

Azure Functions

They allow you to run code in response to events or triggers.
Ideal for building microservices and IoT solutions

Benefits

- **Serverless:** No need to manage infrastructure.
- **Cost-effective:** Pay only for what you use.
- **Scalable:** Automatically scale based on demand.
- **Event-driven:** Trigger functions based on events.
- **Language support:** Various programming languages.
- **Integration:** Seamless integration with Azure services

Key Concepts of Azure Functions

Function App: A logical container for functions.

Trigger: An event that starts the function.

Input and Output Bindings: Connect to data sources.

Stateless: Functions are stateless by design.

Execution Context: Information about the function's environment.

Monitoring and debugging

- Azure Monitor helps you track function performance.
- Application Insights provides deep insights into function behavior.
- Debugging tools in Azure Functions make it easy to identify issues.

Microsoft DevOps

DevOps is the union of people, process, and products to enable continuous delivery of value to our end users.

It is a set of practices and tools that helps teams work together to develop and deliver software.

DevOps is not a quick fix, but it can help teams deploy more frequently, reduce lead time from commit to deploy, reduce change failure rate, and recover from incidents more quickly.

Here are some of the key practices of DevOps:

- Agile planning: Creating a backlog of work that everyone on the team and in management can see, and prioritizing the items so we know what we need to work on first.
- Continuous integration (CI): Automating how we build and test our code, and running that every time a team member commits changes to version control.
- Continuous delivery (CD): Testing, configuring, and deploying from a build to a QA or production environment.
- Monitoring: Using telemetry to get information about an application's performance and usage patterns, and using that information to improve as we iterate.

DevOps can help teams experiment with ways to increase customer adoption and satisfaction, and it can lead to better organizational performance, and often to higher profitability and market share.

Here are some of the benefits of DevOps:

Deploy more frequently

In fact, some teams deploy up to dozens of times per day.

Practices such as monitoring, continuous testing, database change management, and integrating security earlier in the software development process help elite performers deploy more frequently, and with greater predictability and security

Reduce lead time from commit to deploy

Lead time is the time it takes for a feature to make it to the customer. By working in smaller batches, automating manual processes, and deploying more frequently, elite performers can achieve in hours or days what once took weeks or even months.

Reduce change failure rate

A new feature that fails in production or that causes other features to break can create a lost opportunity between you and your users. As high-performing teams mature, they reduce their change failure rate over time.

Recover from incidents more quickly

When incidents do occur, elite performers are able to recover more quickly. Acting on metrics helps elite performers recover more quickly while also deploying more frequently.

If you are interested in learning more about DevOps, there are many resources available online and in libraries. You can also find DevOps training courses and certifications.

A suite of services that provide an end-to-end tool chain

Azure DevOps can help you throughout your application's entire lifecycle — from planning all the way to deploying your application.

Azure DevOps provides several tools you can use for better team collaboration. It also has tools for automated build processes, testing, version control, and package management.

can we use something for Azure if we're not deploying to the cloud? Plus, we deploy to Linux. Does that matter?

Mara: These tools are great whether you're in the cloud or on-premises. It also doesn't matter if we're deploying to Linux or Windows or another platform. Azure DevOps is a suite of services that provides a solution for anyone who wants an enterprise-grade tool chain. Those tools will help us implement all the practices we just talked about.

5 services:

Pipelines: These will let us build, test, and deploy with CI/CD that works with any language, platform, and cloud.

Boards: These are agile tools that help us plan, track, and discuss our work, even with other teams.

Testplans: These are manual and exploratory testing tools

Repos: These provide unlimited, cloud-hosted private, and public Git repos.

Artifacts: These let us create, host, and share packages.

Next, set up an organization. Here's how:

1. If you've never created an Azure DevOps organization, you'll see a window with a **Create new organization** button. If you have, you'll see a link that reads **New organization**. Select the option you see.
2. In the **Azure DevOps Terms of Service and Privacy notification** window, select **Continue**.
3. Create an organization for the Microsoft Learn modules next to the **dev.azure.com/** field. If you're prompted that the name is already taken, just add some numbers to the end to make it unique. For example: Tailspin0523.
4. Choose a location near you where your projects will be hosted.
5. Complete the captcha.
6. Select **Continue**.

Github

GitHub is a development platform that lets you host and review code, manage projects, and build software alongside 50 million developers.

Why is everyone building on GitHub? Because it provides the important DevOps features companies and organizations of all sizes need for their public and private projects. Whether it's planning features, fixing bugs, or collaborating on changes, GitHub is the place where the world's software developers gather to make things and then make them better.

Git is a distributed version-control system (DVCS) that allows multiple developers or other contributors to work on a project. It provides a way to work with one or more local branches and push them to a remote repository

GitHub is a cloud platform that uses Git as its core technology.

Key GitHub features include:

- **Issues:** Issues are where most of the communication between a project's consumers and development team occurs.
- **Discussions**
- **Pull requests** mechanism used to signal that the commits from one branch are ready to be merged into another branch.
- **Notifications:** GitHub offers notifications for virtually every event that takes place within a given workflow.
- **Labels:** Labels provide a way to categorize and organize issues and pull requests in a repository.
- **Actions:** task automation and workflow functionality in a repository. You can use actions to streamline processes in your software-development lifecycle and implement continuous integration and continuous deployment (CI/CD).

- Projects
- Branches: Branches are the preferred way to create changes
- Commit: A commit is a snapshot of a change or changes to one or more files on a branch.
- Forking: Forking a repository makes a copy of the repository in your GitHub account. The parent repository is referred to as the upstream, while your forked copy is referred to as the origin.
- Cloning: cloning a repository makes a copy of the repository and its history on your local machine
- GitHub Pages: is a hosting engine that's built right into your GitHub account

When would you clone a repository versus fork a repository?

If you're working with a repository and have write access, you can clone it to your local machine. From there, you can make modifications and push your changes directly to the origin repository. If you need to work with a repository created by another owner such as github/example and don't have write access, you can fork the repository into your GitHub account, then clone the fork to your local machine.

you can make changes to your local copy , then push them back to your remote origin repository. You can then submit these changes to the github/example upstream repository using a pull request

Azure Cloud

Azure offers a broad set of global cloud-based products including compute, storage, database, analytics, networking, machine learning and AI, mobile, developer tools, IoT, security, enterprise applications, and much more

What is cloud Computing?



