## Lab 11-01: Working with ARM Templates

### Service Introduction

PayFast, a rising FinTech startup offering secure online payments, needed to quickly deploy and manage its cloud infrastructure but faced complexity and manual work with traditional configuration methods. They required a solution to automate deployments, enforce consistency, and enable self-service provisioning for developers.

### Problem

PayFast relied on manually configuring resources through the Azure portal, a slow and error-prone process hindering DevOps agility and developer productivity. Manual deployments led to inconsistencies in resource setup, creating environment drift and making infrastructure troubleshooting challenging.

### Solution

PayFast's success story showcases the power of ARM templates for companies seeking to boost DevOps agility, improve developer productivity, and ensure consistency in their Azure cloud infrastructure. By utilizing the power of automation and code-driven infrastructure, PayFast achieved faster deployments, streamlined operations, and empowered their developers to innovate at a rapid pace.

|  |
| --- |
| Task 1: Working with ARM Template  1. Log in to the **Microsoft Azure** portal and go to the portal menu.      1. Go to the options present at the top of the portal. Click on “**Azure Cloud Shell.”**      1. Click on “**Create Storage.”**      1. The Cloud Shell session will start in a moment. 2. Type **“ls**” to see the list of available items.      1. You will see the .json file in the drive with the name **“ips.”**      1. To see the code, open the code editor by typing “ips.json.”      1. You will see the available code in the ips.json file.   C:\Users\ad\Desktop\Azure_Pic\a5.png  C:\Users\ad\Desktop\Azure_Pic\a6.png   1. Press “**CTRL+Q**” to close the code editor. 2. Type the following command in the Cloud Shell session to create a resource group.      1. You will see output in JSON format.      1. After the resource group, we will deploy the resource in that resource group according to the **“ips.json”** code. 2. To do that, run the following sets of commands:      1. This command will take some time to complete the execution. 2. After the execution process, the following output will appear.      1. Go to the Azure Portal page, and open “**Resource groups.”** Click on the resource group that was just created from the CLI session.      1. Go to the options present at the left corner of the page. Click on “**Deployments”** present inside “**Settings.”**      1. Click on the deployment. From the left corner option, click on “**Template.”**      1. You will see that you can get access to the ARM template that was deployed using CLI.      1. To see another important point, go back to the portal home page. Click on **“Resource groups.”** Now, click on another resource group.      1. In this resource group, there exists a virtual machine.      1. From the left corner option, click on “**Deployments”** present inside “**Settings.”**      1. You will see the default template.      1. Go to the Azure home portal page and click “**Template.”**      1. To create a template, click on **“Create.”**      1. Write the template name and fill in the description section. 2. Click on **“OK.”**      1. Now, on the template page, click on “**Refresh.”**      1. You will see that your created template has appeared. Click on it. You will see the “**Publisher**” and “**Modified”** fields.      1. With the template option, you can easily deploy a template. When you click “**deploy,**” you will see the “**Custom deployment,”** which will allow editing the template and parameters. |

## Lab 11-02: Create a Website Hosted in Azure

### Service Introduction

Creating a website hosted in Azure is a straightforward process that leverages Azure App Service, a fully managed platform allowing developers to build, deploy, and scale web applications seamlessly. To initiate the process, users can utilize the Azure portal to create an App Service plan and a corresponding web app. Developers can then deploy their website code, whether it is developed in .NET, Java, Node.js, Python, or other supported languages, using various deployment options like Git, Azure DevOps, or Visual Studio integration. Azure provides features such as automatic scaling, continuous deployment, and integration with Azure DevOps, streamlining the development workflow. Moreover, custom domains and SSL certificates can be easily configured through Azure App Service, ensuring a secure and personalized web presence. With built-in monitoring, logging, and integration with Azure services, hosting a website in Azure offers a robust and scalable solution for delivering web content to users worldwide.

### Problem

An organization has shifted its resources from on-premises to the cloud. The developer wants to create a new website for the organization using the Azure service. How could this be possible?

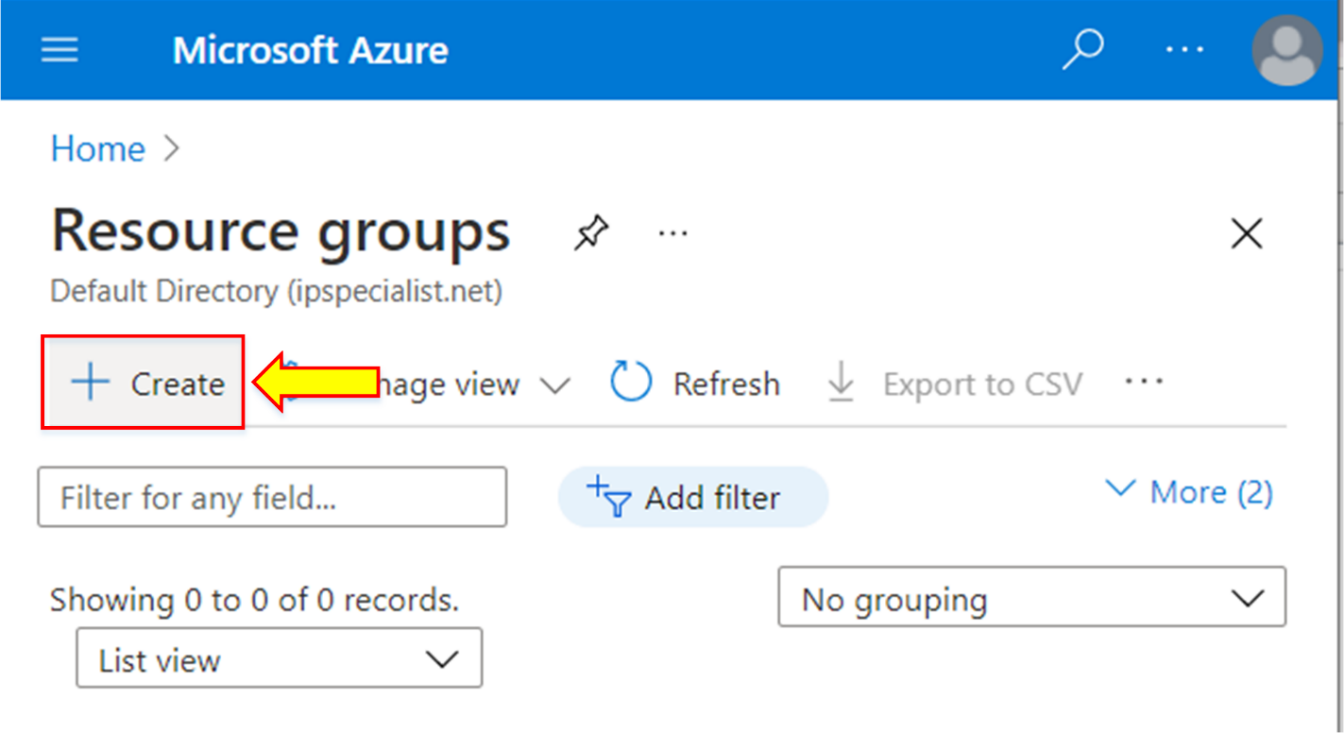
### Solution

Using Azure App Service, the developer can host the website by following the given steps.

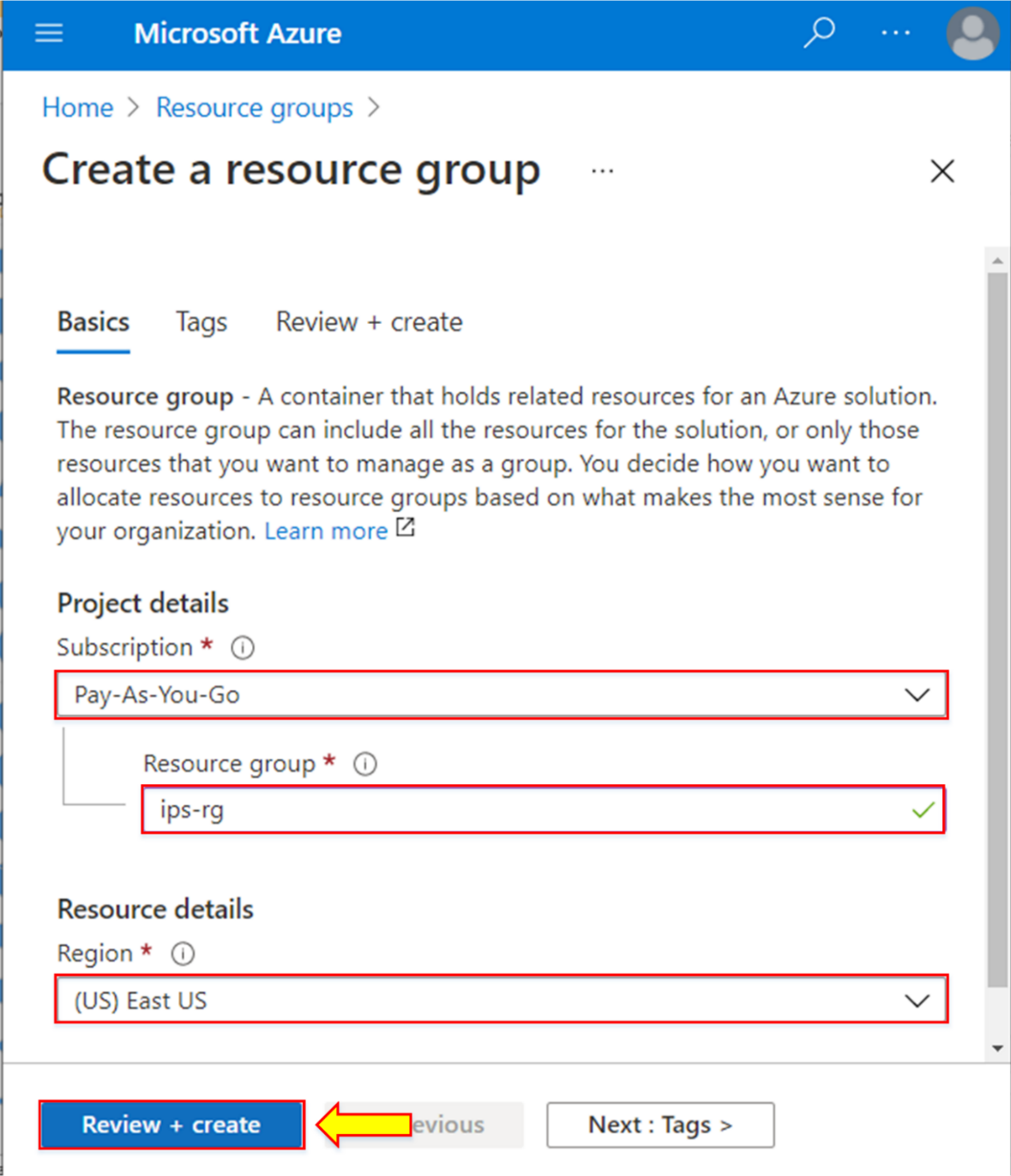
1. Log in to the **Microsoft Azure** portal and go to the portal menu.
2. Before creating a website, we will first create a resource group in which the website will reside. Therefore, click on “**Resource groups**” from the portal home page.



1. Click on **“+ Create.”**



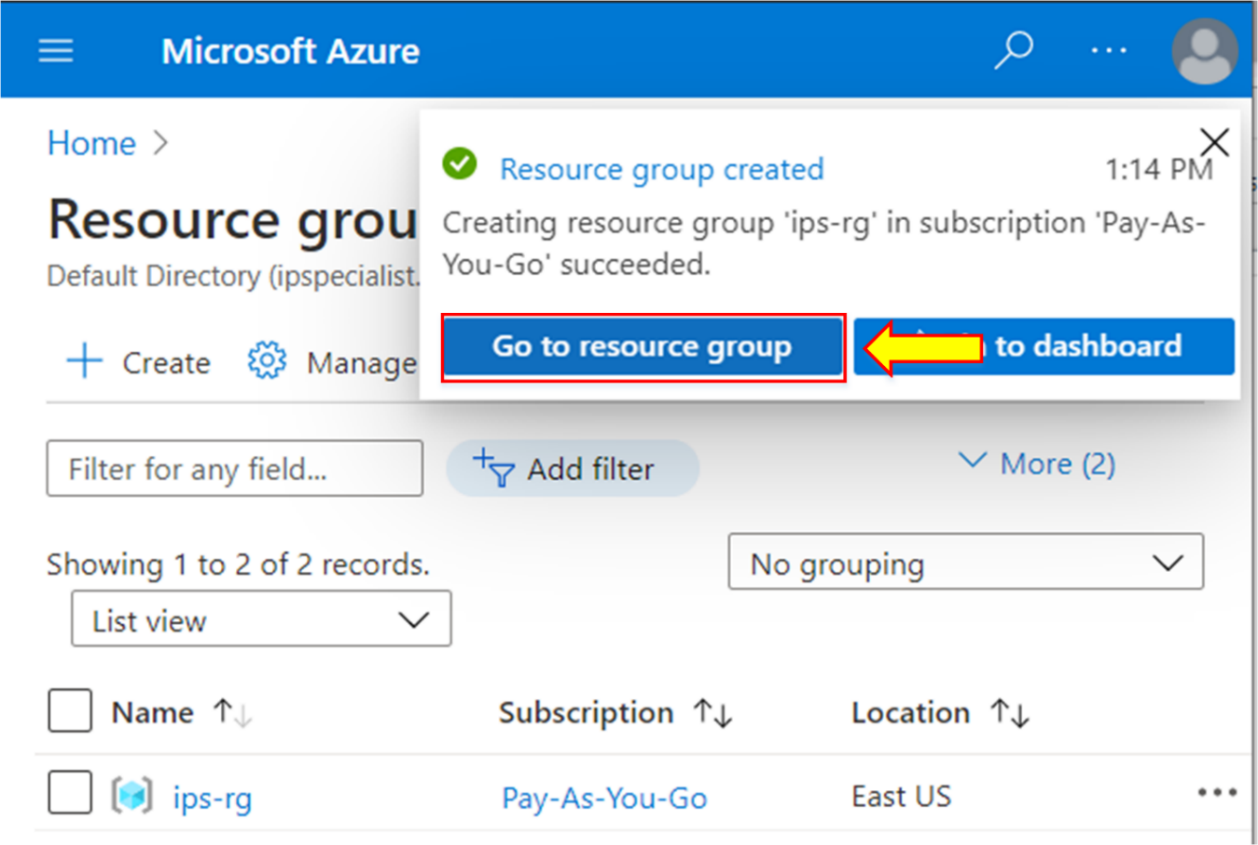
1. Enter the configuration for the resource group. Inside the “**Project details,”** select your subscription.
2. Write the unique name for the resource group.
3. Inside the “**Resource details,”** select your nearest location from the list.
4. Click on **“Review + create.”**



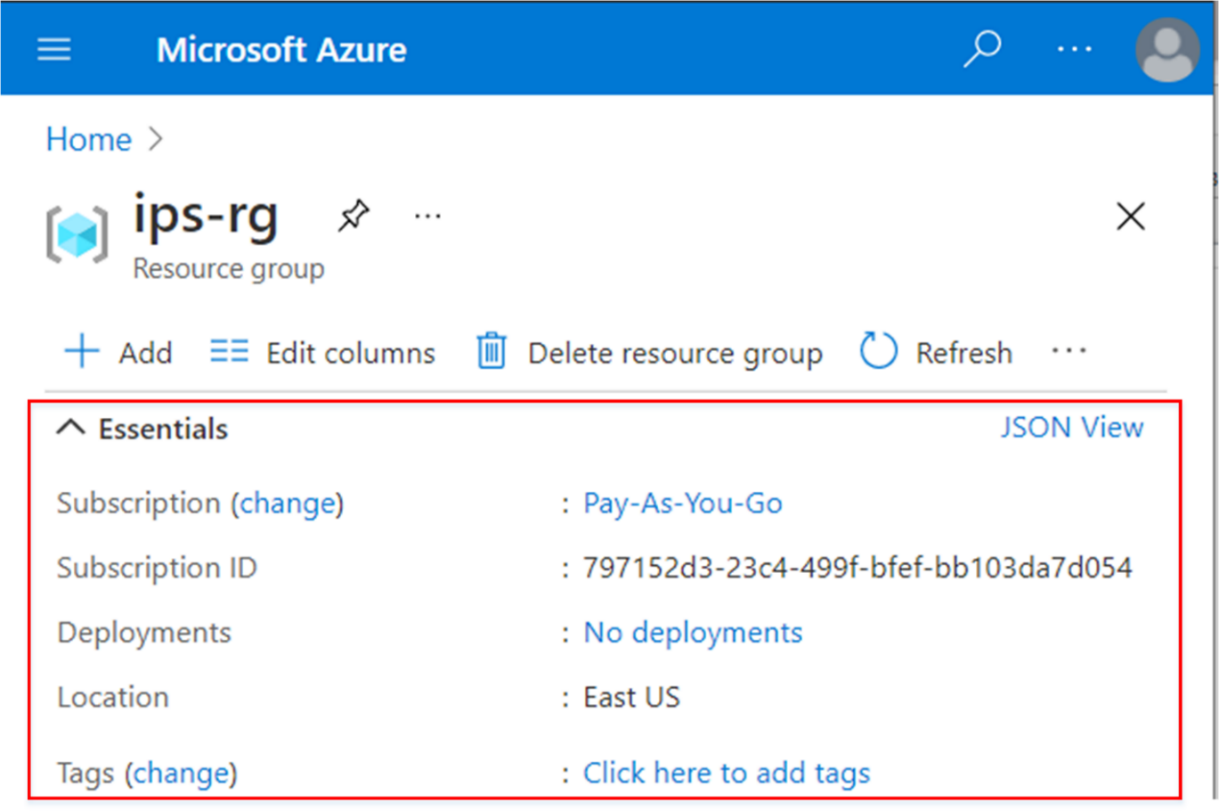
1. Once the validation is passed, click on “**Create.”**



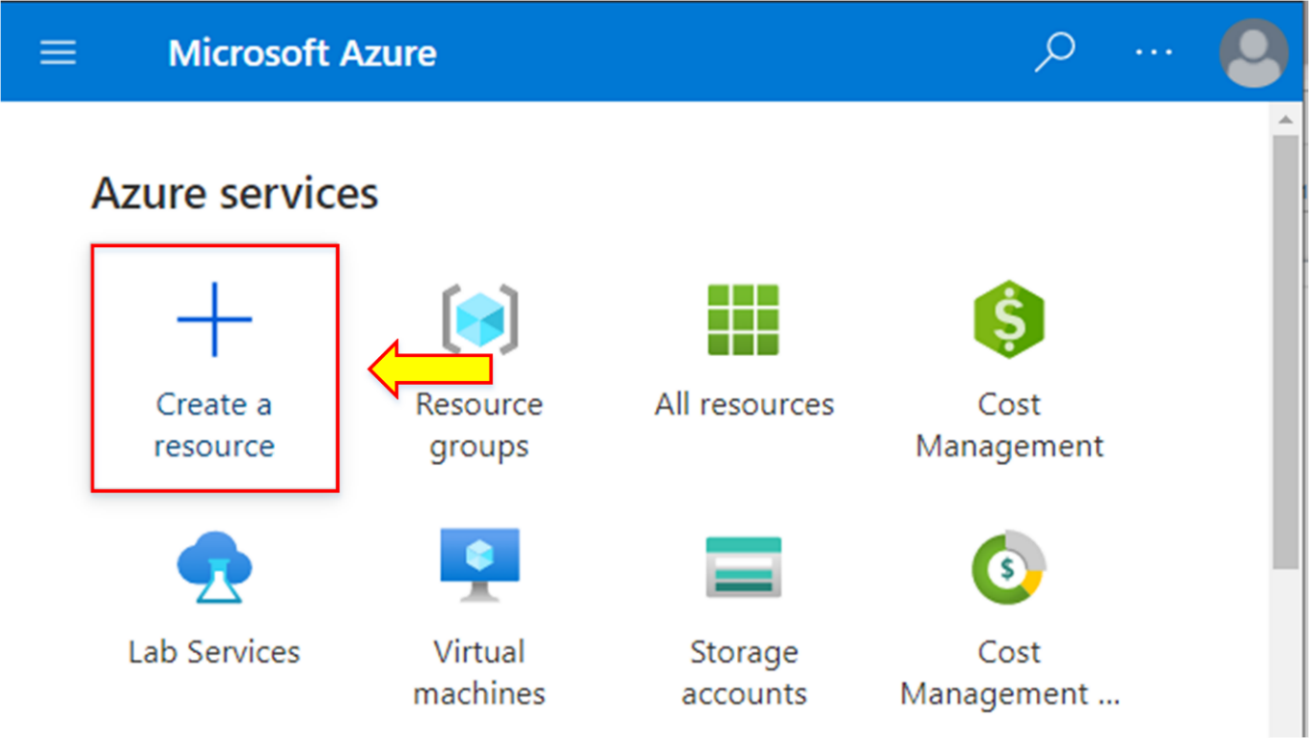
1. When the deployment is completed, a notification will appear. Click on **“Go to resource group.”**



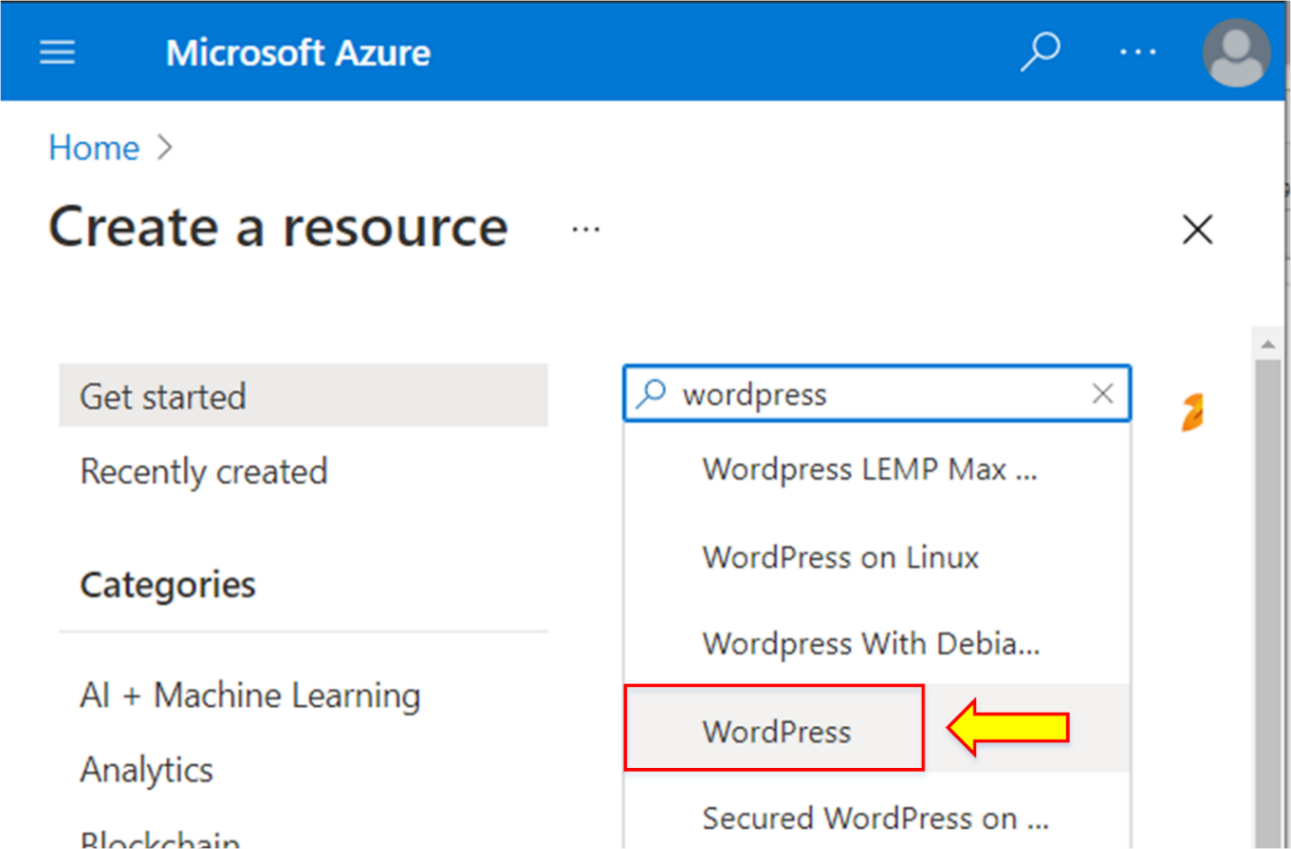
1. The overview page of the new resource group will appear.



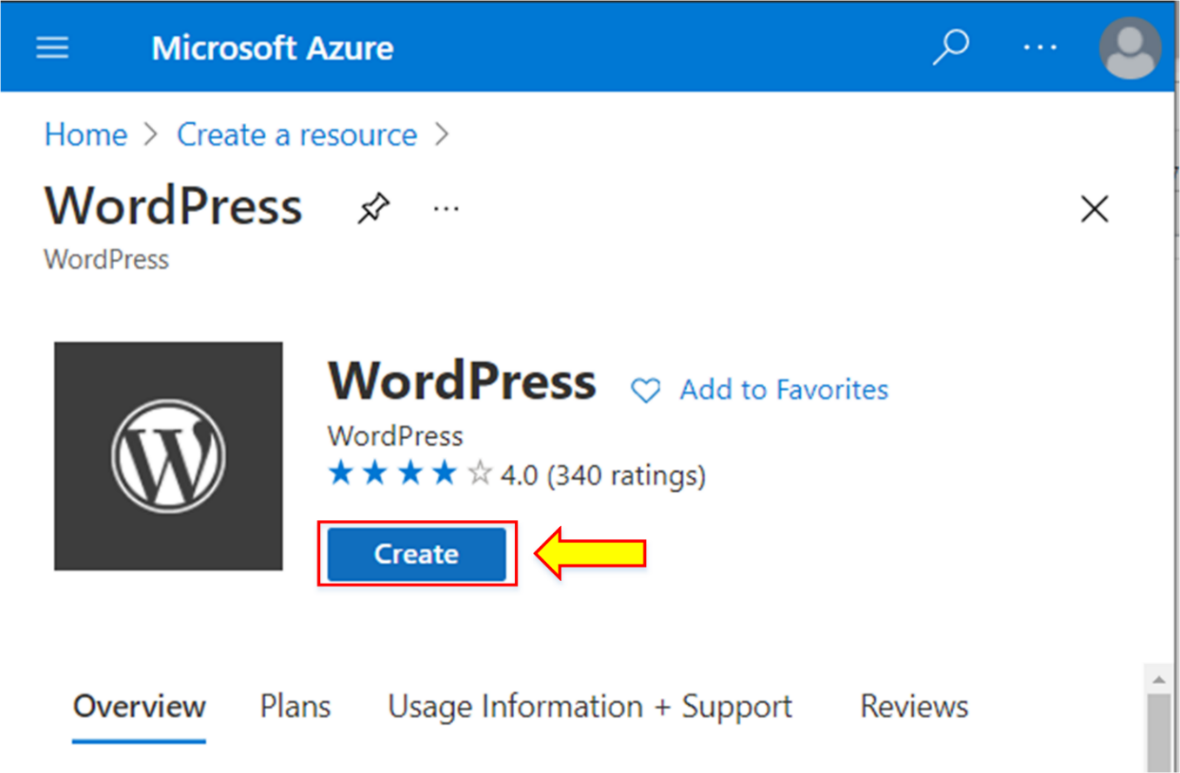
1. Now, go to the portal home page, and click on “**Create a resource**” for the website.



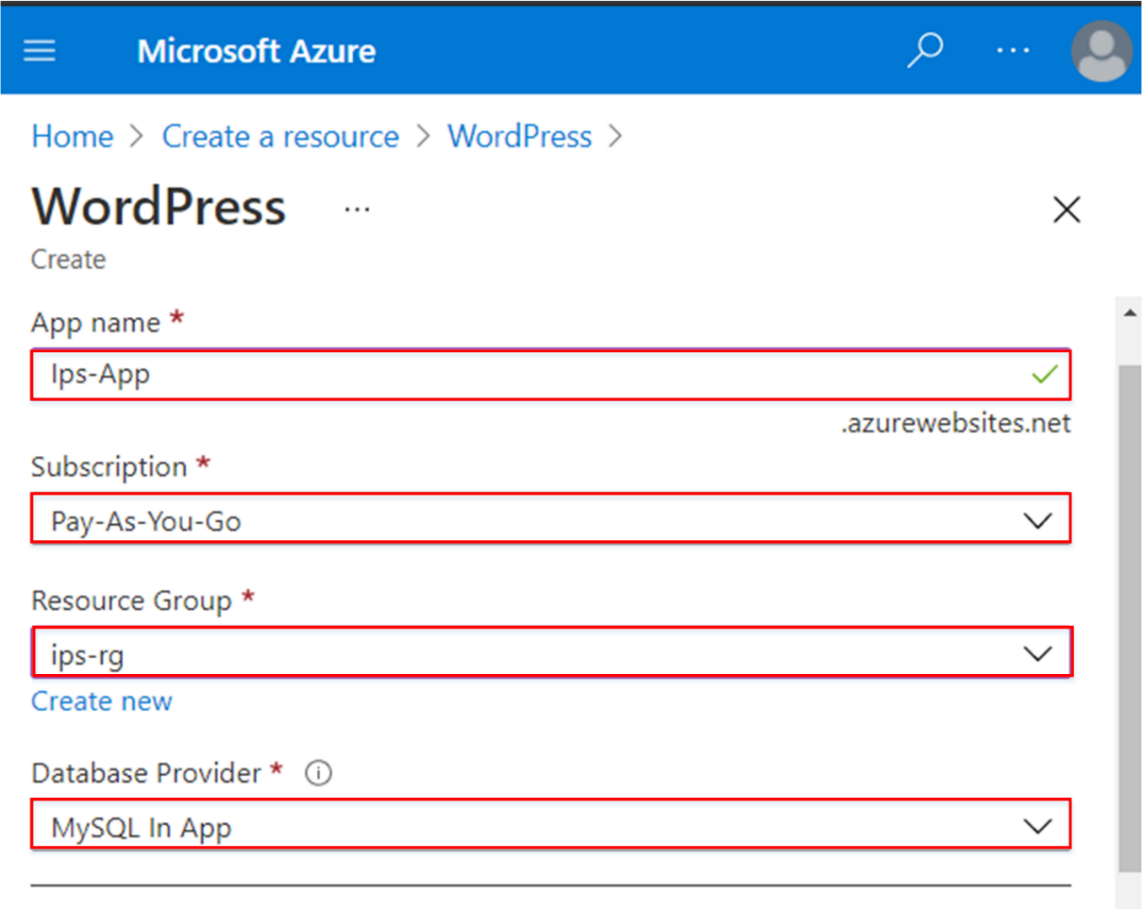
1. In the search bar of “**Azure Marketplace,**” search and open “**WordPress.”**



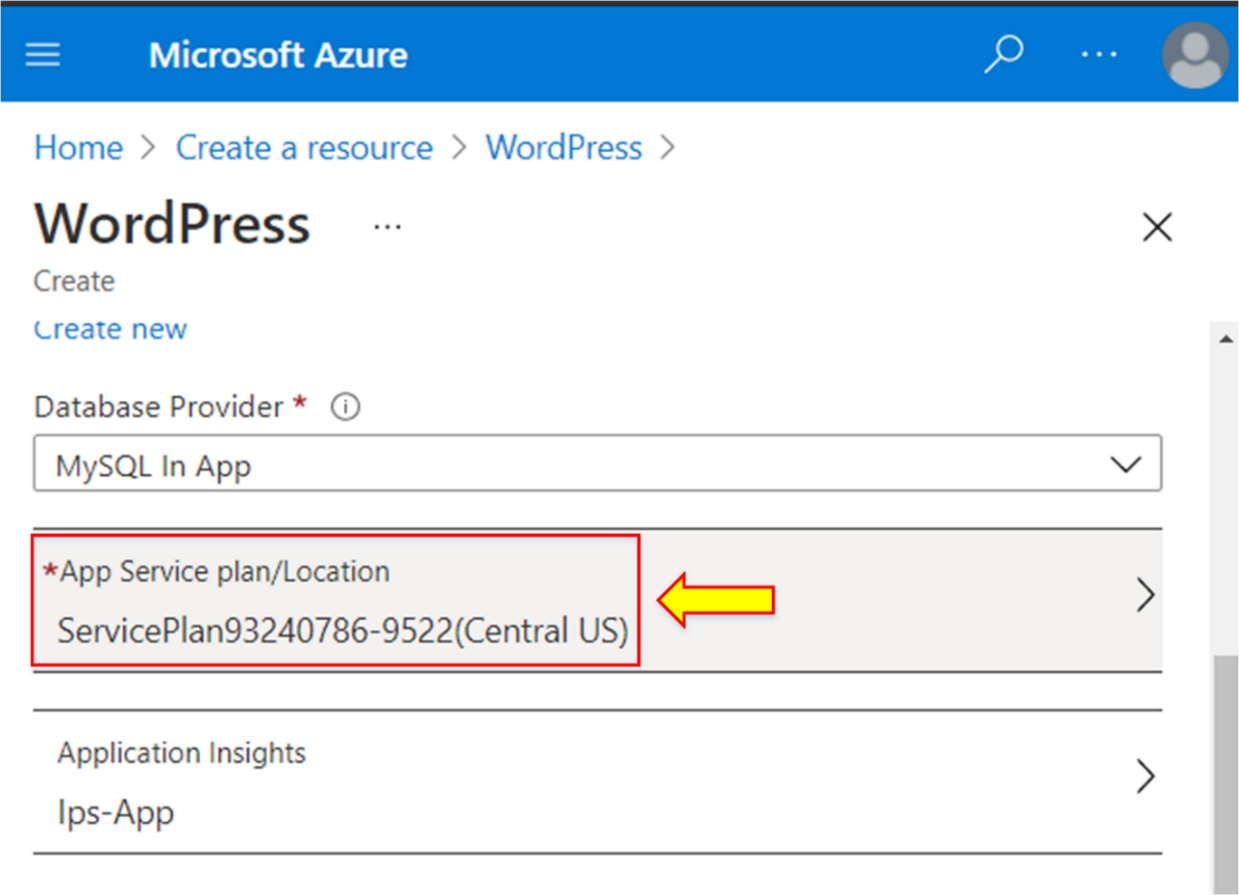
1. Click on “**Create.”**



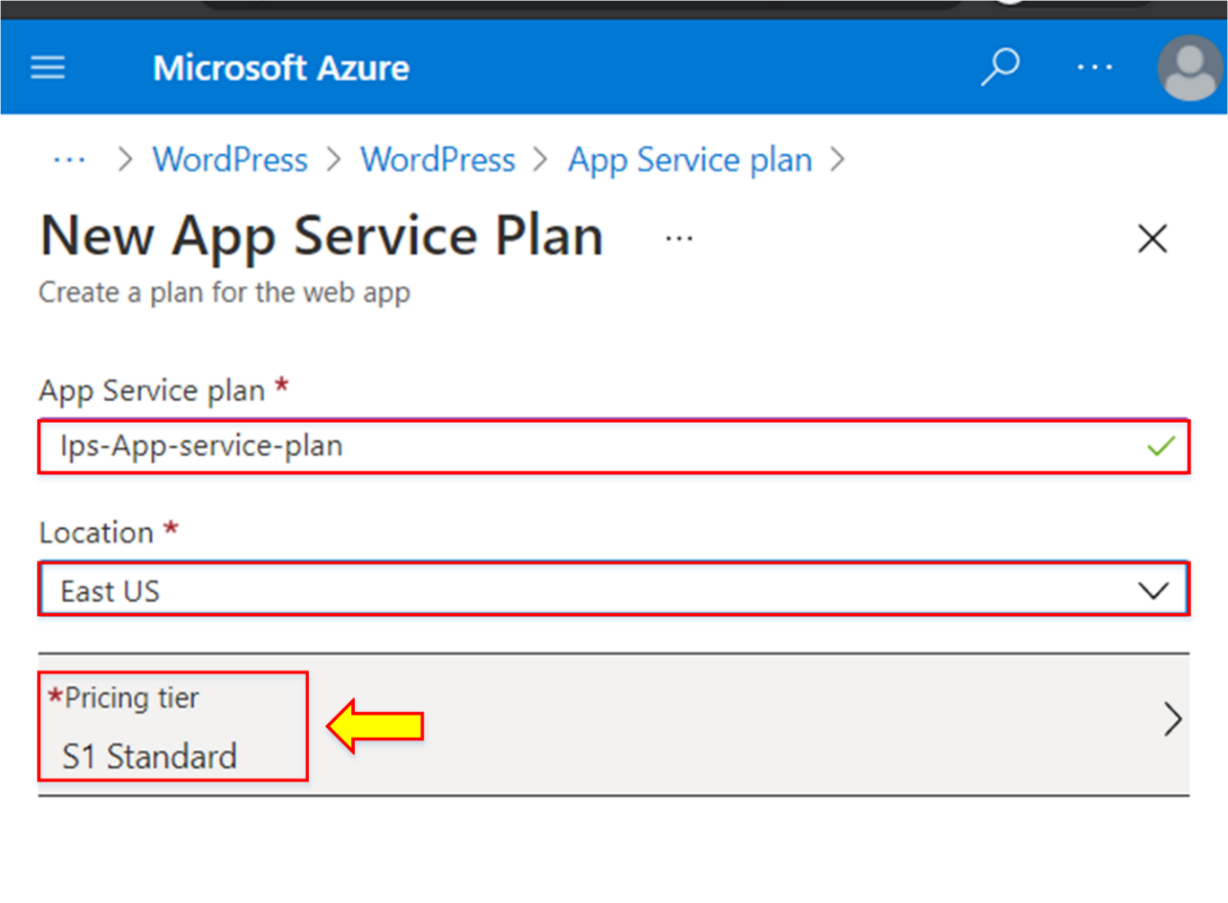
1. On the configuration pane, write the app name.
2. Select your subscription.
3. Choose the recently created resource group.
4. Select “**MySQL In-App**” as “**Database Provider.”**



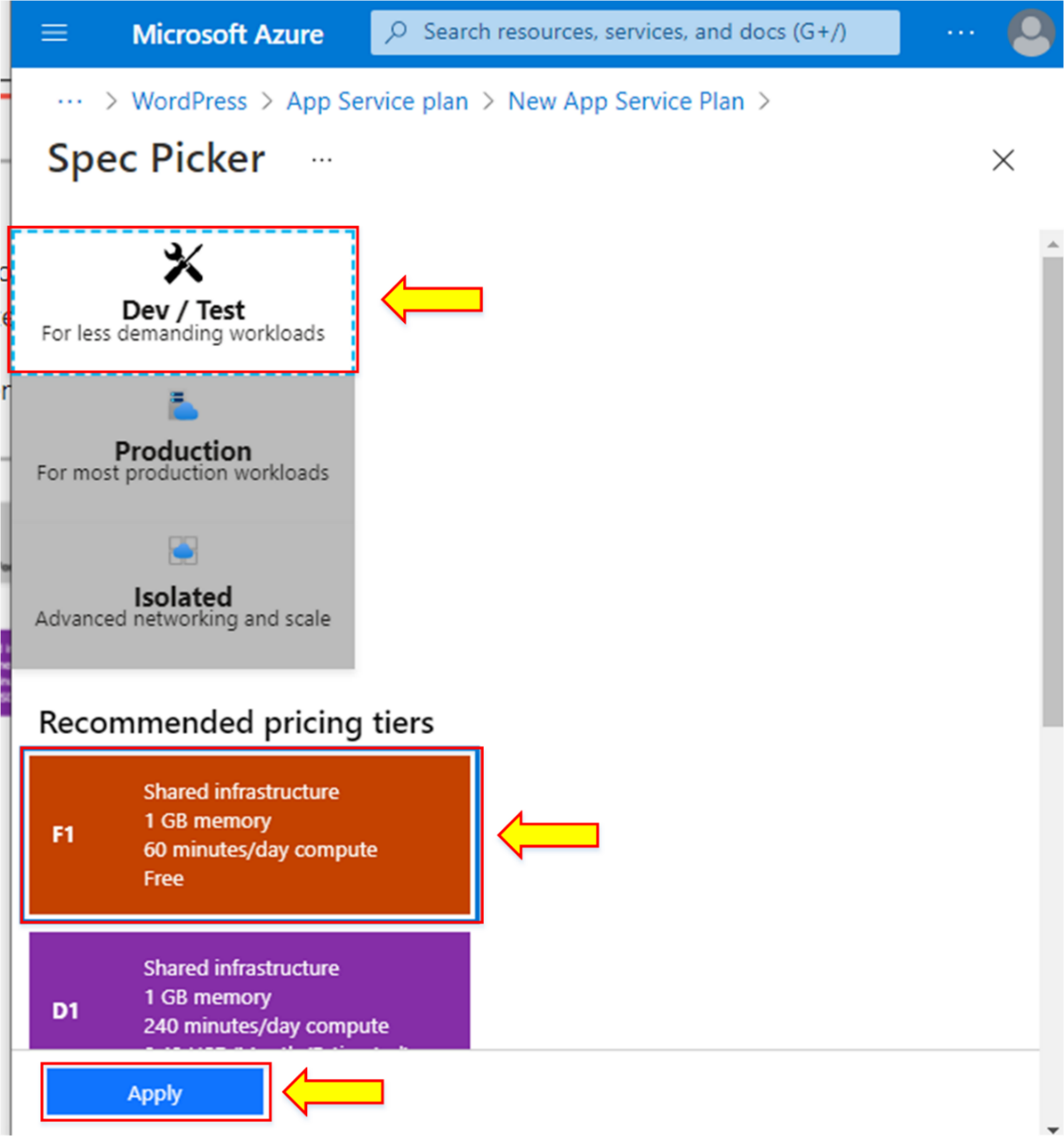
1. Click on “**App Service plan/Location.”**



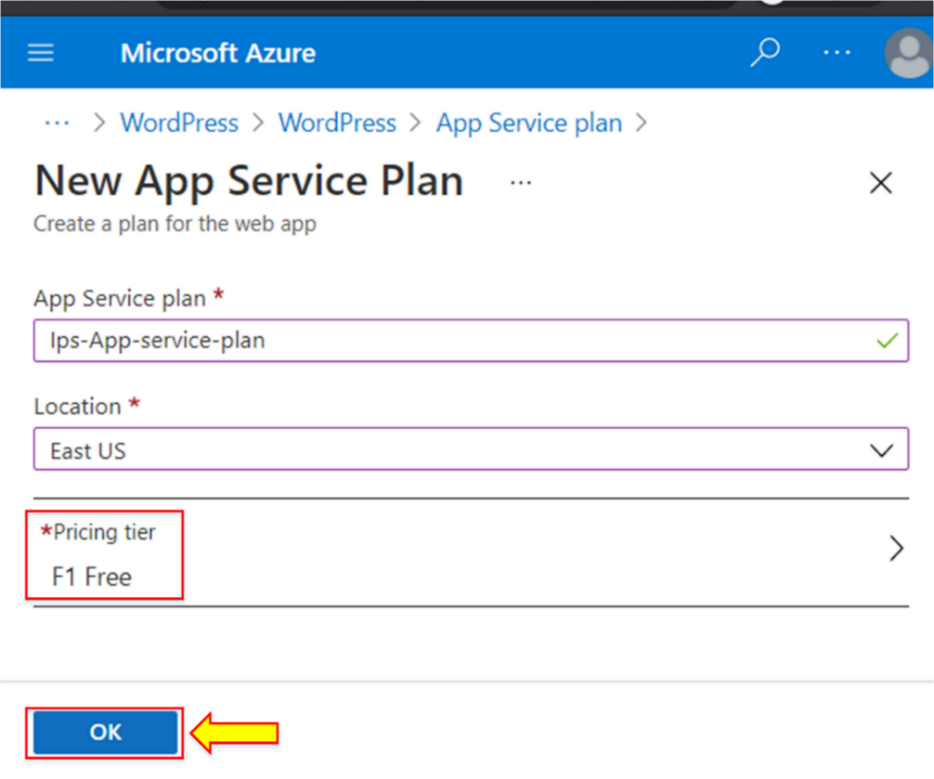
1. On the “**New App Service Plan**” dialog box, write the app service plan name.
2. Choose the nearest location.
3. Click on the “**Pricing tier**” option.



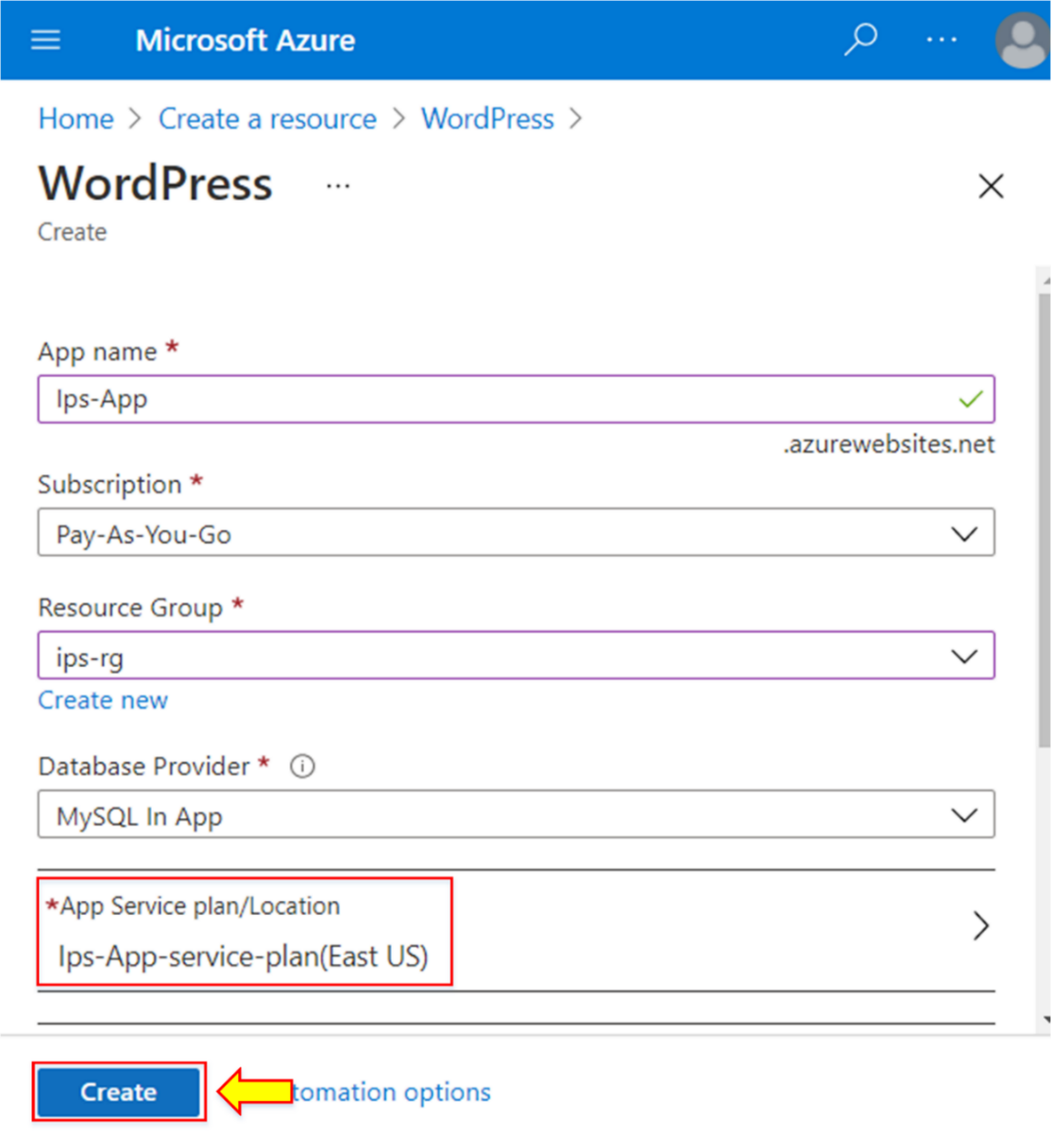
1. Select the “Dev / Test” category to change the pricing tier.
2. Scroll down and select the “**F1”** pricing tier.
3. Click on “**Apply.”**



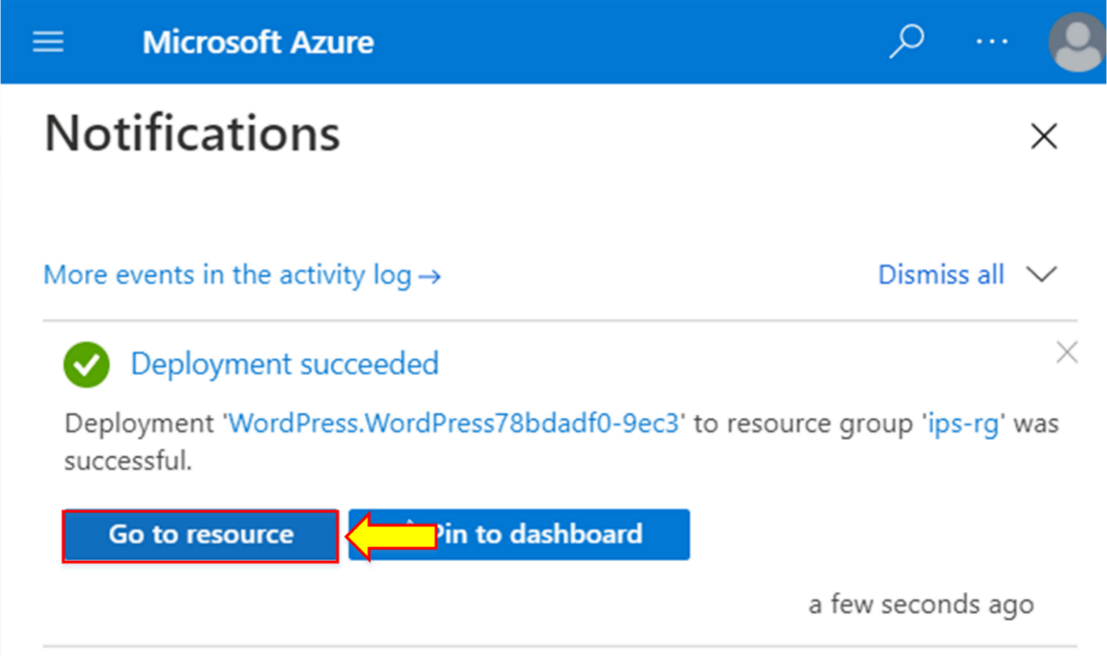
1. Afterward, return to the **“New App Service Plan”** dialog box and click on “**OK.”**



1. After setting up the “**App Service plan/Location,**” click “**Create.”**



1. A notification will appear as soon as the deployment is successful. Click on “**Go to resource.”**



1. The overview page of the website will appear, containing all the configuration details.
2. Navigate to the given **“URL.”**



1. A new “**WordPress**” site will appear. Select the language and click “**Continue**” to add the content.

