

WHITEPAPER



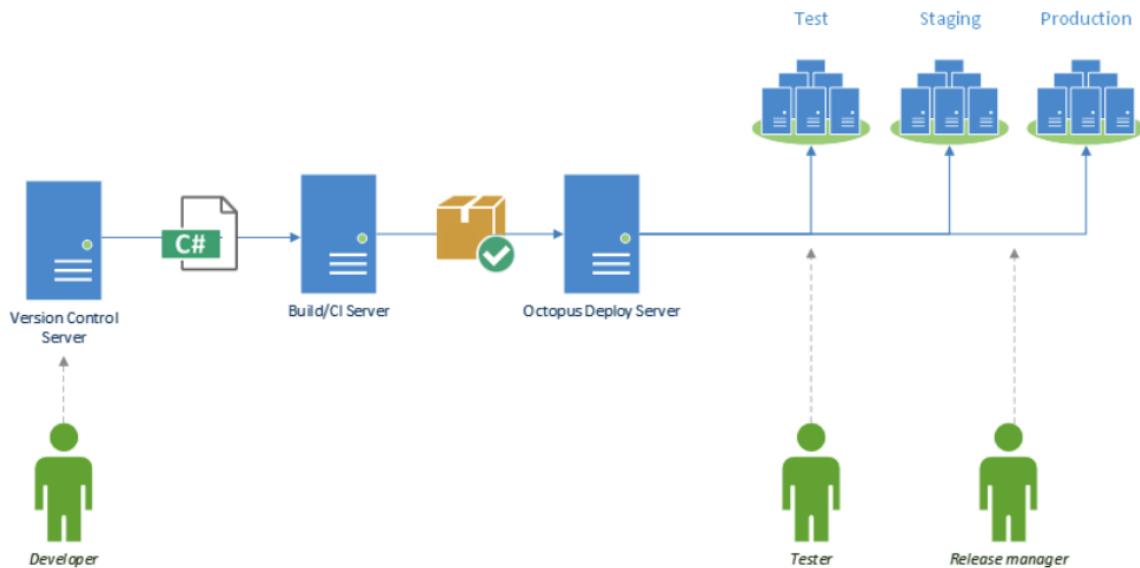
Octopus Deploy

Automated deployment for Packages

- ❖ Octopus is a deployment automation server.
- ❖ All your deployments, in one place.
- ❖ Deploy on-premises or to the cloud, securely.
- ✓ **Demo** on how **Octopus Deploy** versions the packages created manually and deploys it to different environments -->

Octopus Deploy is a user-friendly automated deployment server designed and optimized for .NET developers.

It is built with the **goal** of making repeatable, reliable automated deployments available to everyone.



Process

Developers commit their code into your existing source control system→

- You might be using Git, Team Foundation Server, Subversion or Mercurial; the choice is up to you.

Your CI/build server compiles the code and runs unit tests→

- Again, you might be using TeamCity, Jenkins, Bamboo, Team Foundation Server or CruiseControl.NET; the choice is up to you.

Your application is packaged into a NuGet package→

- When the build is done, your CI/build server bundles all of the files – the binaries, images, scripts, configuration files and so on – needed to deploy your application into a NuGet package

The job of Octopus, is to take these packages and push them to the machines that they will be deployed to.

In Octopus Deploy, we are provided a shared deployment dashboard that team members can use to easily answer questions like:

- What version of an application is deployed in each environment?
- When the application was last deployed, and who deployed it?
- How long the deployment took?
- What went wrong with the deployment?

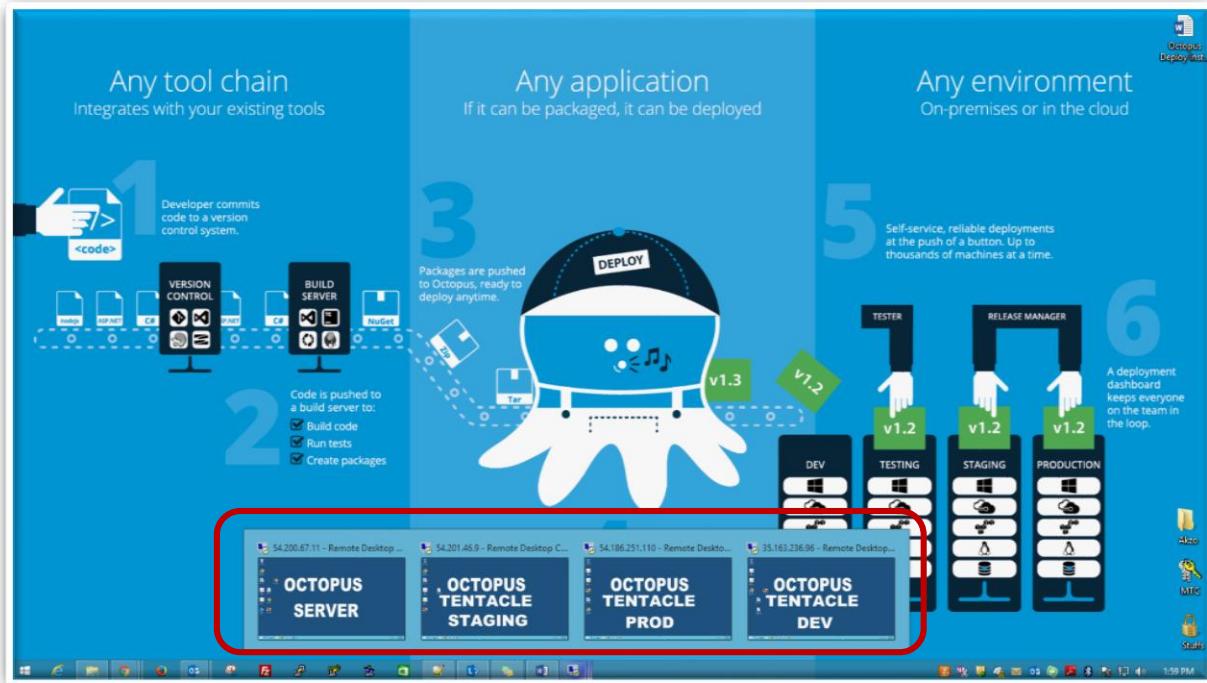
This web-based interface is also where deployments are configured and initiated, where configuration variables are managed, and where environments and machines are maintained.

How does it work?

We have already installed Octopus Server, it's a simple web based system which is easily installed by a friendly setup process. After that is done the next step is to install a small agent service called a Tentacle on each target server we are intending to deploy to.

Here are my four windows machines where I have launched:

1. Octopus Master
2. Octopus Dev Tentacle
3. Octopus Staging Tentacle
4. Octopus Prod Tentacle



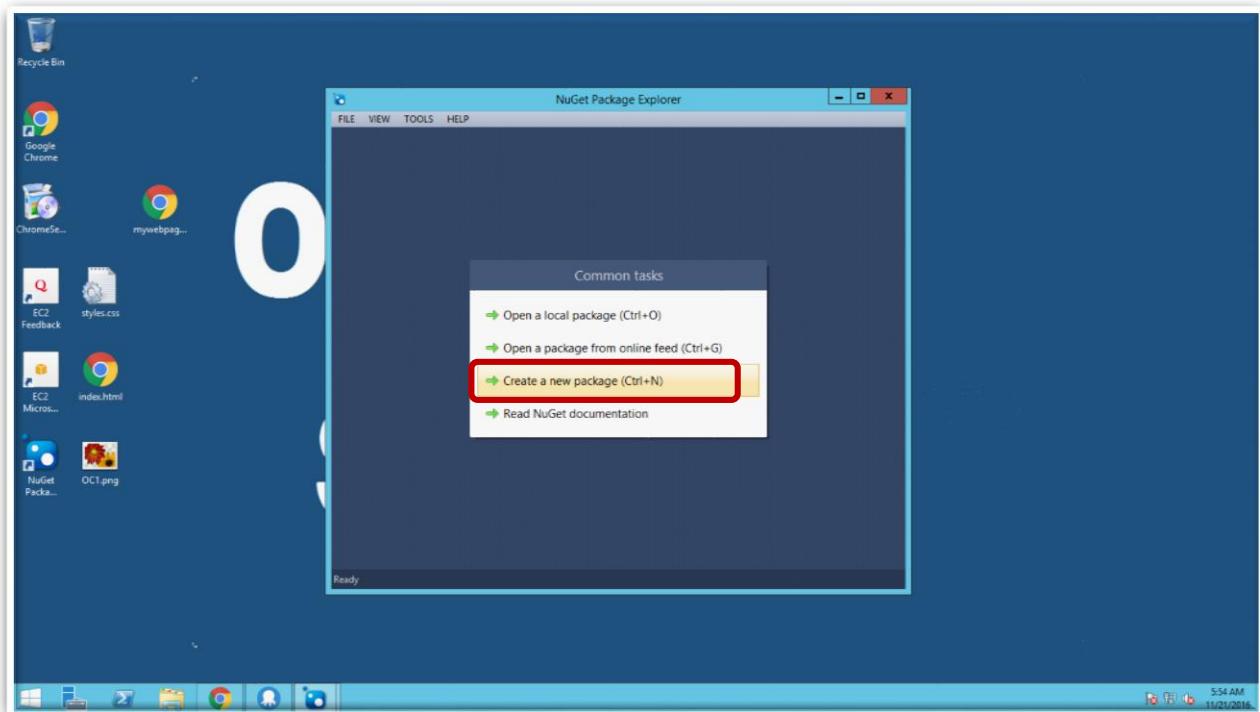
I have installed Octopus server in Octopus Master, created three environments and added one machine in each environment.

Note → The installations of Octopus server and Octopus Tentacles are explained in another Document. Interested can mail me to get the same.

The image shows the Octopus Deploy web interface. At the top, there's a navigation bar with links for Dashboard, Projects, Environments, Library, Tasks, administrator, Configuration, and Help. Below the navigation, there are three main sections: 'Development', 'Staging', and 'Production'. Each section contains a list of machines. In the 'Development' section, there is one machine named 'Web001'. In the 'Staging' section, there is one machine named 'Web01'. In the 'Production' section, there is one machine named 'WEB03'. Each machine entry has an 'Edit' button, a 'Check health' button, and a 'Add deployment target' button.

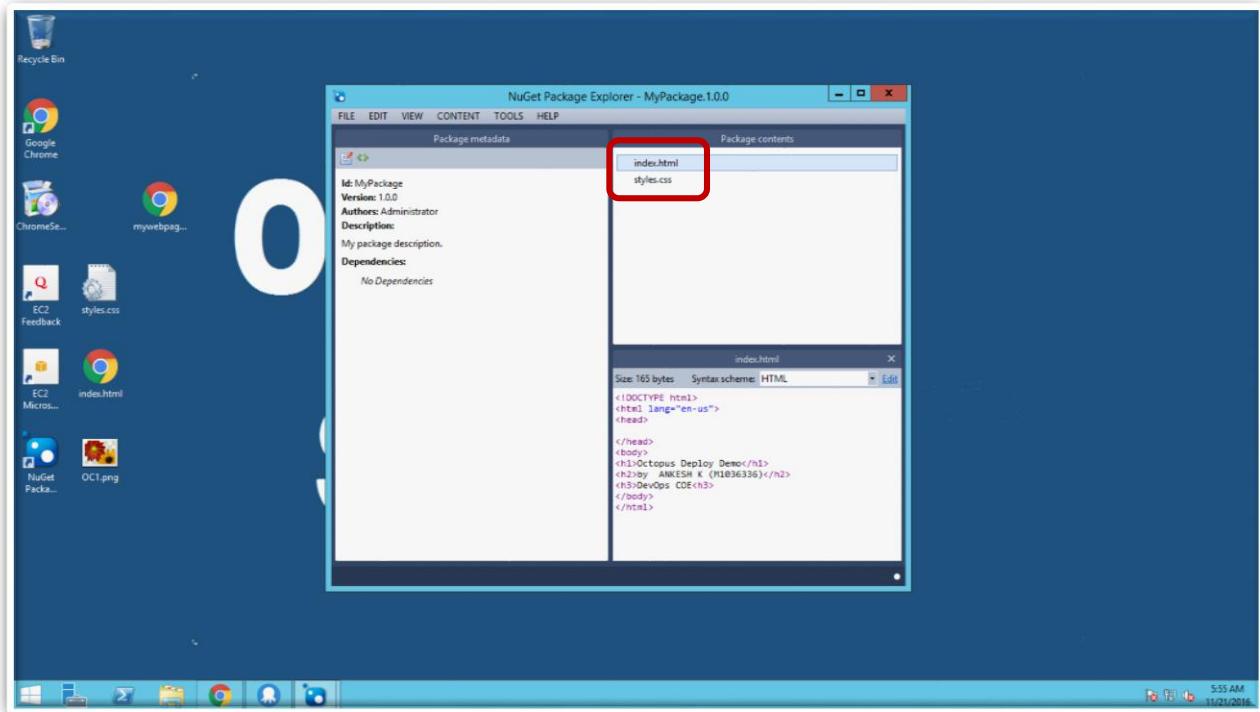
Octopus deploys the NuGet packages. So we will download NuGet Package Explorer and create a new package.

Click > **Create a new package (Ctrl+N)**

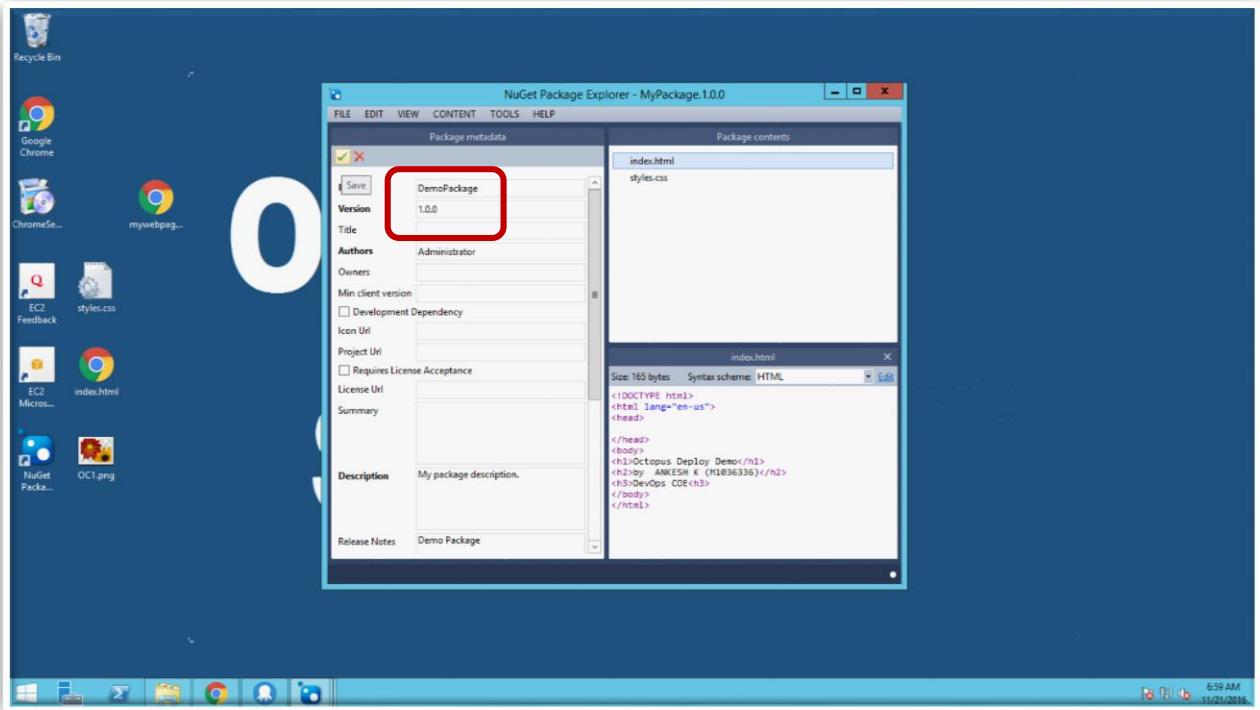


We will create an html webpage (here named as **index.html**) and insert our file inside the **Package contents** box of NuGet Package Explorer.

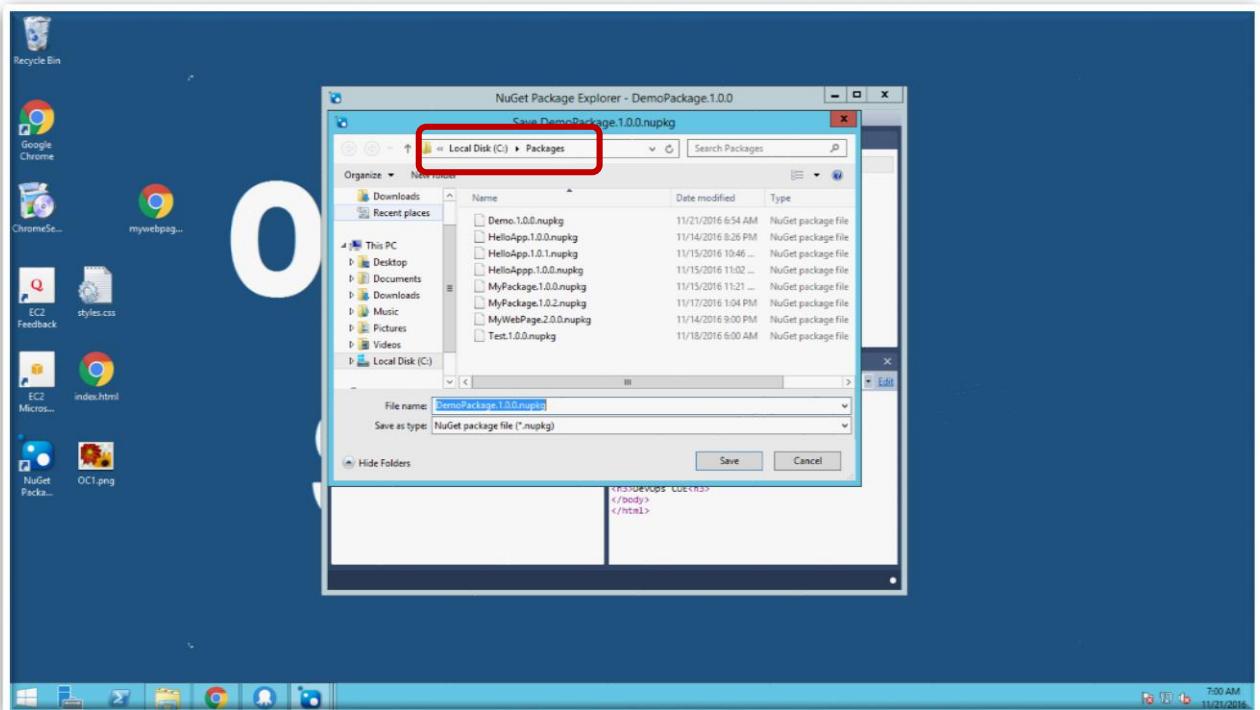
Right click on the file and click **View** to see the file.



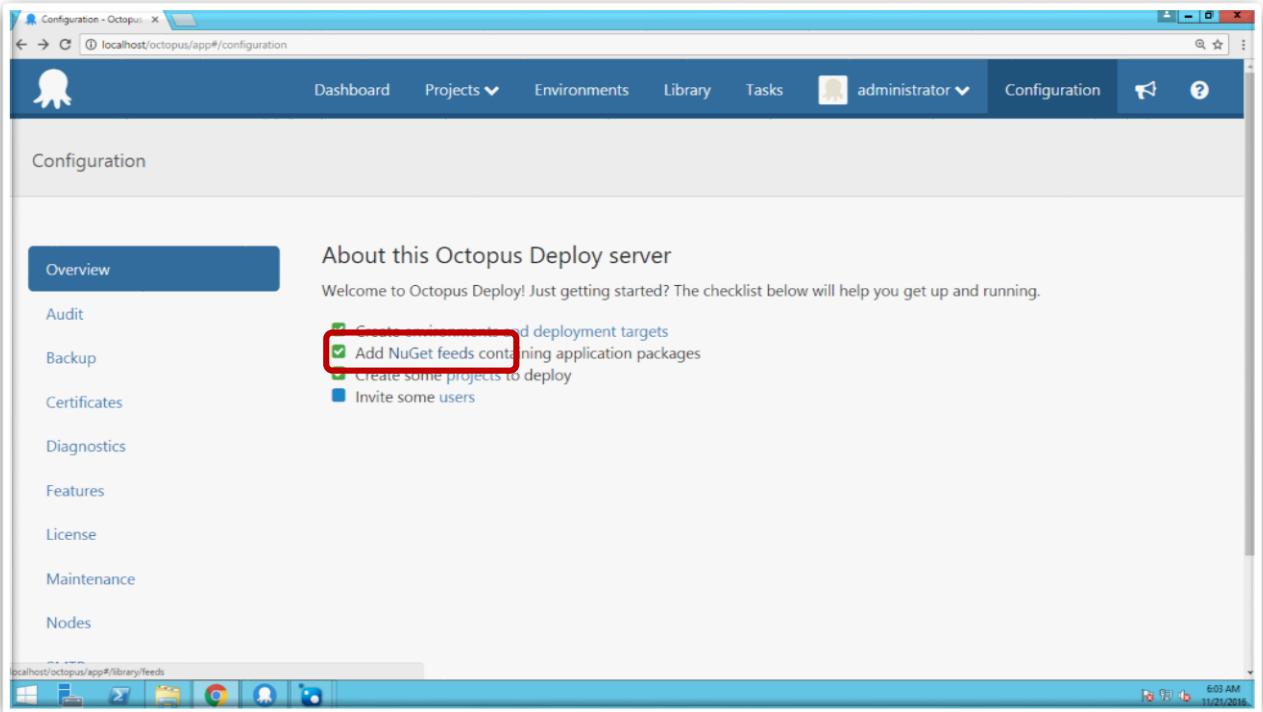
We can change the package name (**DemoPackage**) and version number (**1.0.0**) which helps us in versioning of the codes and packages.



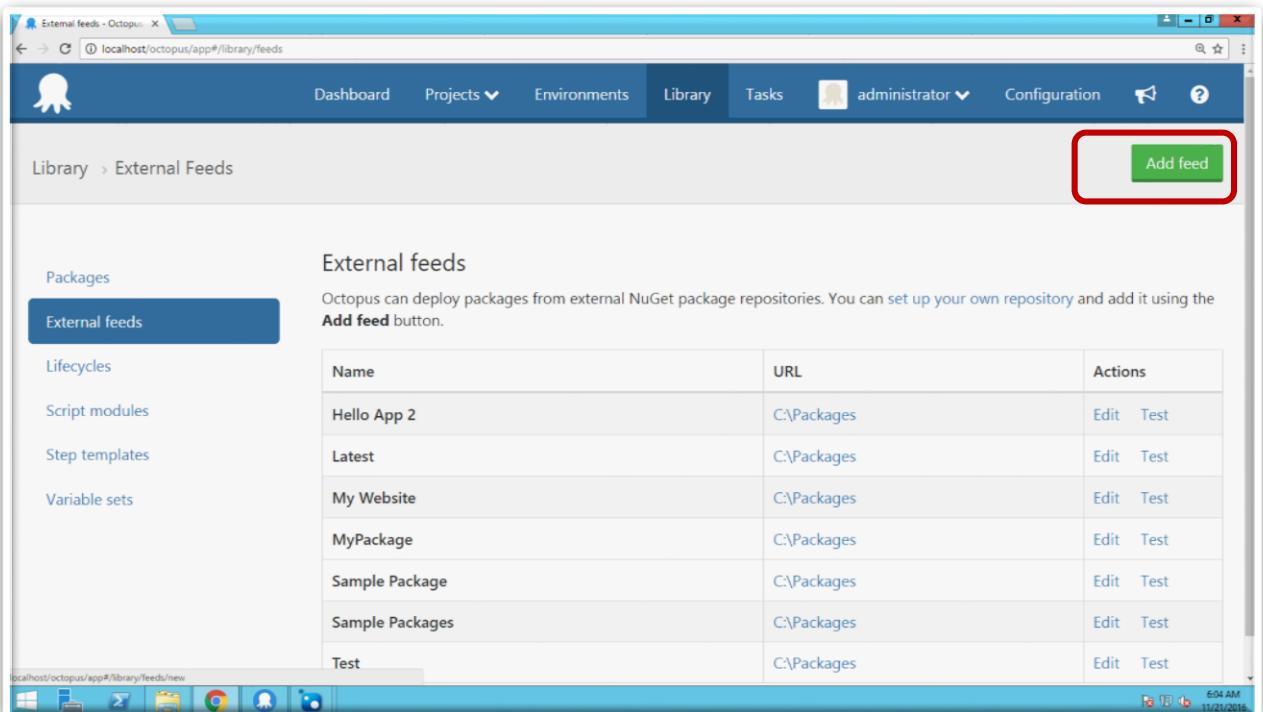
We will then save it in a folder **C:\Packages** (you can create your own folder where NuGet packages will be saved).



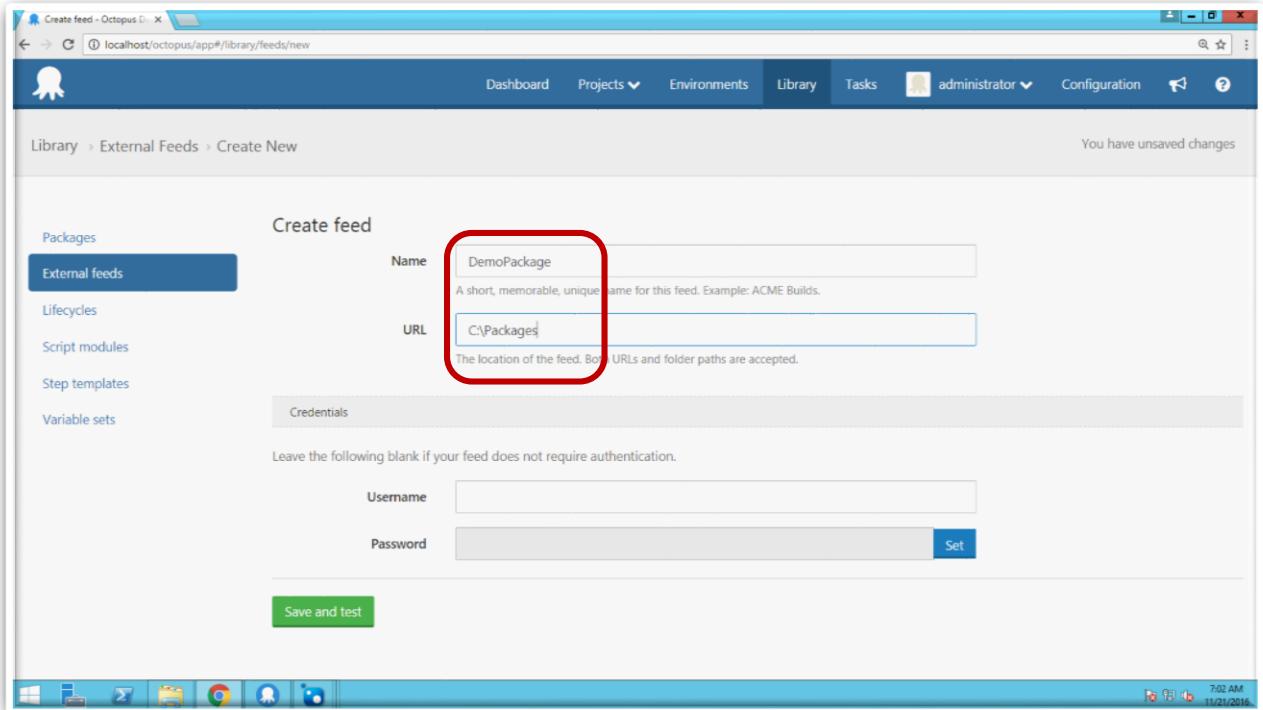
We will add this package in the **Configuration** of Octopus Server so that it can understand from where it has to take the package. Click > **Add NuGet feeds**.



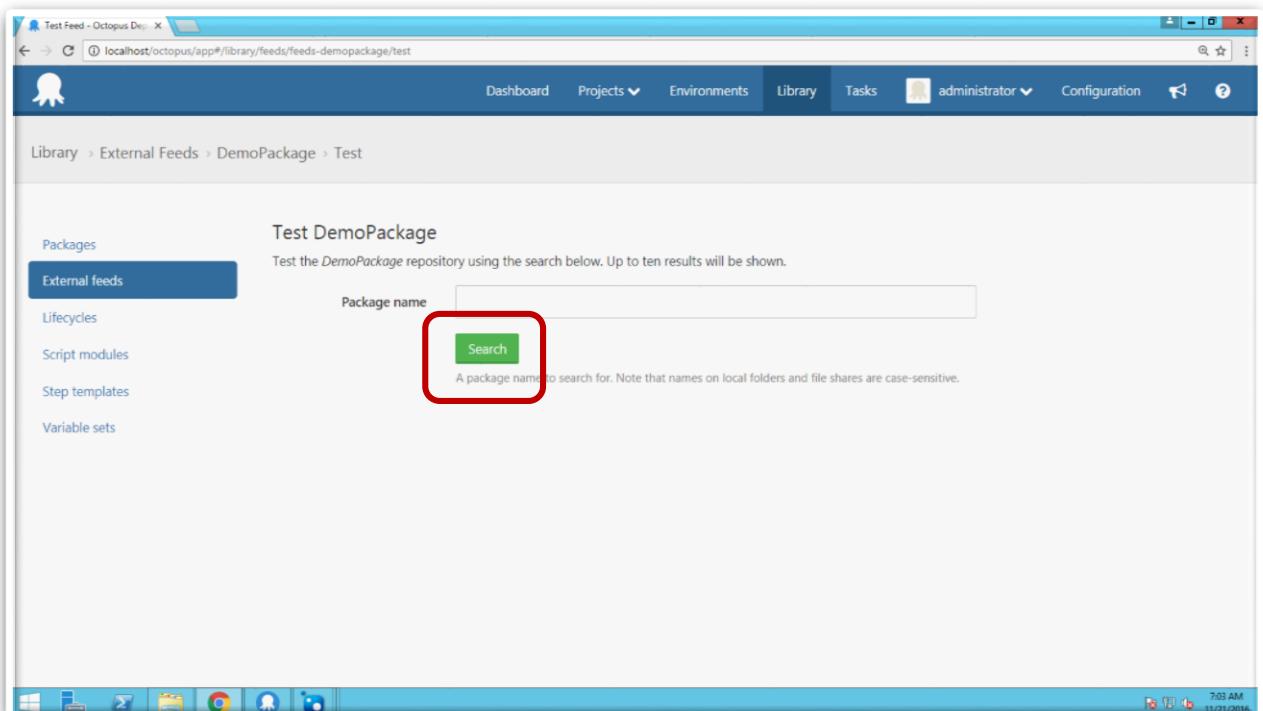
Click > **Add feed**



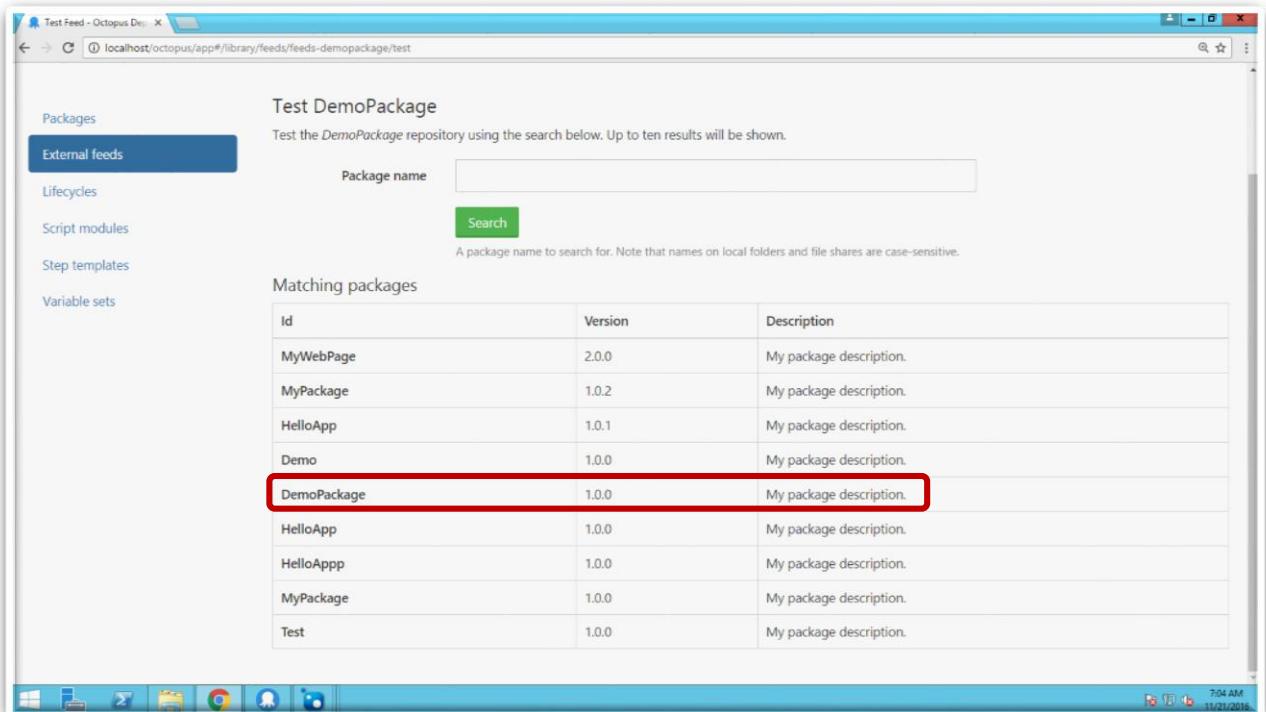
Give your feed a proper name (**DemoPackage**) and the path (**C:\Packages**) where you have stored your packages and save it.



To check whether the package is added or not, click > **Search**.



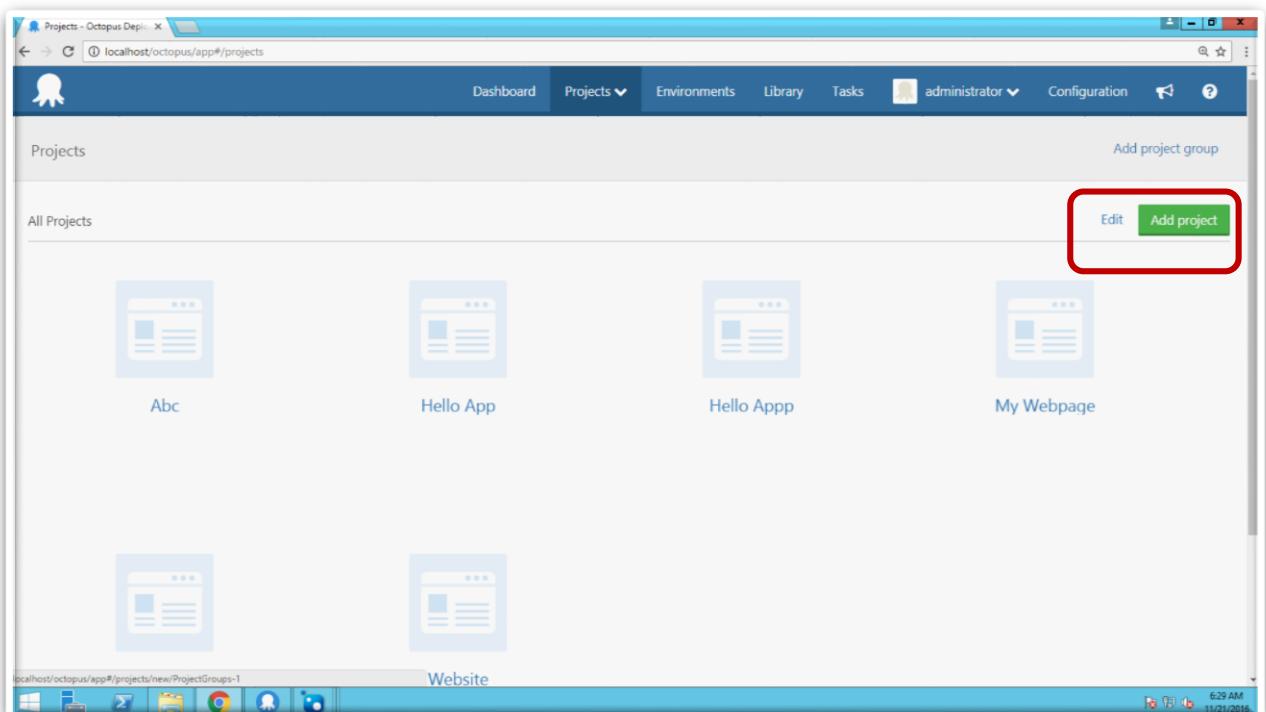
You can see a package named **DemoPackage** is added to the feeds.



The screenshot shows the 'Test Feed - Octopus Deploy' window. On the left, there's a sidebar with links: Packages, External feeds (which is selected and highlighted in blue), Lifecycles, Script modules, Step templates, and Variable sets. The main area is titled 'Test DemoPackage' and contains a search bar with placeholder text 'A package name to search for. Note that names on local folders and file shares are case-sensitive.' Below the search bar is a table titled 'Matching packages'. The table has columns for 'Id', 'Version', and 'Description'. The rows list various packages, with 'DemoPackage' being the last one listed and highlighted with a red rectangular border around its entire row.

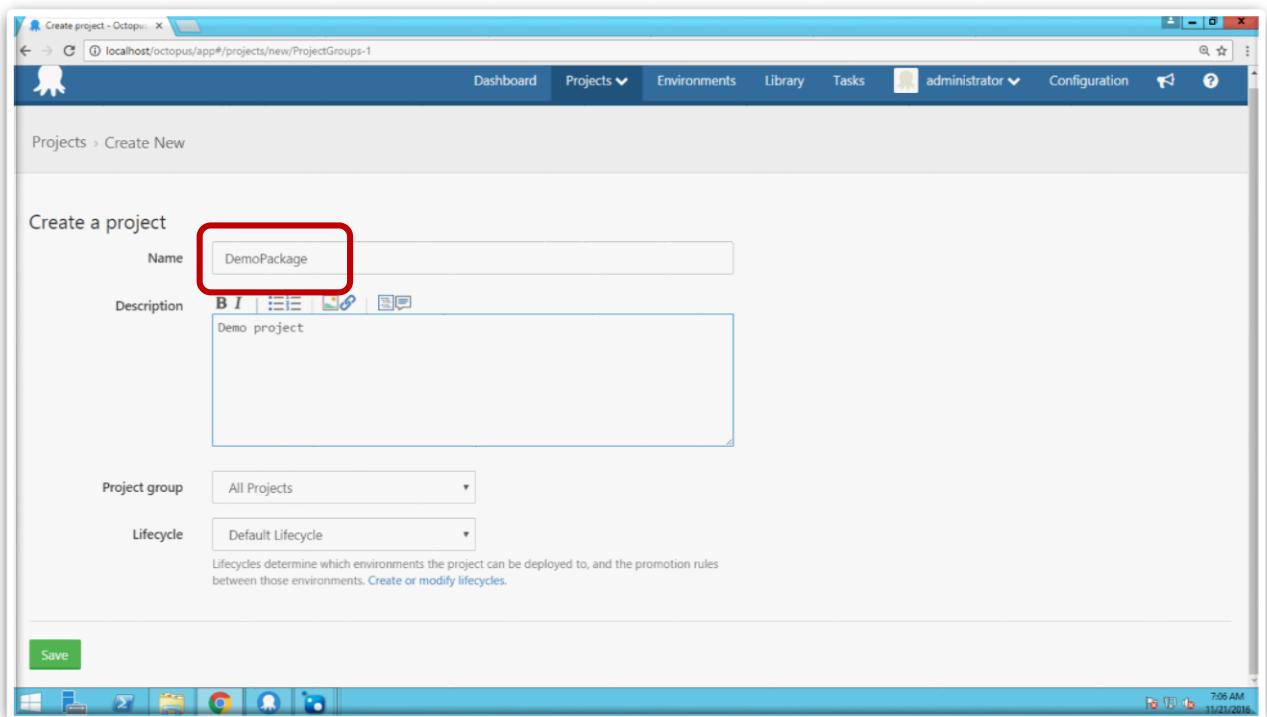
Id	Version	Description
MyWebPage	2.0.0	My package description.
MyPackage	1.0.2	My package description.
HelloApp	1.0.1	My package description.
Demo	1.0.0	My package description.
DemoPackage	1.0.0	My package description.
HelloApp	1.0.0	My package description.
HelloAppp	1.0.0	My package description.
MyPackage	1.0.0	My package description.
Test	1.0.0	My package description.

We will then add a project. Click on **Project > All > Add project**.

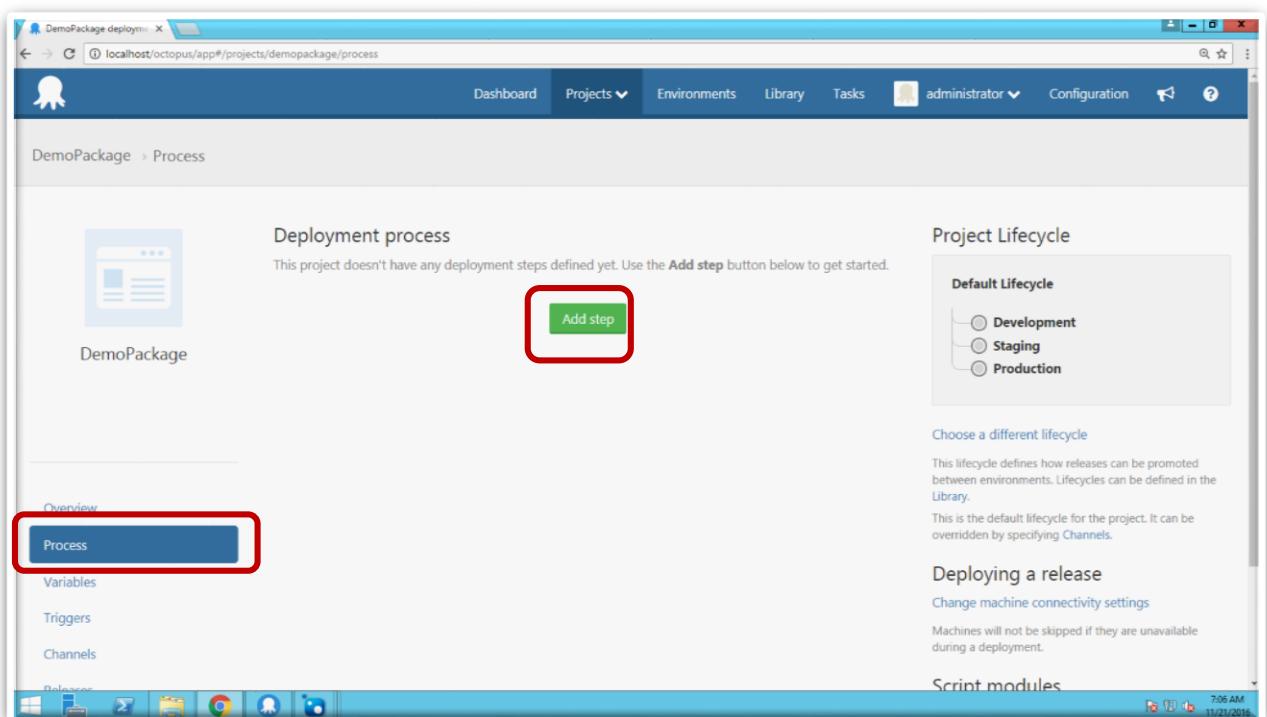


The screenshot shows the 'Projects - Octopus Deploy' window. At the top, there's a navigation bar with links: Dashboard, Projects (selected and highlighted in blue), Environments, Library, Tasks, administrator, Configuration, and a help icon. Below the navigation bar is a section titled 'Projects' with a sub-section 'All Projects'. In this section, there are four items: 'Abc' (with a small icon), 'Hello App' (with a small icon), 'Hello Appp' (with a small icon), and 'My Webpage' (with a small icon). To the right of these items is a button labeled 'Edit' and another button labeled 'Add project' (which is highlighted with a red rectangular border). Further down the page, there are two more project icons: 'Website' (with a small icon) and another unnamed project (with a small icon).

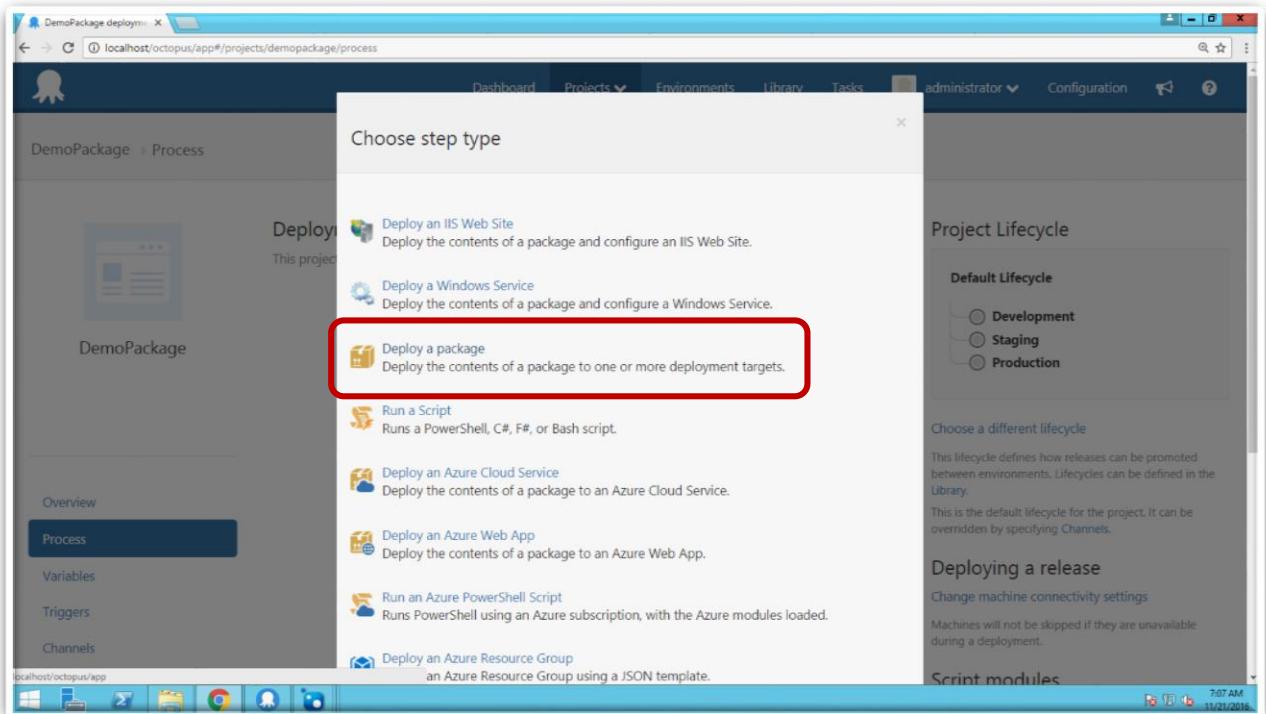
Give a name to your project (**DemoPackage**) and write a description about it. Save it.



Now click on **Process > Add step**. This will configure your project as per your requirement and you have to manually add the steps.

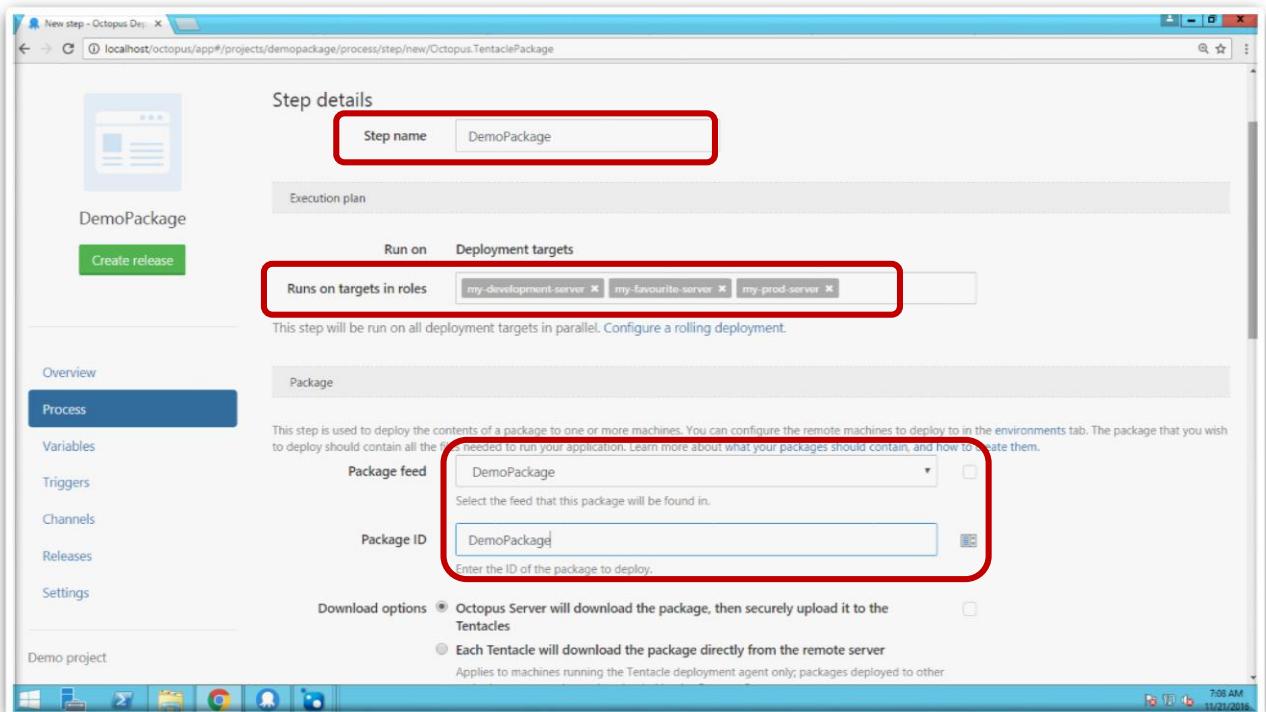


We have a NuGet package to deploy. So here we will select: **Deploy a package**.

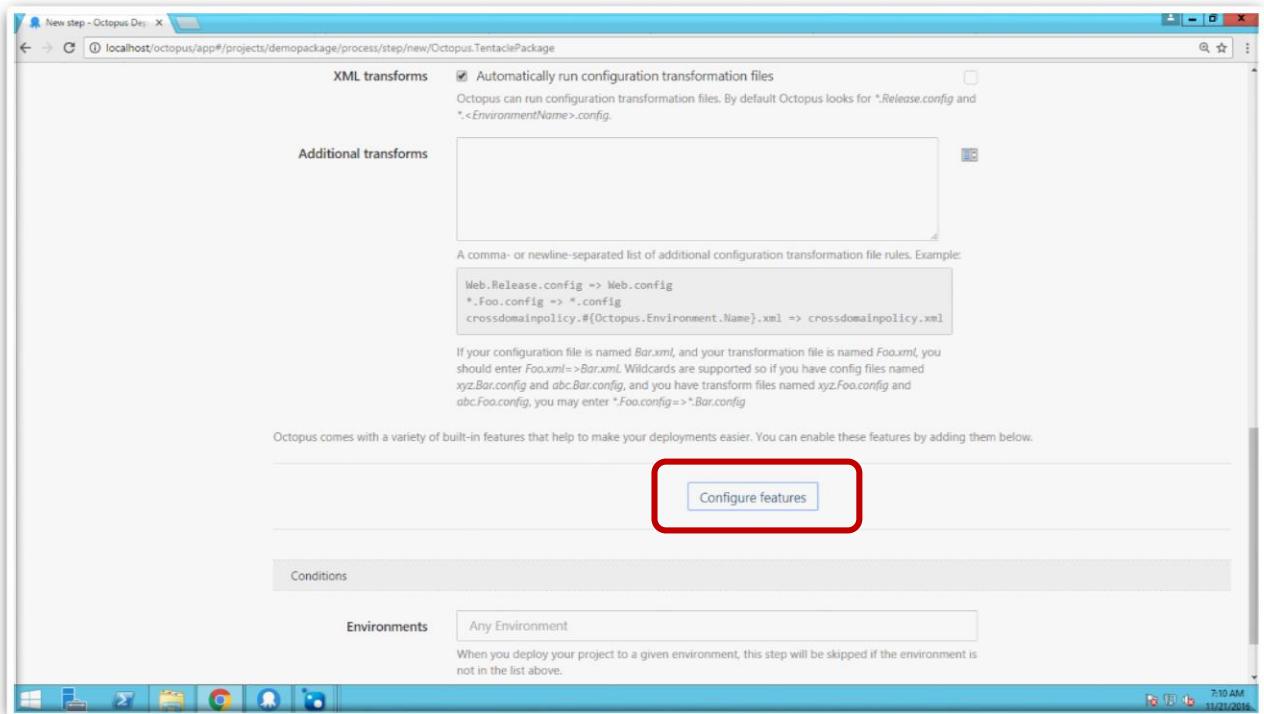


Mention the step name (**DemoPackage**) and the roles (**my-development-server**, **my-favourite-server**, **my-prod-server**) where you want the package to be deployed. Then, add the Package feed (**DemoPackage**) and Package ID

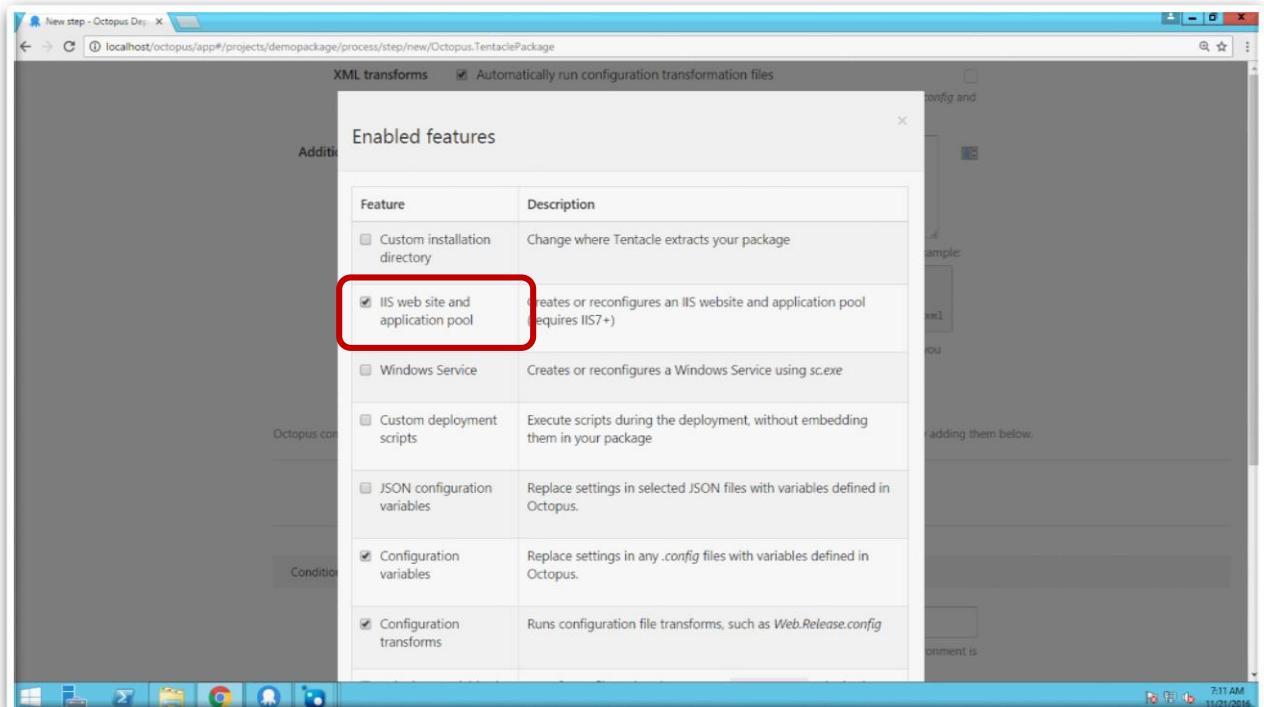
Note→ Detailed explanation about Roles and Targets is available on another document about the basics and installation of Octopus Deploy.



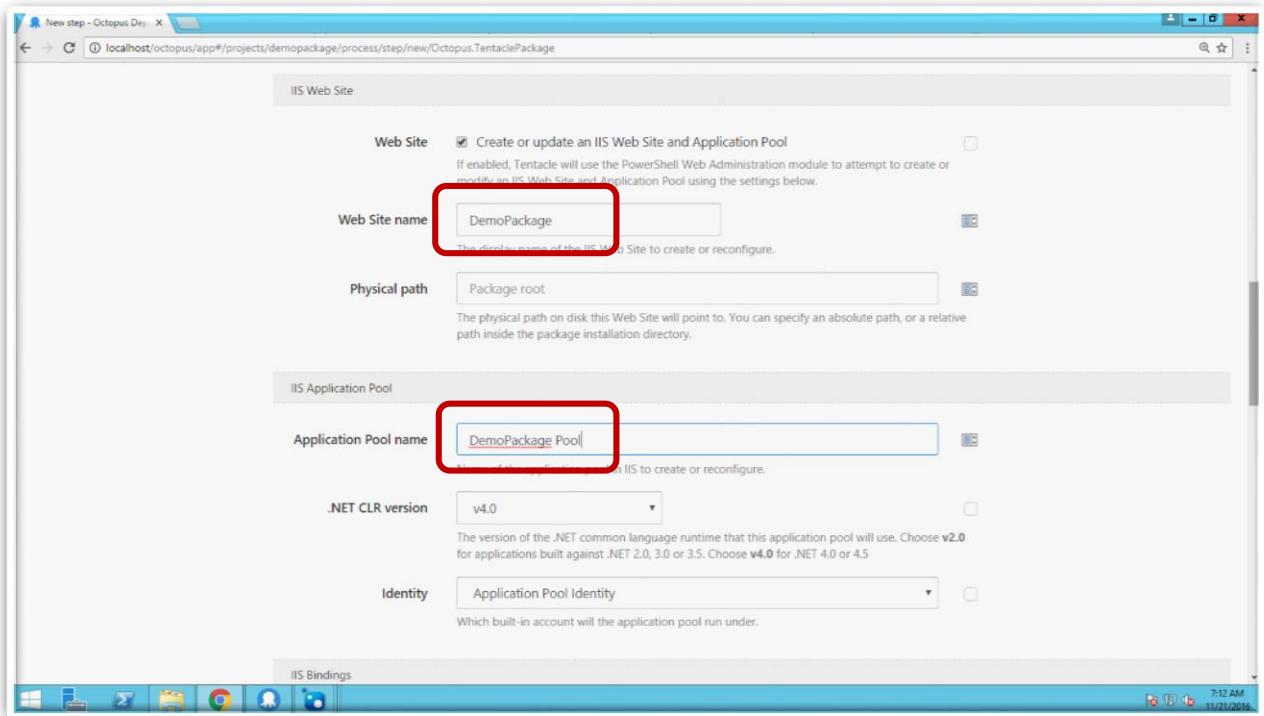
We are going to deploy an IIS website. So, we have to add an IIS website feature to the process.



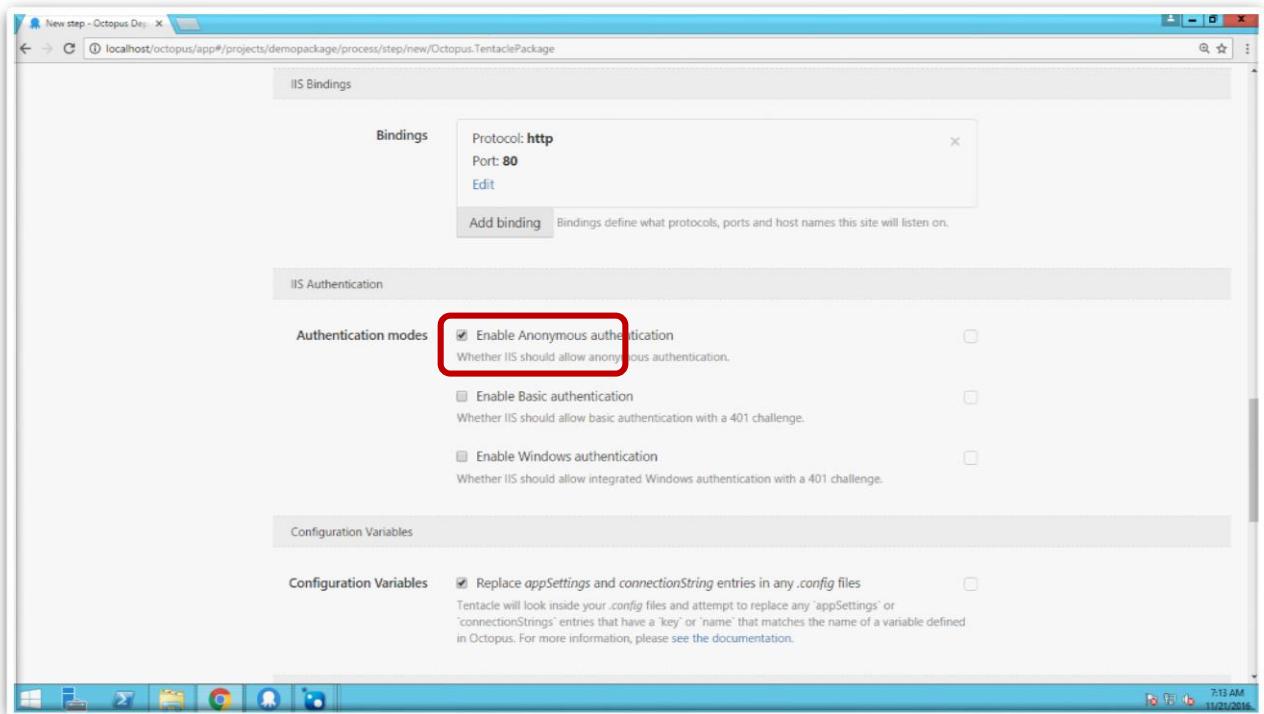
Here, Click > IIS web site and application pool.



We have to give a Web Site name (**DemoPackage**) and an application pool name (**DemoPackage Pool**).



We are not giving any authentication so Click > **Enable Anonymous authentication**. Save it.



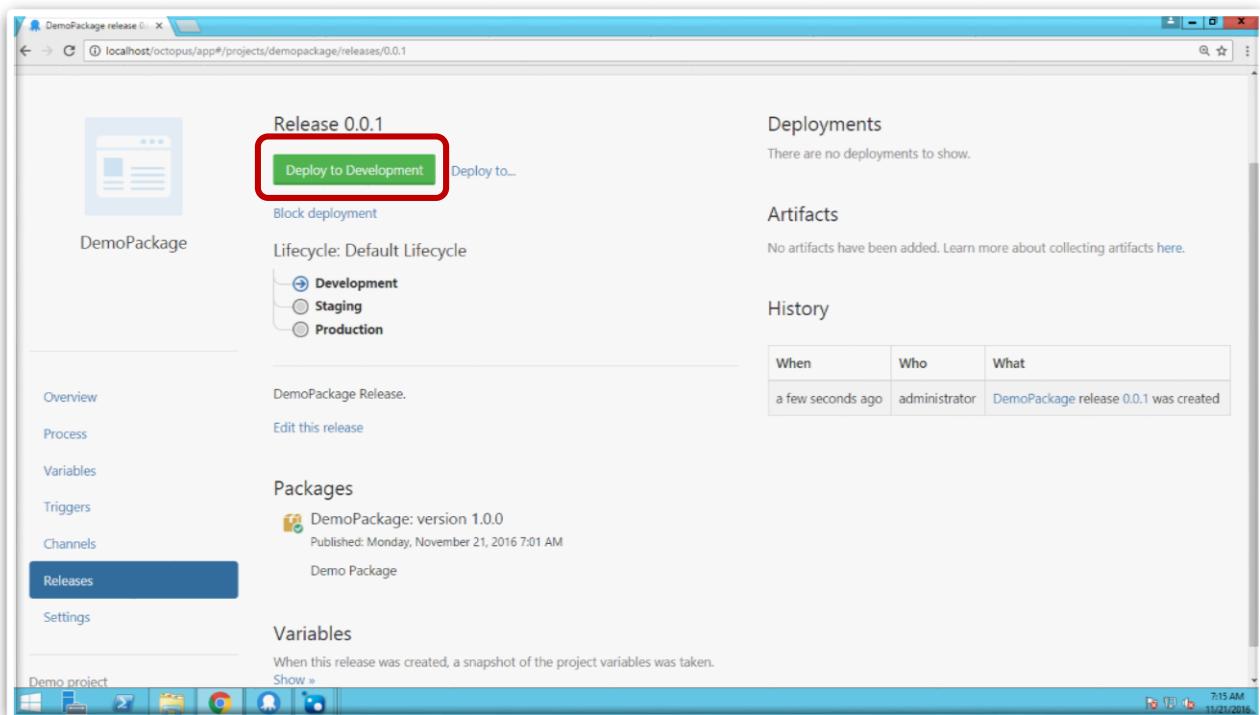
We have to click on **Create Release** to release this deployment to different environments.
 We can see the life cycle: Development > Staging > Production.

The screenshot shows the Octopus Deploy interface for a project named 'DemoPackage'. On the left, there's a sidebar with tabs: Overview, Process (which is selected), Variables, Triggers, Channels, Releases, and Settings. Below the sidebar is a section for 'Demo project' with icons for Windows, IIS, File, and others. The main area is titled 'Deployment process' and shows a step named '1. DemoPackage' which deploys 'DemoPackage' from 'DemoPackage' to machines in roles: 'my-development-server', 'my-favourite-server', and 'my-prod-server'. There's a green 'Add step' button. To the right, there's a 'Project Lifecycle' panel showing a 'Default Lifecycle' with 'Development', 'Staging', and 'Production' stages. Below it, there's a note about choosing a different lifecycle and another section titled 'Deploying a release' with machine connectivity settings. At the bottom right, there's a status bar showing '7:13 AM 11/21/2016'.

Here we can give a customized version number (**0.0.1**) to version the deployment.

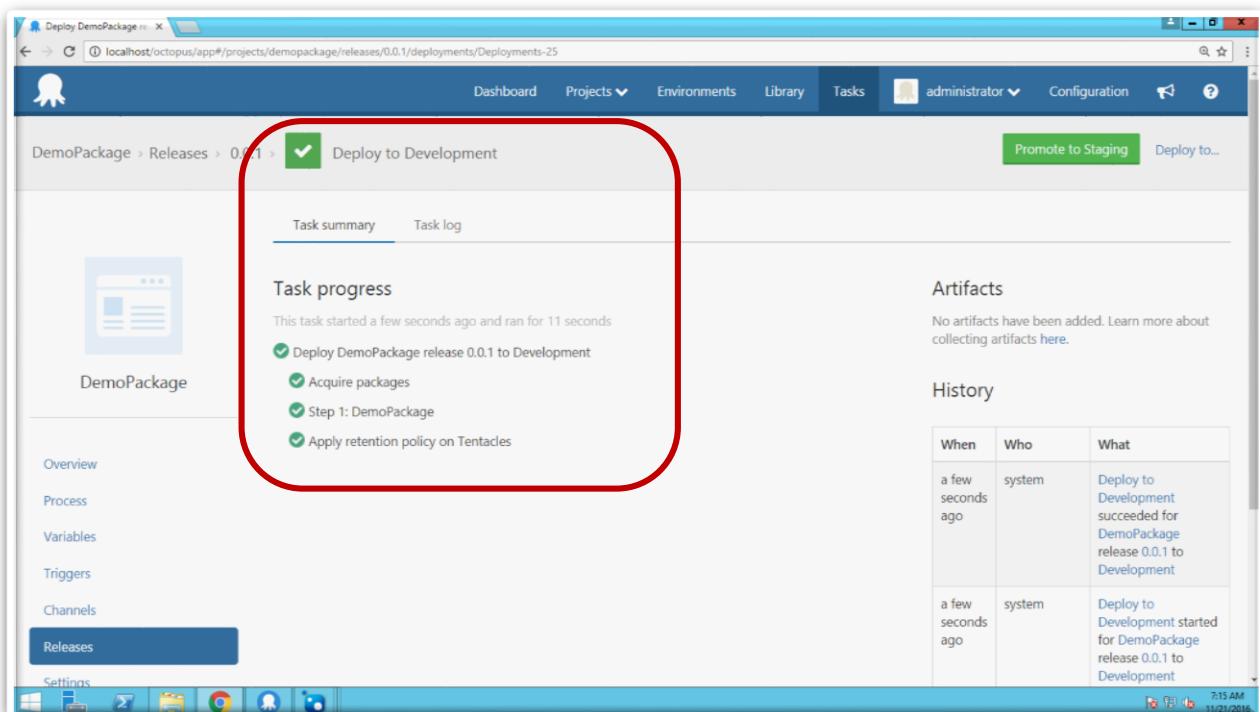
This screenshot shows the 'Create release' dialog for 'DemoPackage'. The sidebar on the left has tabs: Overview, Process, Variables, Triggers, Channels, Releases (which is selected), and Settings. The main area has a 'Create release' title and a 'Version' input field containing '0.0.1' (which is highlighted with a red box). Below it is a note: 'Enter a unique version number for this release with at least two parts. See examples.' There's also a 'Packages' table with one row for 'DemoPackage' set to 'Latest'. The 'Release notes' section contains the text 'DemoPackage Release.' and a summary input field. At the bottom right is a 'Save' button. The status bar at the bottom right shows '7:14 AM 11/21/2016'.

First of all, it will deploy to Development. Click > Deploy to Development.



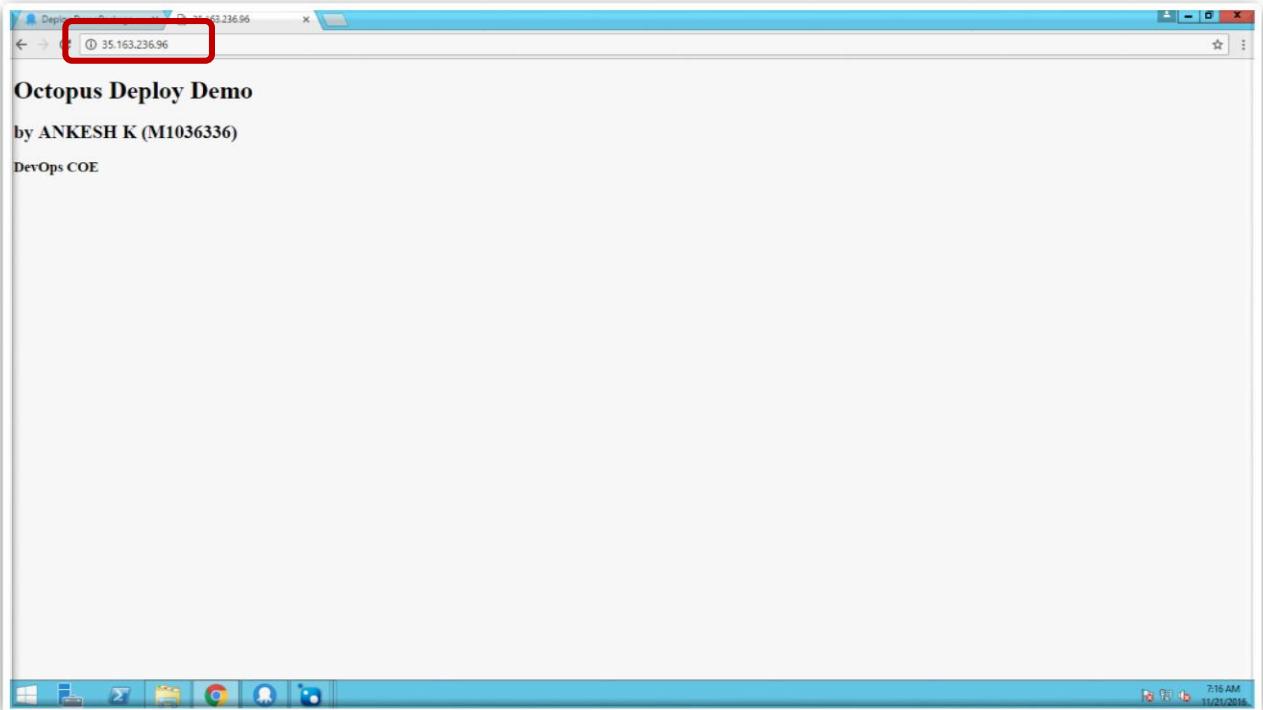
The screenshot shows the Octopus Deploy web interface. On the left, there's a sidebar with links like Overview, Process, Variables, Triggers, Channels, Releases (which is selected and highlighted in blue), and Settings. The main area has a title 'Release 0.0.1' and a 'Deploy to Development' button, which is circled in red. Below this, there's a 'Lifecycle' section with 'Development' selected. To the right, there are sections for 'Deployments' (empty), 'Artifacts' (empty), and 'History' (empty). A table in the 'History' section shows a single entry: 'a few seconds ago' by 'administrator' with the message 'DemoPackage release 0.0.1 was created'. The bottom of the screen shows a Windows taskbar with various icons.

Now it is successfully deployed to the development.



This screenshot shows the deployment details for 'DemoPackage' release 0.0.1 to Development. The 'Task progress' section is highlighted with a red box. It lists four tasks: 'Deploy DemoPackage release 0.0.1 to Development', 'Acquire packages', 'Step 1: DemoPackage', and 'Apply retention policy on Tentacles', all of which have been completed successfully (indicated by green checkmarks). To the right, there are sections for 'Artifacts' (empty) and 'History'. The 'History' table shows two entries: one for the deployment itself and another for the start of the deployment process. The bottom of the screen shows a Windows taskbar.

We will put the Development server IP in our browser and we can see our webpage deployed in that server.



Once the package is deployed in the development server. It will ask our permission to deploy it to the staging environment. Click > **Promote to Staging**.

A screenshot of the Octopus Deploy application interface. The top navigation bar includes 'Dashboard', 'Projects', 'Environments', 'Library', 'Tasks', 'administrator', 'Configuration', and a help icon. The main content area shows a deployment task for 'DemoPackage' release '0.0.1' to 'Development'. The 'Task summary' tab is selected, showing a 'Task progress' section with a list of successful steps: 'Deploy DemoPackage release 0.0.1 to Development', 'Acquire packages', 'Step 1: DemoPackage', and 'Apply retention policy on Tentacles'. To the right, there are sections for 'Artifacts' (no artifacts added), 'History' (two log entries), and 'Deploy to...' (a green button). A red box highlights the 'Promote to Staging' button in the top right of the main content area. The bottom of the screen shows the Windows taskbar with pinned icons.

We will promote that package to the staging environment. And it will show success once the deployment is done.

The screenshot shows the Octopus Deploy interface for a release named 'DemoPackage' version 0.0.1. A red box highlights the 'Task progress' section, which lists the following steps:

- Deploy DemoPackage release 0.0.1 to Staging
- Acquire packages
- Step 1: DemoPackage
- Apply retention policy on Tentacles

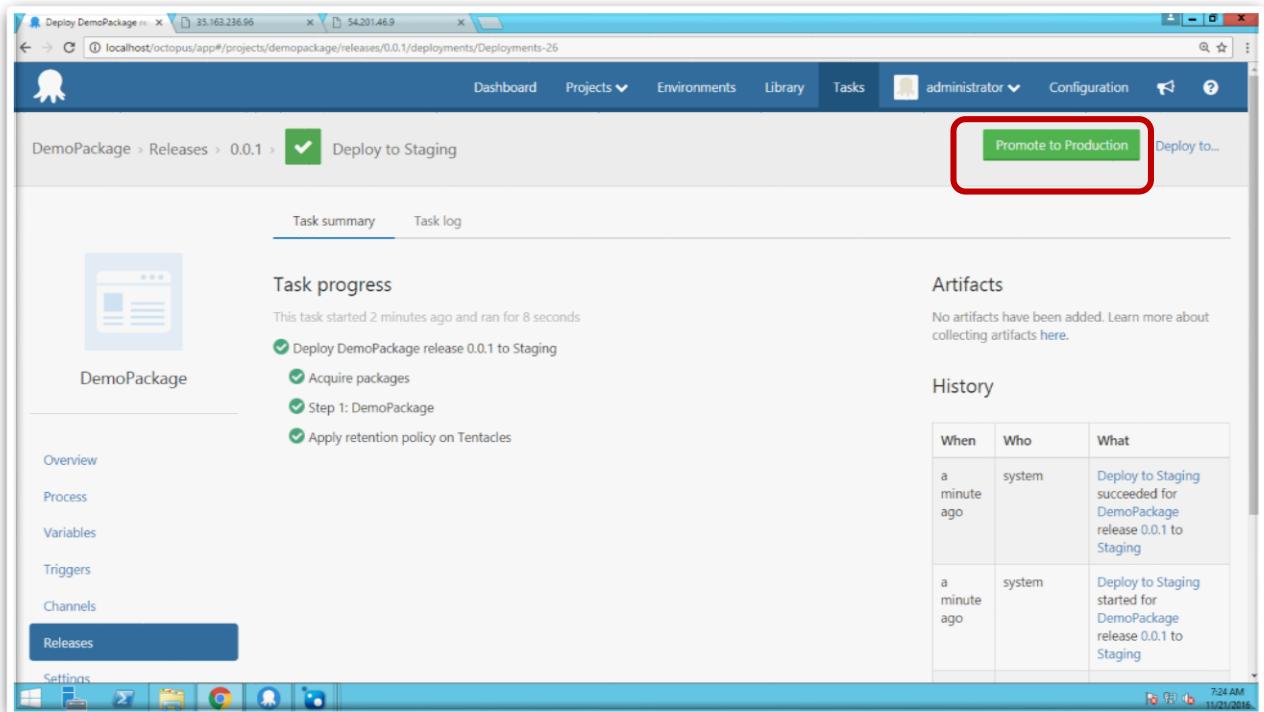
The 'Task log' tab is visible below the task progress. To the right, there are sections for 'Artifacts' (none) and 'History' (two entries: 'Deploy to Staging succeeded' and 'Deploy to Staging started').

We can put our Staging Machine IP to check our website deployed in staging environment.

The screenshot shows a web browser window displaying a website titled 'Octopus Deploy Demo' with the URL '54.201.46.9'. A red box highlights the URL bar. The page content includes the title, author ('by ANKESH K (M1036336)'), and a footer ('DevOps COE').

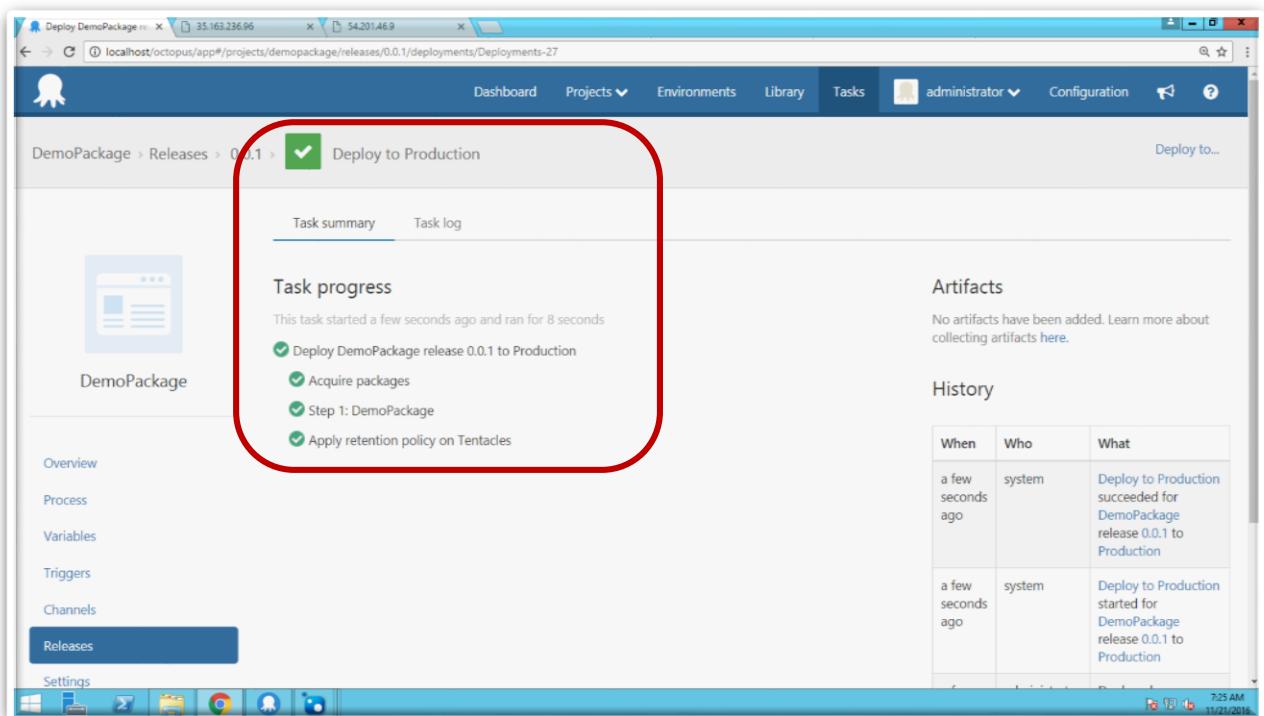
Now Octopus Deploy will ask our permission to deploy the package to production environment. If everything is correct in staging, we will promote our package to the production environment.

Click > **Promote to Production**.



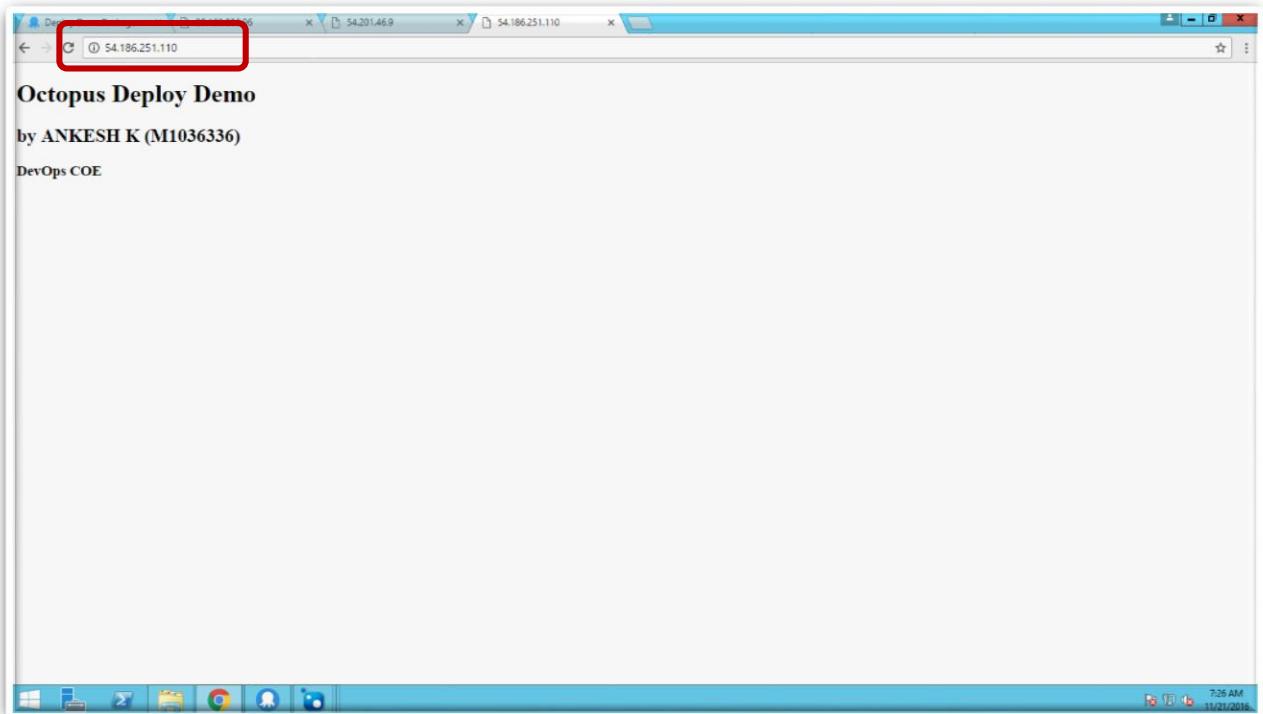
The screenshot shows the Octopus Deploy interface for a project named 'DemoPackage'. The 'Releases' tab is selected. Under the '0.0.1' release, the 'Deploy to Staging' step has been completed successfully, indicated by a green checkmark icon. A red box highlights the 'Promote to Production' button, which is also associated with the 'Deploy to...' link. The 'Task summary' section shows the task started 2 minutes ago and ran for 8 seconds, with all steps completed successfully. The 'Artifacts' section indicates no artifacts have been added. The 'History' section shows two log entries: one for the deployment to Staging and another for the promotion attempt. The system timestamp in the bottom right corner is 7:24 AM on 11/21/2016.

Now it is successfully deployed to the production environment.



The screenshot shows the Octopus Deploy interface for the same project 'DemoPackage'. The 'Releases' tab is selected, and the 'Deploy to Production' step has been completed successfully, indicated by a green checkmark icon. A red box highlights the 'Task progress' section, which shows the task started a few seconds ago and ran for 8 seconds, with all steps completed successfully. The 'Artifacts' section indicates no artifacts have been added. The 'History' section shows two log entries: one for the deployment to Production and another for the promotion attempt. The system timestamp in the bottom right corner is 7:25 AM on 11/21/2016.

We can check that by putting the Production machine IP in our browser.

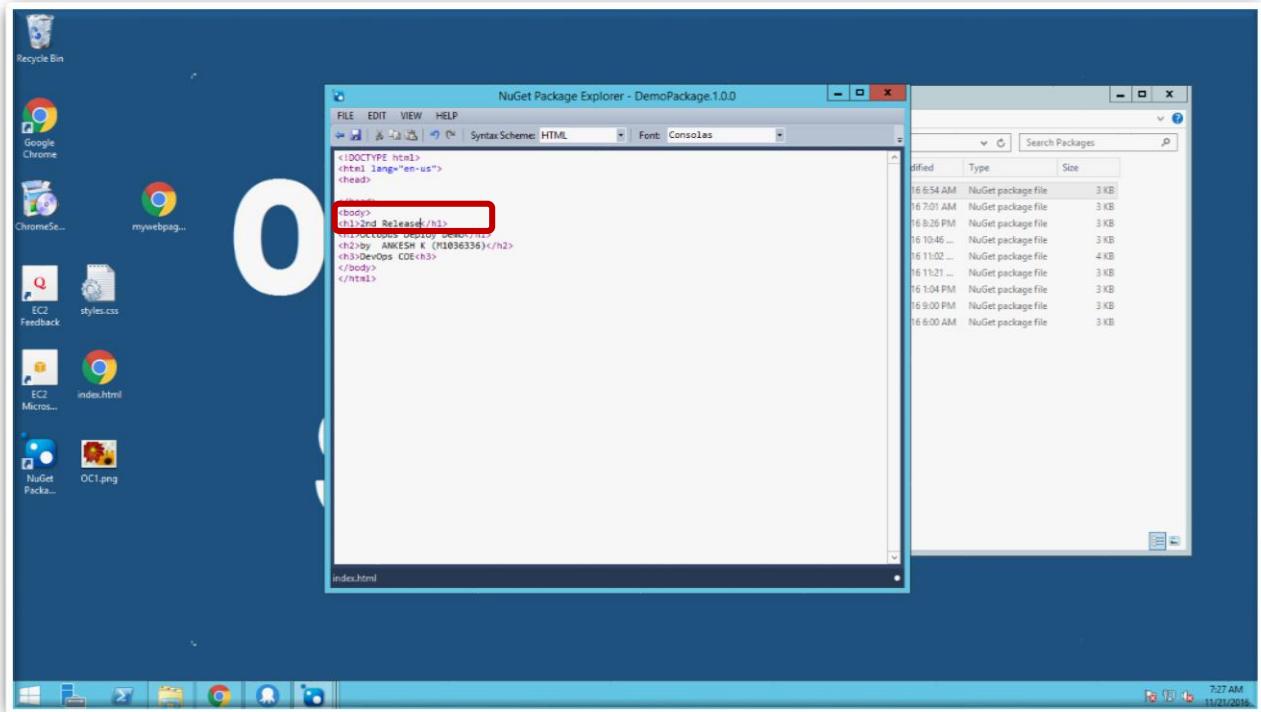


So this is the workflow of the Octopus Server:

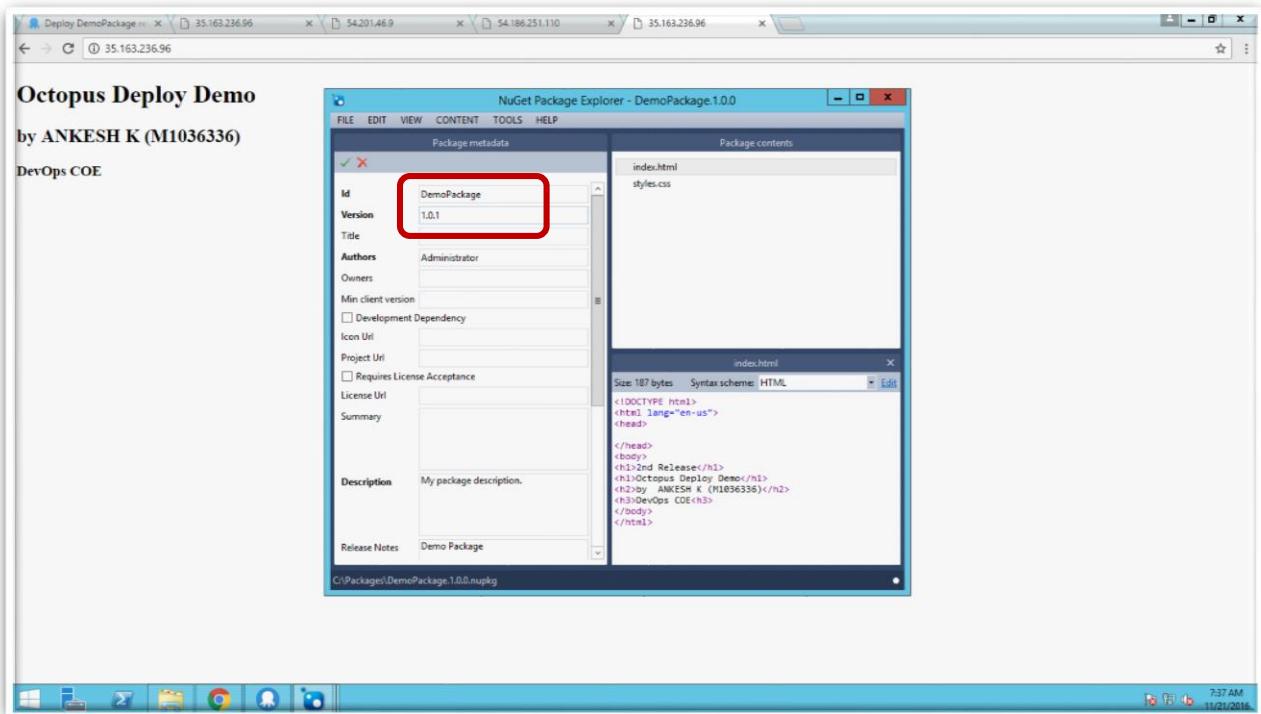
1. Add machines in different environments as tentacles to the Octopus Server.
2. Assign machines with different roles and targets.
3. Create packages using NuGet Package Explorer.
4. Configure the packages in the Octopus Server.
5. Create a project and configure steps of deployment.
6. Deploy packages in different environments as per our requirement.
7. Versioning the deployment.

Now we will see how can we version our deployment →

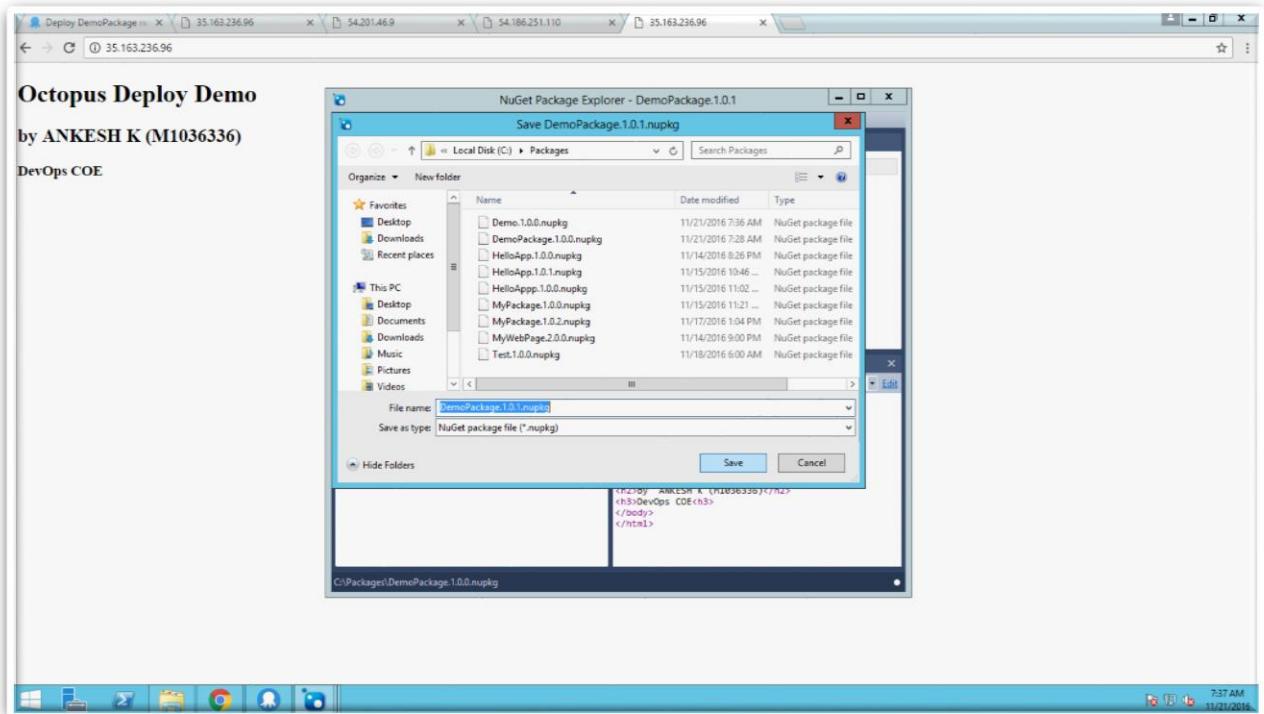
We will make some changes in our source code.



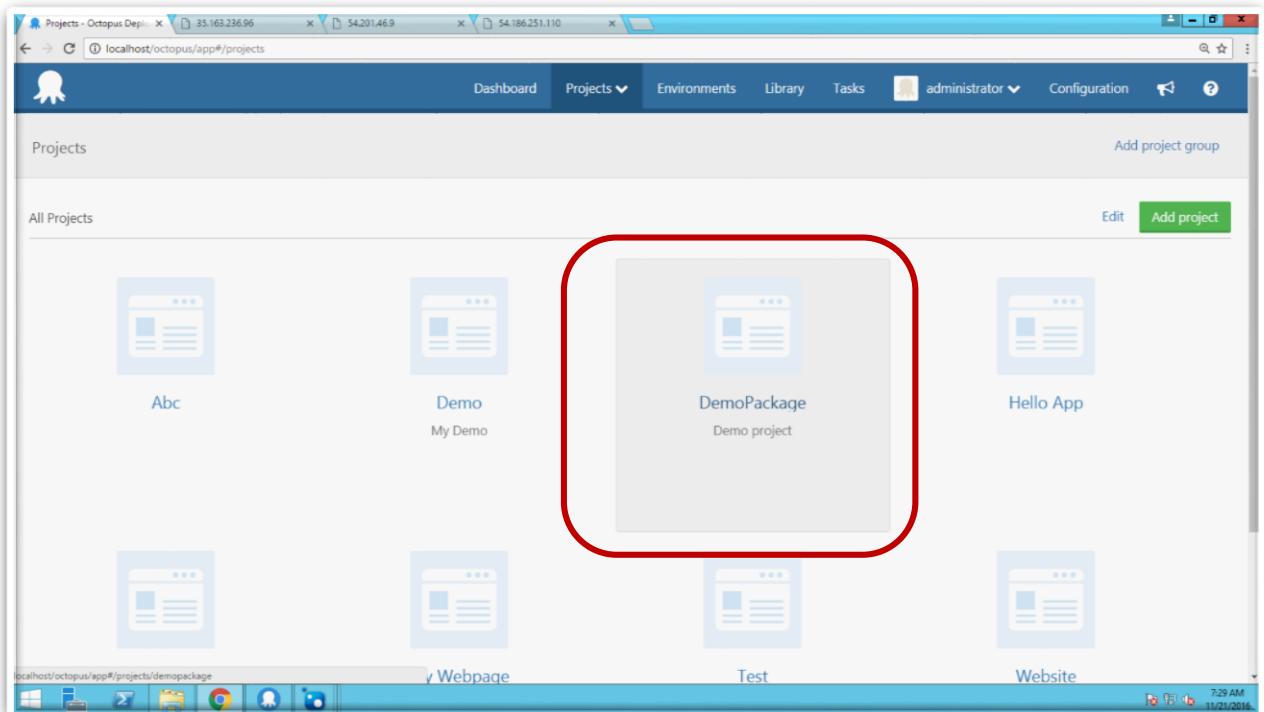
Save it with a different version number (**1.0.1**) in NuGet package explorer.



Save the package in the same folder where other packages are stored.



Now we will go inside our project **DemoPackage**.



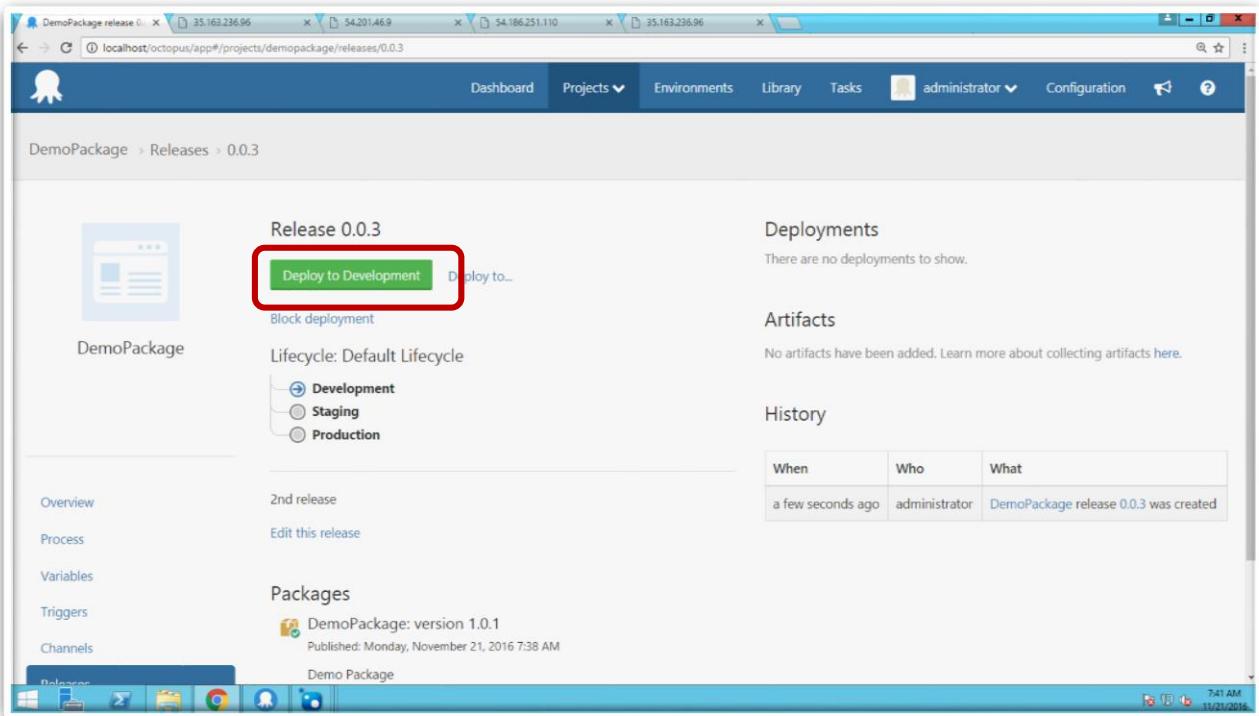
This is the project dashboard. Click > **Create release**.

The screenshot shows the Octopus Project Dashboard for 'DemoPackage'. On the left, there's a sidebar with 'Overview', 'Process', 'Variables', 'Triggers', and 'Channels'. The main area has tabs for 'Release', 'Development', 'Staging', and 'Production'. Under the 'Release' tab, there's a table showing releases 0.0.2 and 0.0.1 across environments. A green button labeled 'Create release' is highlighted with a red box. The URL in the browser is localhost/octopus/app#/projects/demopackage. The system status bar at the bottom right shows '7:38 AM 11/21/2016'.

Check out the latest DemoPackage version (**1.0.1**) and save it.

The screenshot shows the 'Create release' dialog for 'DemoPackage'. It has fields for 'Version' (0.0.3) and 'Latest' (radio button selected). Below is a table for 'Packages' with one row for 'DemoPackage'. The 'Latest' column is highlighted with a red box. The URL in the browser is localhost/octopus/app#/projects/demopackage/releases/create. The system status bar at the bottom right shows '7:40 AM 11/21/2016'.

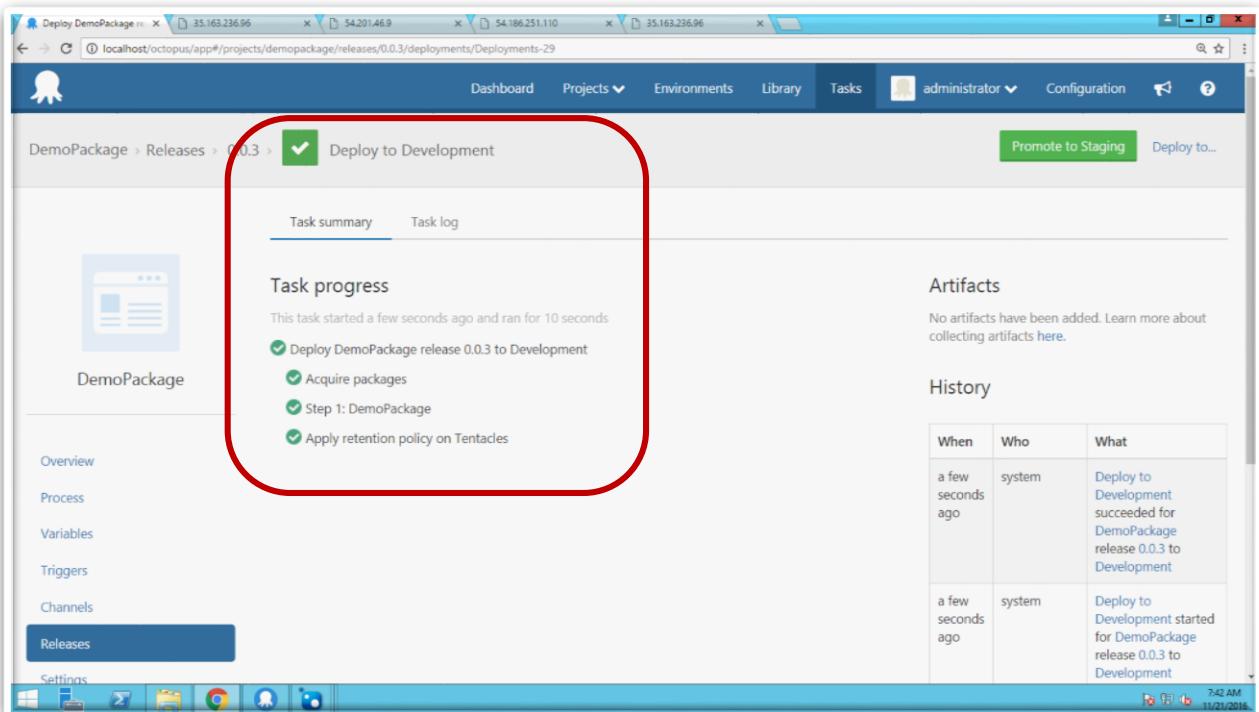
Now check the release version (**0.0.3**) and deploy it to Development environment.



The screenshot shows the Octopus Deploy interface. On the left, there's a sidebar with links like Overview, Process, Variables, Triggers, Channels, and Releases (which is currently selected). The main area displays 'Release 0.0.3' for 'DemoPackage'. It includes a 'Lifecycle' section with 'Development' selected. A prominent green button labeled 'Deploy to Development' is highlighted with a red box. To the right, sections for 'Deployments' (empty), 'Artifacts' (empty), and 'History' (empty) are shown. At the bottom, a table lists the creation of the release by 'administrator'.

When	Who	What
a few seconds ago	administrator	DemoPackage release 0.0.3 was created

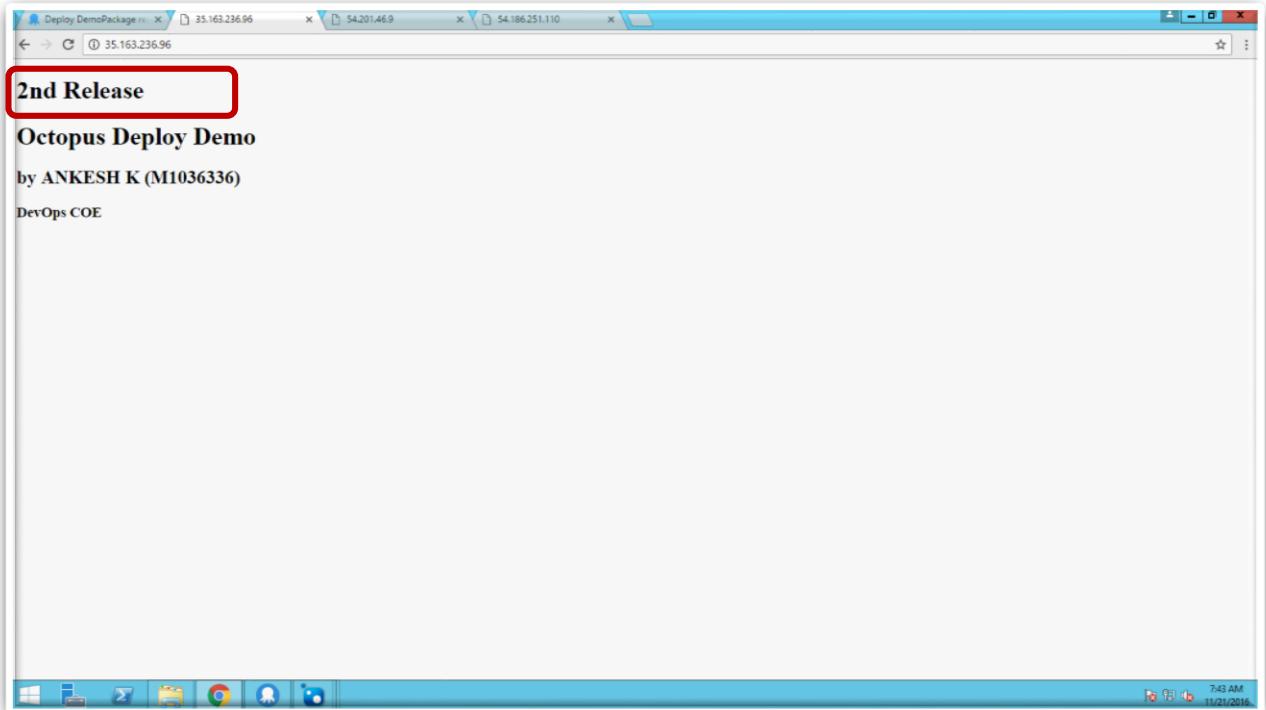
Once it is successful, it will ask you to promote the package to the staging environment.



This screenshot shows the 'Task summary' tab for the deployment task. The task has completed successfully, indicated by a green checkmark icon. The task log shows four steps: 'Deploy DemoPackage release 0.0.3 to Development', 'Acquire packages', 'Step 1: DemoPackage', and 'Apply retention policy on Tentacles'. To the right, sections for 'Artifacts' (empty) and 'History' (empty) are visible. The history table shows two entries: one for the deployment to Development and another for the start of the deployment.

When	Who	What
a few seconds ago	system	Deploy to Development succeeded for DemoPackage release 0.0.3 to Development
a few seconds ago	system	Deploy to Development started for DemoPackage release 0.0.3 to Development

We can check our latest website in our development server and we can see the changes made in the second release.



This is our **project dashboard** where we can see the releases in different environment with their release numbers.

The screenshot shows the Octopus Deploy project dashboard for the 'DemoPackage' project. The dashboard has a navigation bar with 'Dashboard', 'Projects', 'Environments', 'Library', 'Tasks', 'administrator', and 'Configuration'. On the left, there's a sidebar with 'DemoPackage' and a 'Create release' button. The main area shows a table of releases across three environments: Development, Staging, and Production. The table has columns for 'Release', 'Development', 'Staging', and 'Production'. There are three rows: one for release 0.0.3 (Development), one for release 0.0.2 (Staging), and one for release 0.0.1 (Production). The release 0.0.1 row in the Production column is highlighted with a red box. The status bar at the bottom shows '7:44 AM 11/21/2016'.

This is Octopus Deploy Dashboard. Here we can see which release version is running in which environment for all the projects.

The screenshot shows the Octopus Deploy Dashboard interface. At the top, there are three browser tabs: 'Dashboard - Octopus Deploy' (active), '35.163.236.96' (inactive), and '54.201.46.9' (inactive). The main header includes 'Dashboard', 'Projects', 'Environments', 'Library', 'Tasks', 'administrator', and 'Configuration'. Below the header, the page title is 'Dashboard' and there is a 'Configure dashboard' link. The main content area is titled 'All Projects' and shows releases for four projects: 'Abc', 'Demo', 'DemoPackage', and 'Hello App'. Each project has columns for 'Development', 'Staging', and 'Production'. The 'DemoPackage' row is highlighted with a red box. The data is as follows:

Project	Development	Staging	Production
Abc	✓ 0.0.1 November 17th 2016	✓ 0.0.1 November 17th 2016	✓ 0.0.1 November 17th 2016
Demo	○ 0.0.2 November 21st 2016		
DemoPackage	✓ 0.0.3 November 21st 2016	✓ 0.0.1 November 21st 2016	✓ 0.0.1 November 21st 2016
Hello App		✓ 0.0.1 November 14th 2016	
Hello App	✓ 0.0.4 November 15th 2016	✓ 0.0.4 November 15th 2016	✓ 0.0.4 November 15th 2016
My Webpage		✓ 0.0.1 November 14th 2016	

The bottom of the screen shows a Windows taskbar with icons for File Explorer, Task View, File, Mail, and Start. The system tray shows the date and time as '7:54 AM 11/21/2016'.

---END---