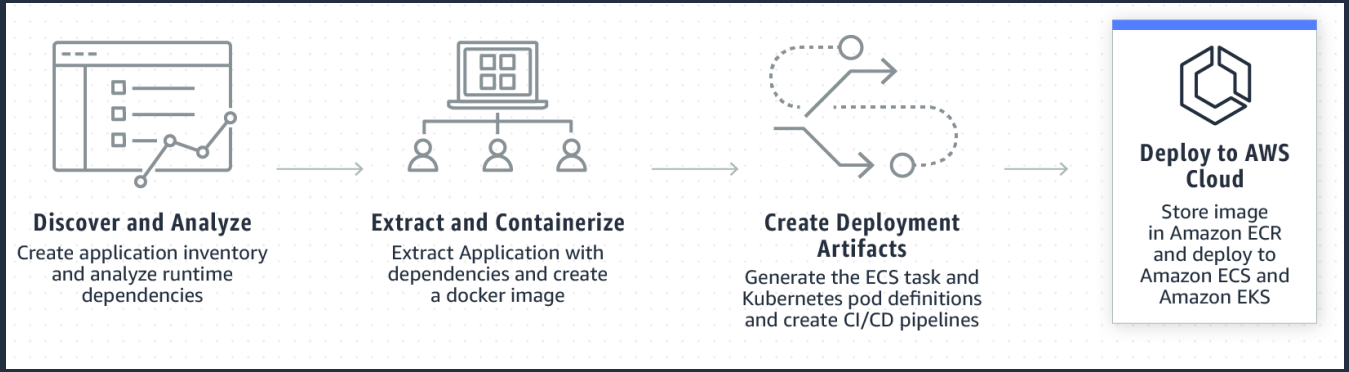
App2Container

실행 중인 애플리케이션을 컨테이너로 변환하고
AWS가 관리하는 컨테이너에서 실행하도록 도와주는 명령줄 도구

컨테이너로 전환하여 다른 개발 언어와
동일한 방식으로
표준화된 모니터링 및 운영 가능

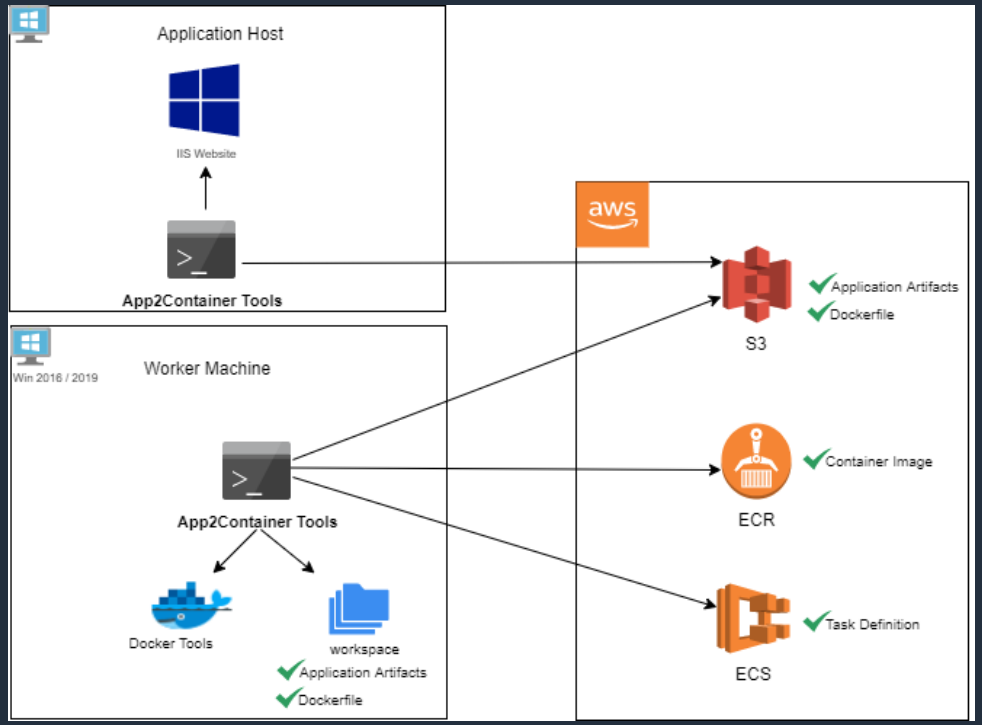
애플리케이션을 분석하고
지정한 배포 환경에 맞는 방식으로
컨테이너 이미지 자동 생성

올바른 네크워킹 및 보안에 맞게 잘
구성된 클라우드 환경에
기존 애플리케이션 용이하게 배포



© 2024, Amazon Web Services, Inc. or its Affiliates.

App2Container 동작 방식



1. 현재의 애플리케이션을 분석

```
app2container inventory
app2container analyze
                    --application-id <ID>
```

2. 애플리케이션 아티팩트, 도커 파일 (이미지) 생성

```
app2container containerize
                    --application-id <ID>
```

3. AWS(ECS, EKS, App Runner)로 배포를 위한 아티팩트 생성, [직접 배포]

```
app2container generate app-deployment
        --application-id <ID> [--deploy]
app2container generate pipeline
        --application-id <ID> [--deploy]
```

© 2024, Amazon Web Services, Inc. or its Affiliates.

Administrator: Windows PowerShell

```
PS C:\a2cInstaller\a2cInstaller>
```

.NET 애플리케이션을 모니터링

© 2024, Amazon Web Services, Inc. or its Affiliates.

모니터링은 장애를 감시하는 것 이상입니다

예상대로 작동하나요?

사용법은 무엇인가요?

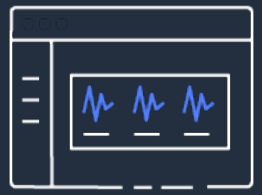
비즈니스에 미치는 영향은 무엇인가요?

© 2024, Amazon Web Services, Inc. or its Affiliates.

통합 가시성을 위한 AWS 서비스 CloudWatch 및 X-Ray

수집

- 로그
- 메트릭
- 메트릭 필
- StatsD & CollectD



모니터

- Cross-Account, Cross-Region Dashboards
- Automatic Dashboards
- Metric Math

Act

- 이상 징후 탐지
- Metric Math Alarms
- Search Expressions

분석

- ServiceLens
- Contributor Insights
- Container Insights
- Logs Insights
- X-Ray Analytics

© 2024, Amazon Web Services, Inc. or its Affiliates.

Digital Skills Badges - .NET on AWS

https://aws.amazon.com/developer/language/net/badges-and-training

© 2024, Amazon Web Services, Inc. or its Affiliates.

Resources

Monolithic to Microservice Journey for .NET Applications

.NET on AWS youtube playlist

.NET on AWS Blog

Getting started with AWS App2Container

© 2024, Amazon Web Services, Inc. or its Affiliates.

감사합니다!

© 2024, Amazon Web Services, Inc. or its Affiliates.