# Introduction to the Azure Portal

The new Azure portal is an all-in-one, work-anywhere experience. Now you can manage App Service Apps, databases and Visual Studio Team Services projects in a reimagined UX personalized to your work style. It was built from the ground up to put your *applications* at the center of the experience.

This unified hub radically simplifies building, deploying, and managing your cloud resources. Imagine a single, easy-to-use console built just for you, your team, your projects. It brings together all of the cloud resources, team members, and life-cycle stages of your application and provides you with a centralized place to plan, develop, test, provision, deploy, scale, and monitor those applications. This approach can help teams embrace a DevOps culture by bringing both development and operations capabilities and perspectives together in a meaningful way.

The new portal allows each user to transform the portal home page (called the *Dashboard*) into their own customized dashboard. Stay on top of the things that matter most by pinning them to your **Dashboard**. Re-size parts to show more or less data. Drill in for all the details. And see insights (and opportunities) across apps and resources. New components include the following:

- **Simplified Resource Management**. Rather than managing standalone resources such as Azure App Service Web Apps, Visual Studio Projects or databases, customers can now create, manage and analyze their entire application as a single resource group in a unified, customized experience, greatly reducing complexity while enabling scale. Today, the new Azure Manager is also available through the latest Azure SDK for customers to automate their deployment and management from any client or device.

- **Integrated billing**. A new integrated billing experience enables developers and IT pros to take control of their costs and optimize their resources for maximum business advantage.

- **Marketplace**. A rich marketplace of applications and services from Microsoft and the open source community. This integrated marketplace of free and paid services enables customers to leverage the ecosystem to be more agile and productive.

- **Visual Studio Team Services**. Microsoft announced key enhancements through the Microsoft Azure Portal. This includes Team Projects supporting greater agility for application lifecycle management and the lightweight editor code-named "Monaco" for modifying and committing Web project code changes without leaving Azure. Also included is Application Insights, an analytics solution that collects telemetry data such as availability, performance and usage information to track an application's health. Visual Studio integration enables developers to surface this data from new applications with a single click.

## Creating a Web App + SQL

Historically, managing a resource (a user-managed entity such as a database server, database or Web App) in Microsoft Azure required you to perform operations against one resource at a time. When developing for the cloud today, we are often managing *individual resources* (databases, storage, cloud services, virtual machines, and so on). It is left up to us, as cloud developers and IT professionals, to piece these resources together in a meaningful way and manage them over time. The Microsoft Azure portal was designed to bring together all of the individual resources of an application into a consolidated view. Resource group is a new concept in Azure that serves as the lifecycle boundary for all of its resources.

In this task, you will learn about the portal and how to create a new Web App and a SQL Database using it.

1. Open a browser and browse to http://portal.azure.com and log in using your credentials.

2. The first thing you will see is the **Dashboard**. This is your home page, where you can see dynamic data from your resources and all the details you care about. You can customize it as you see fit.

   > **Note:** You can right-click the tiles of the Dashboard to customize it. You can pin or unpin tiles and change their size.
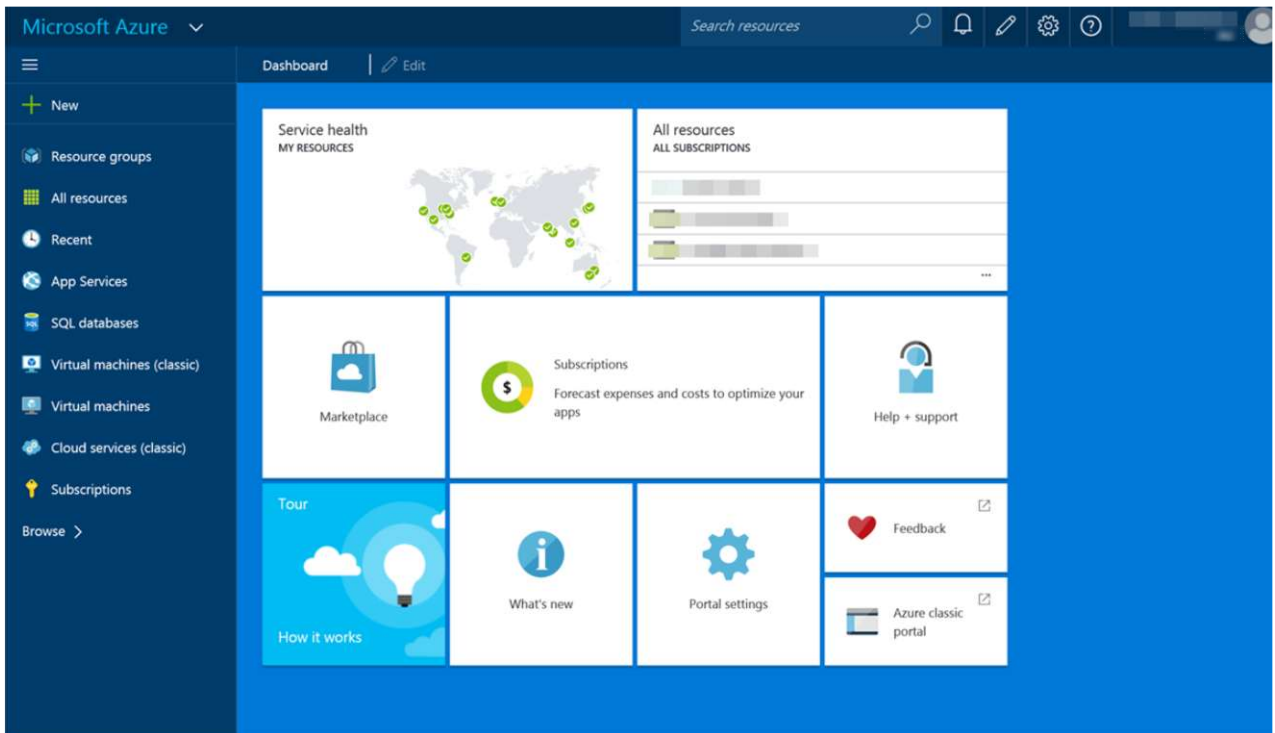
*Your Home Page: The Dashboard*

3. On the left side, you will see the **Hub Menu**. This is the navigation menu where you can access all of your resources and options. Click the **New** button at the top of the **Hub Menu**.
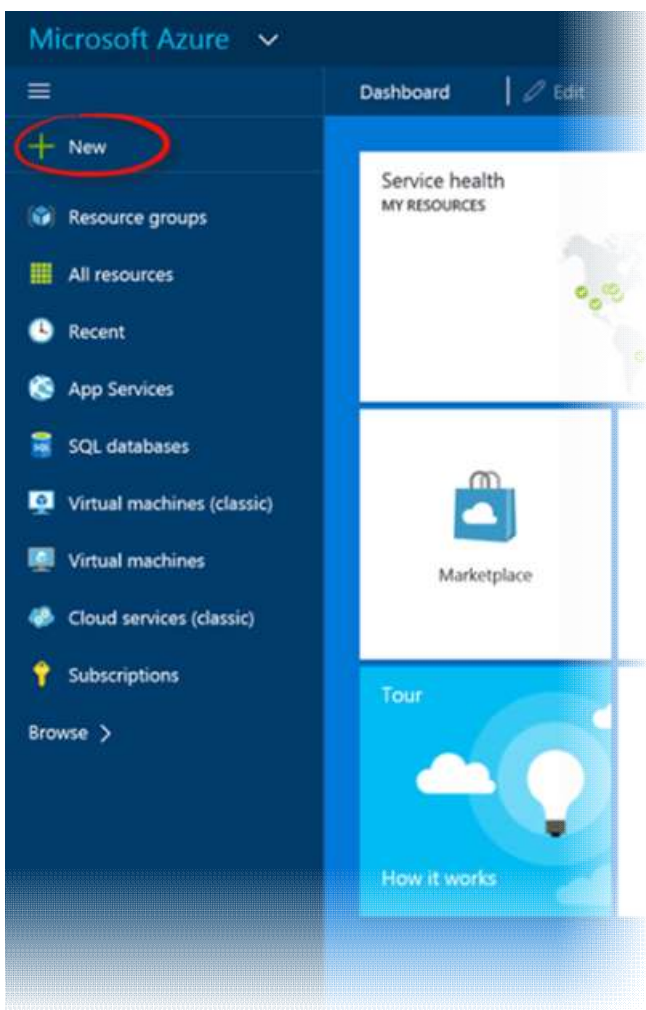


*Creating a new resource*

4. A panel is displayed with different options. You can choose one of the options to create a new resource. Click the **Web + mobile** category to see all the available options.



*Displaying every resource type*

On the next screen, you get all the available options for Web, Mobile Apps and other Azure App Service Apps. Click on the **See all**. You can find ready to use templates.

*Azure Marketplace*

5. Find the **Web app + SQL** option and then click it.

*Selecting Web app + SQL*

Then click **Create**.
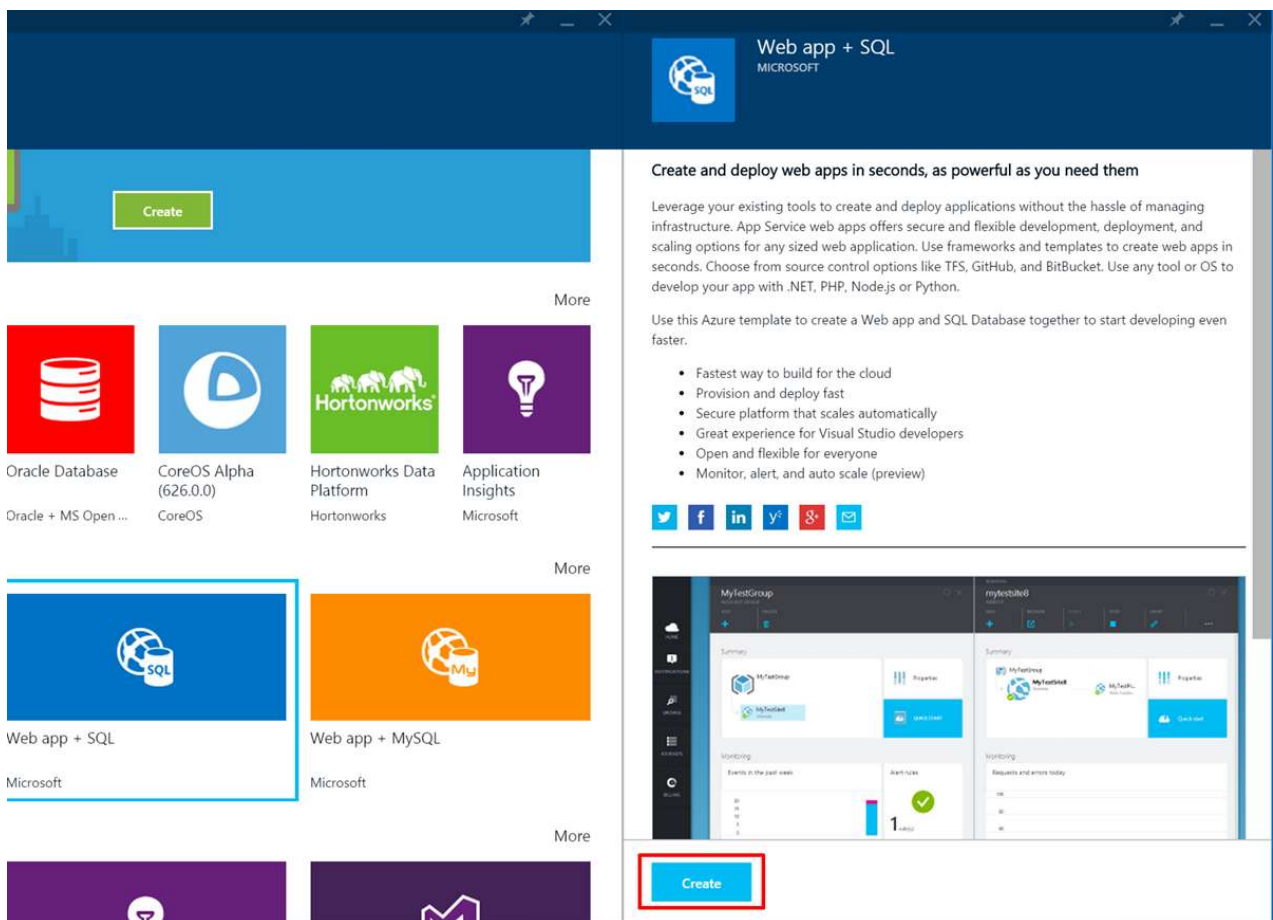
*Create a new Web app + SQL*

A *blade* is opened. Blades are your entry point to discovering insights, performing actions and building applications. This particular *blade* collects input from you to create a new **Web App**.

> **Note:** For more information about *blades*, you can click the **Tour** tile on your **Dashboard**. On the **Tour** blade, scroll down to the bottom and click **Learn more**. A new blade is opened with further information. You can continue the **Tour** to learn the basics of Blades, Commands, Lenses and more.
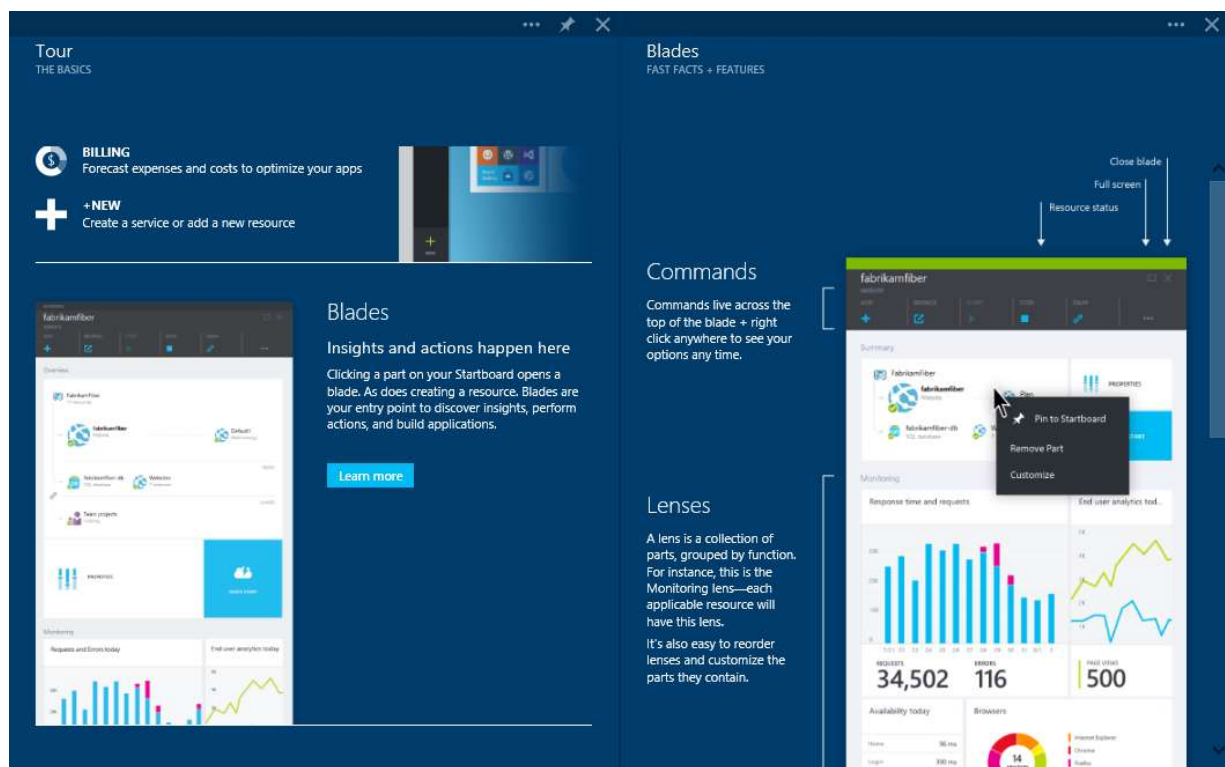
6. Select *App Service Name* option, another blade is opened which displays the options to create a new **Web app**. Select a URL for your Web app, for example *myMSAzureWebsite*. Take into account that this name must be unique. Also fill in the **Create new App Service Plan** or **Select existing** if you already`have one.
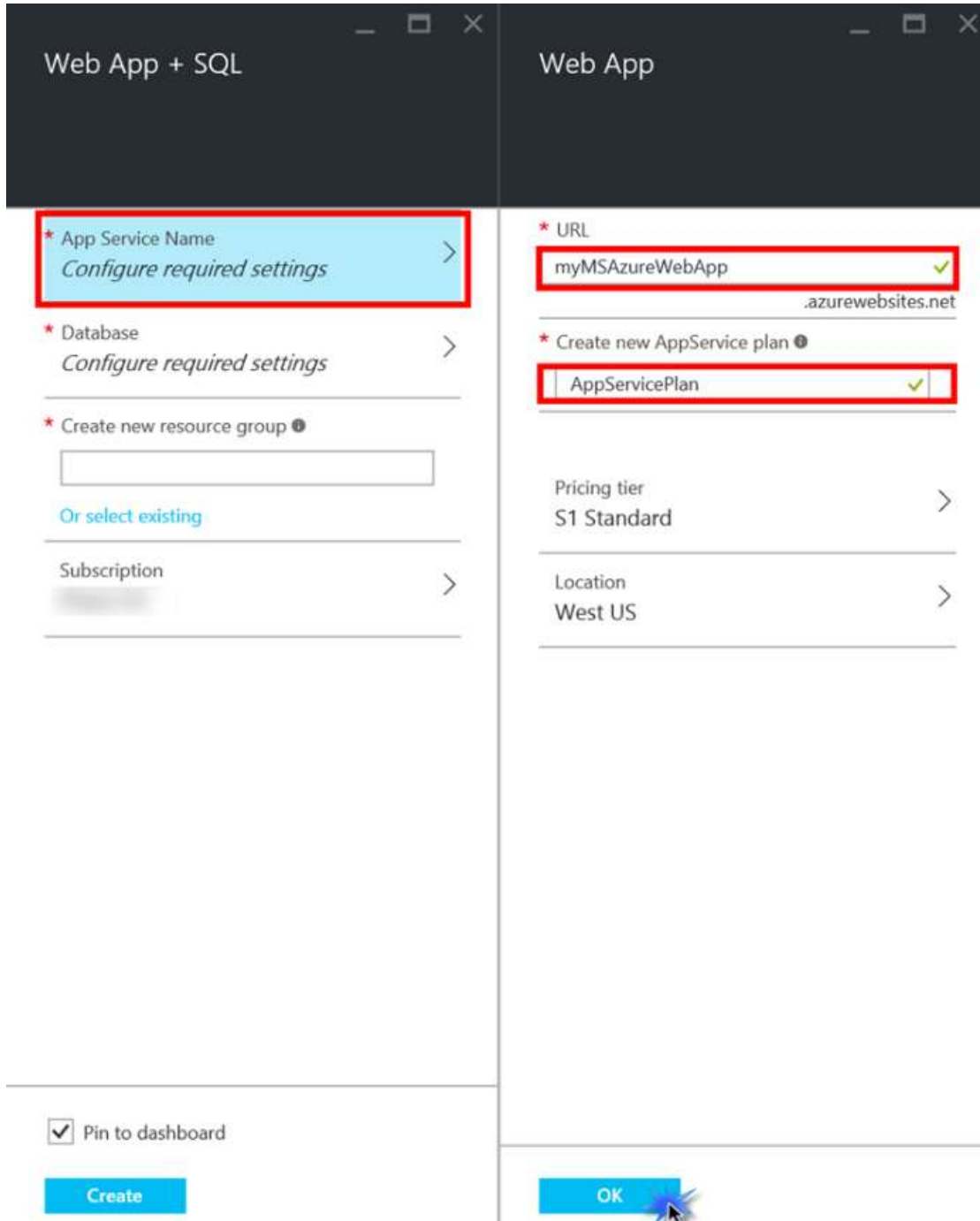


*Creating the App Service Plan*

7. If you would like to change the Pricing Tier, click on the **Pricing Tier**. In the *Choose your pricing tier* blade, you can choose the App Service plan that fits your needs.

App Service plans represent a set of features and capacity that you can share across your App Service Apps, including Web Apps. App Service plans support several pricing tiers (e.g. Free, Shared, Basic, Standard and Premium), each with its own capabilities. There are a couple of differences among these tiers. Plans in the Free and Shared tier provide sites with a shared infrastructure, meaning that your sites share resources with other customers' sites. App Service plans in the Basic, Standard and Premium tiers provider sites with a dedicated infrastructure, meaning that only the app or apps you choose to associate with this plan will be running on those resources. In this tier you can configure your App Service Plan to use one or more virtual machine instances. As we are going to use the default App Service plan, just close this blade.

**Note:** For all tiers (except 'Shared') you pay one price for the App Service plan based on the tier and your chosen capacity with no additional charge for each site that uses the plan. Shared App Service plans are different; due to the nature of the shared infrastructure, you are charged separately for each app in the plan. Click *Select* once you select a relevant plan. This should take you to the previous blade for *Web App* options.



*Selecting a Pricing Tier*

8. Select a relevant location and click OK.

9. Click **Database** to change the settings for your new database.

   **Note:** If there are any existing databases asociated to the user, the **Database** blade will show up. Select **Create a new Database**.
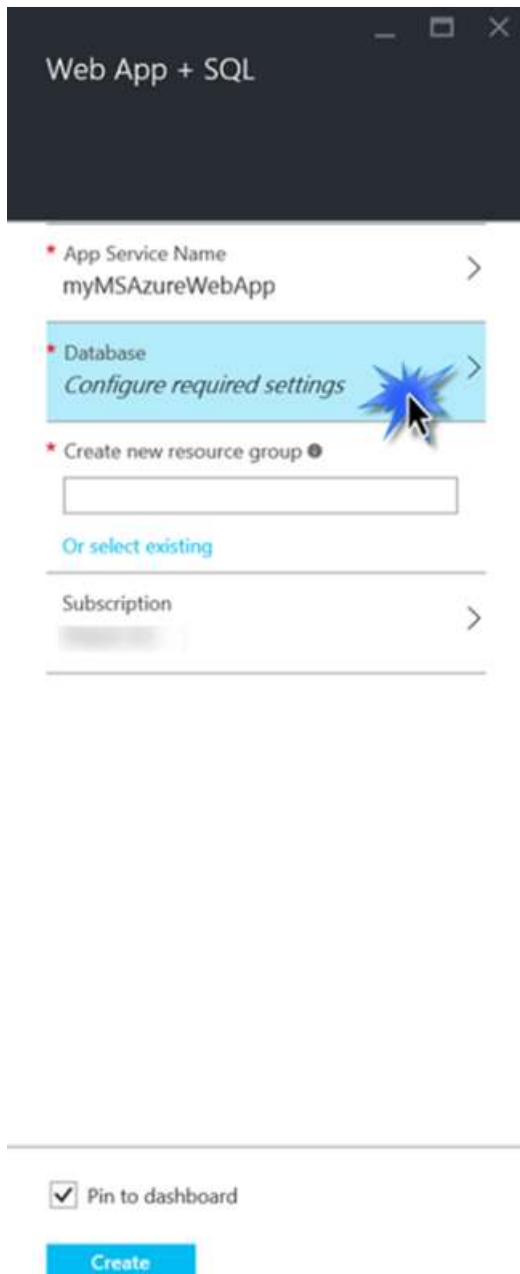
*Changing your database settings*

10. Set a name for the database, e.g. *mywebsite-db*, and click the **Server** option.

> **Note:** You can also enter in the **Pricing Tier** section and explore the different pricing tiers.
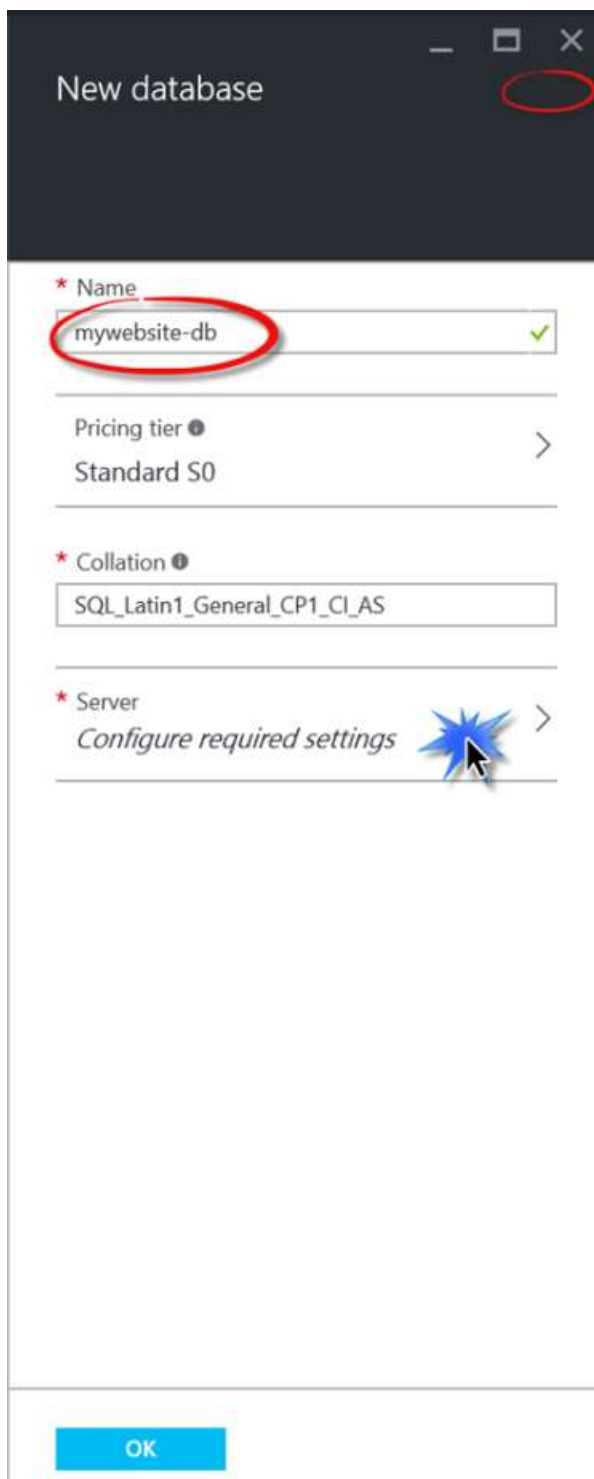
*Database Settings*

11. Enter the **Server Name**, **Server admin login** and a **Password**. Click **OK** to go back to the **New Database** blade, and click **OK** to close it.

*Configuring the Database Server*

12. When you create an application that consists of several resources working together (like in this example, a Web app + SQL), it is always created in its own resource group so that you can manage the lifecycle of all related assets. Choose a name for the **Resource Group**, for example *MyResourceGroup*, and click the **Web app** option.

> **Note:** Resource group names can only contain letters, numbers, periods, underscores and dashes.
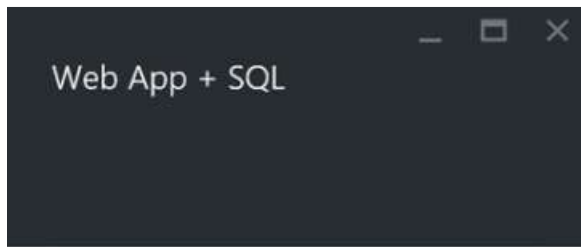
## Web App + SQL

* App Service Name
myMSAzureWebApp

* Database
mywebsite-db

* Create new resource group

MyResourceGroup

Or select existing

Subscription

☑ Pin to dashboard

Create

*New Resource Group*

13. Now you are ready to create your resource group. Click **Create**.

14. On the right top, click bell icon for **Notifications**. You can see status of the creation process in the **Notifications** section.
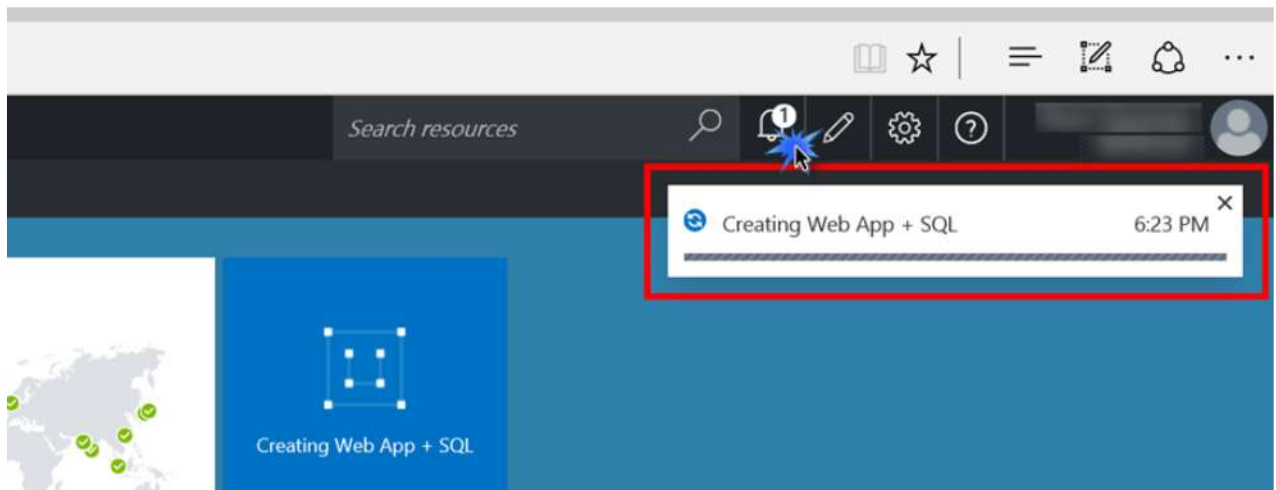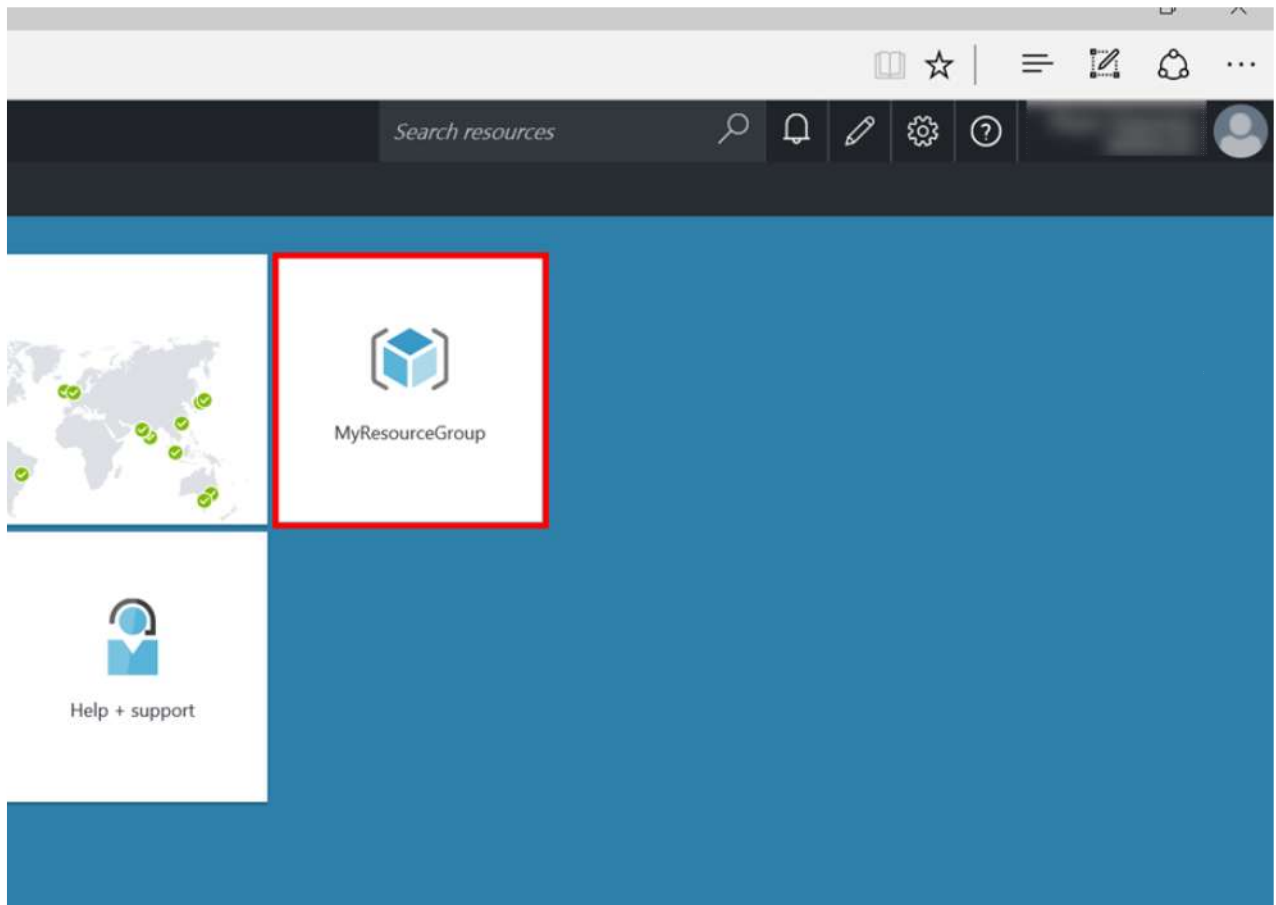
*Notifications*

15. Once the Resource group is successfully created the notification message accordingly displays the relevant message.
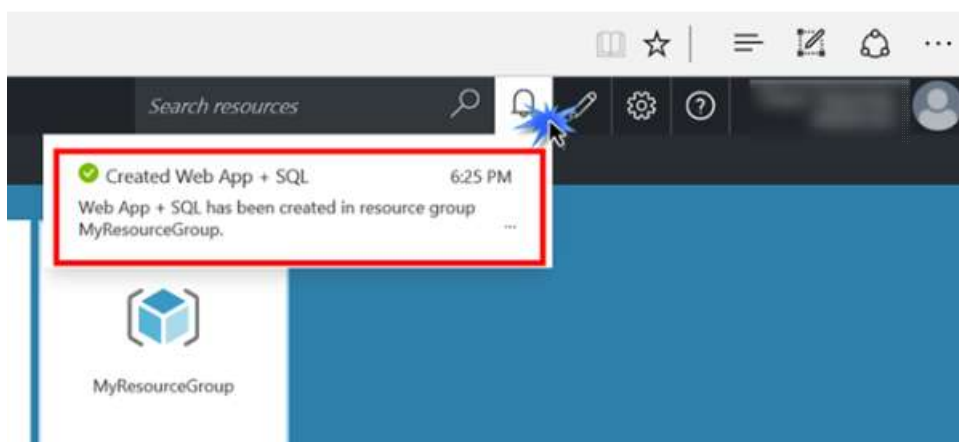


*Created Resource Group Notification*

16. Notice that the recently created resource group was automatically added to the Dashboard for easy access.

17. Click on the name of the resource group to open it. You created your new resource group, which includes a Web App, a SQL Server and a SQL Database.
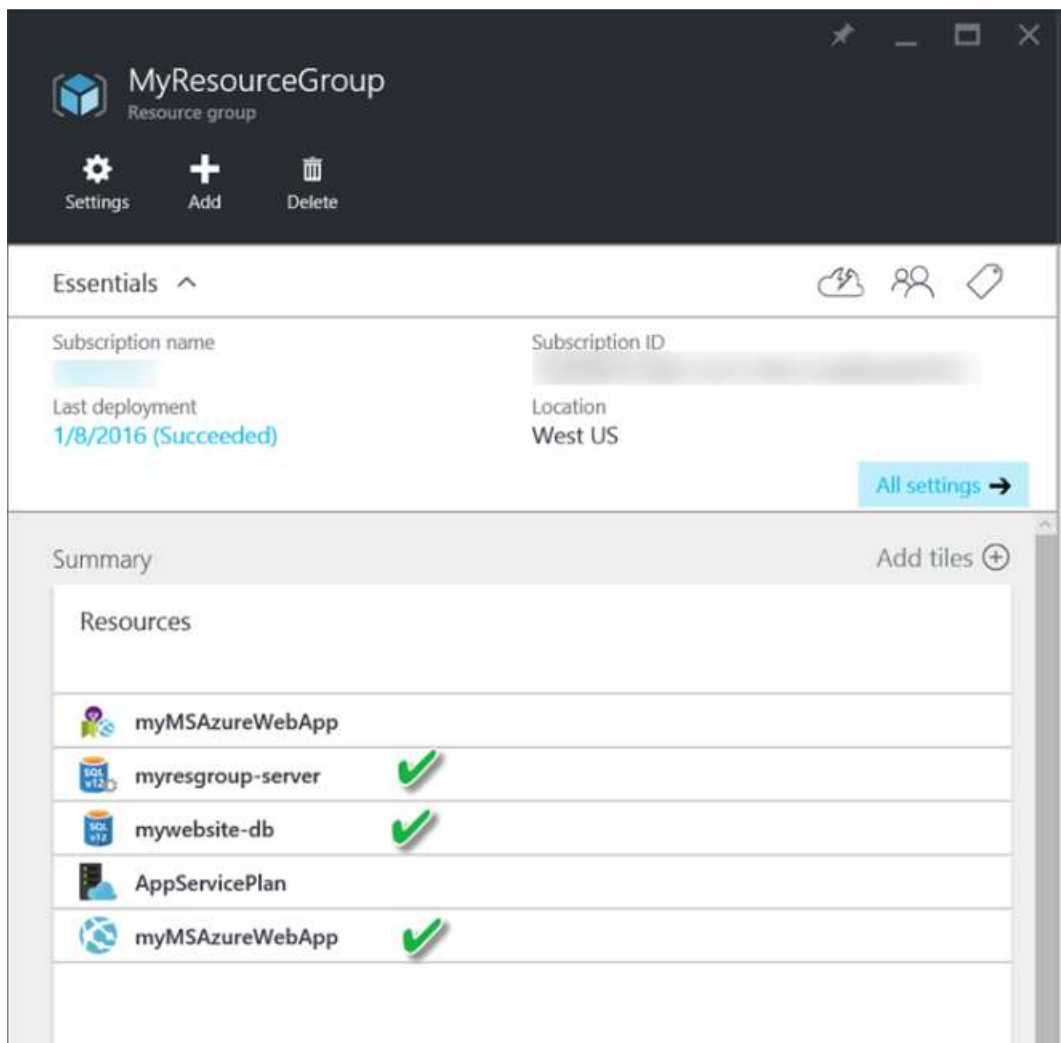
**Note:** It is good practice to cleanup any resources that will not be used. To delete the demo resources just created, follow these steps:

i. In the Resource Group, click **DELETE** at the top of the page.



*Clicking Delete*

i. In the Confirmation blade that opens, type the name of the resource group and click the **Delete** button.

*Delete Confirmation*

After a few moments you should see a notification in the Notifications Hub indicating the resource group was deleted.
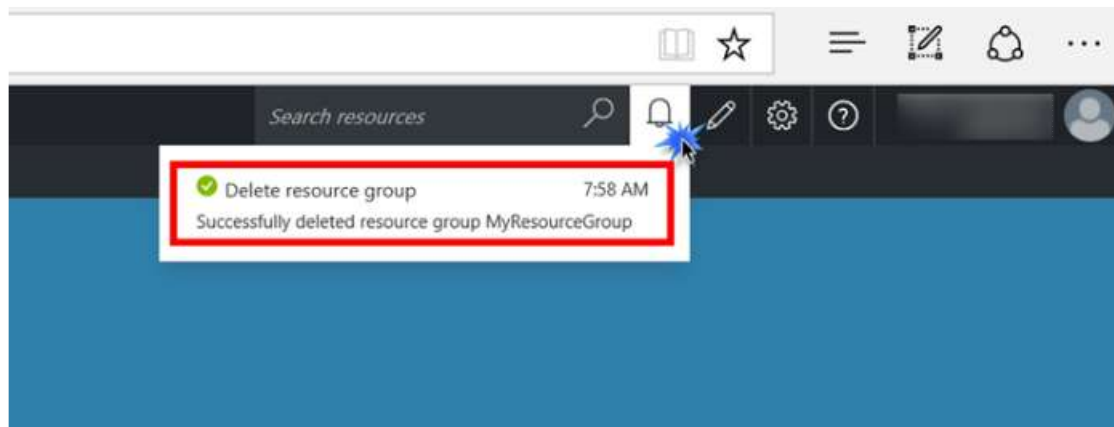
i. Navigate to the Dashboard. Once there, find and right-click the tile for the resource group. Click **unpin from Dashboard**.



*Unpinning Resource Group from DashBoard* The tile is removed.

## Summary

The new Azure portal offers an exciting look into the future of DevOps. This is a first-of-its-kind experience which brings together all of the individual resources, people, and life-cycle stages of your application into a unified portal. In this lab, you learned about the portal and how to create a new resource group by building a Web App.