Agile Devops Tools – Azure Artifacts (NuGet, NPM) – Package Management

[**🎯 Purpose** 1](#_Toc205557541)

[**📘 Theory** 1](#_Toc205557542)

[**🧰 Prerequisites** 2](#_Toc205557543)

[**🔧 Step-by-Step – Create and Use Azure Artifacts** 2](#_Toc205557544)

[**📸 Project Structure Snapshot (Example)** 4](#_Toc205557545)

[**✅ Summary** 4](#_Toc205557546)

**✅ Task 5: Azure Artifacts (NuGet, NPM) – Package Management**

**🎯 Purpose**

Azure Artifacts allows teams to **create, host, and share** packages (like NuGet, npm, Maven, etc.) from private or public sources, making it easier to **manage dependencies** across multiple projects and teams.

**📘 Theory**

**What is Azure Artifacts?**

Azure Artifacts is a service within **Azure DevOps** that provides **integrated package management**. It supports:

* **NuGet** (for .NET)
* **npm** (for Node.js)
* **Maven** (for Java)
* **Universal Packages** (any binary)

**Key Features:**

| **Feature** | **Description** |
| --- | --- |
| Feed Creation | Feeds are like package repositories. Each feed can host multiple packages. |
| Scoping | Set permissions (project, org, team) for feeds. |
| Upstream Sources | Connect to external package sources (like NuGet.org, npmjs.com). |
| Retention Policies | Automatically clean up unused packages. |
| Integration with Pipelines | Use packages easily in CI/CD workflows. |

**Typical Use Cases:**

* Publish internal libraries/packages (NuGet, npm)
* Share common code across teams securely
* Manage third-party dependencies with better control
* Cache public packages for faster builds

**🧰 Prerequisites**

* Azure DevOps account and project
* Basic knowledge of Git and CI/CD
* Installed tools: Visual Studio or npm/nuget CLI

**🔧 Step-by-Step – Create and Use Azure Artifacts**

**✅ Step 1: Enable Azure Artifacts Extension (if not already enabled)**

Go to your **Azure DevOps Project**  
→ Click on **Organization Settings**  
→ Click **Extensions** → Ensure **Azure Artifacts** is enabled

**✅ Step 2: Create a Feed**

1. Navigate to **Artifacts** (left menu)
2. Click on **“+ New feed”**
3. Name your feed (e.g., MyCompanyFeed)
4. Set visibility:
   * Private to your team/project
   * Organization-wide
5. Create feed

**✅ Step 3: Connect Feed with Tools**

**For NuGet (.NET)**  
In Visual Studio → Tools → NuGet Package Manager → Package Sources  
Add new source:

Name: MyFeed

Source: https://pkgs.dev.azure.com/{organization}/{project}/\_packaging/{feed-name}/nuget/v3/index.json

Then add the following to nuget.config:

<configuration>

<packageSources>

<add key="MyFeed" value="https://pkgs.dev.azure.com/ORG\_NAME/PROJECT\_NAME/\_packaging/FEED\_NAME/nuget/v3/index.json" />

</packageSources>

</configuration>

**For npm (Node.js)**  
Run the following:

npm login --registry=https://pkgs.dev.azure.com/ORG\_NAME/\_packaging/FEED\_NAME/npm/registry/ --scope=@YOUR\_SCOPE

Update .npmrc:

@YOUR\_SCOPE:registry=https://pkgs.dev.azure.com/ORG\_NAME/\_packaging/FEED\_NAME/npm/registry/

**✅ Step 4: Publish a Package**

**.NET (NuGet)**

nuget pack YourLibrary.csproj

nuget push YourLibrary.nupkg -Source "MyFeed"

**npm**

npm publish --registry https://pkgs.dev.azure.com/ORG\_NAME/\_packaging/FEED\_NAME/npm/registry/

**✅ Step 5: Consume the Package**

**.NET**

* In any project → Install via NuGet Package Manager → Select your feed

**npm**

npm install @yourscope/yourpackage

**✅ Step 6: Use in Azure Pipeline (Optional)**

Example YAML snippet:

steps:

- task: NuGetAuthenticate@1

- script: dotnet restore YourProject.sln

Or for npm:

steps:

- task: npmAuthenticate@0

inputs:

workingFile: .npmrc

- script: npm install

**📸 Project Structure Snapshot (Example)**

Azure DevOps Project

├── Pipelines

├── Repos

├── Artifacts

│ └── MyCompanyFeed

│ ├── MyLibrary.1.0.0.nupkg

│ └── @yourscope/my-npm-lib@1.0.0

**✅ Summary**

| **Area** | **Details** |
| --- | --- |
| Tool | Azure Artifacts |
| Purpose | Manage internal/external package dependencies |
| Supports | NuGet, npm, Maven, Universal |
| Use Case | Publishing and consuming packages securely |
| Integrated with | Azure Pipelines, Visual Studio, npm |