**Run your first Batch job in the Azure portal**

This quickstart shows how to use the Azure portal to create a Batch account, a *pool* of compute nodes (virtual machines), and a *job* that runs basic *tasks* on the pool. After completing this quickstart, you will understand the key concepts of the Batch service and be ready to try Batch with more realistic workloads at larger scale.

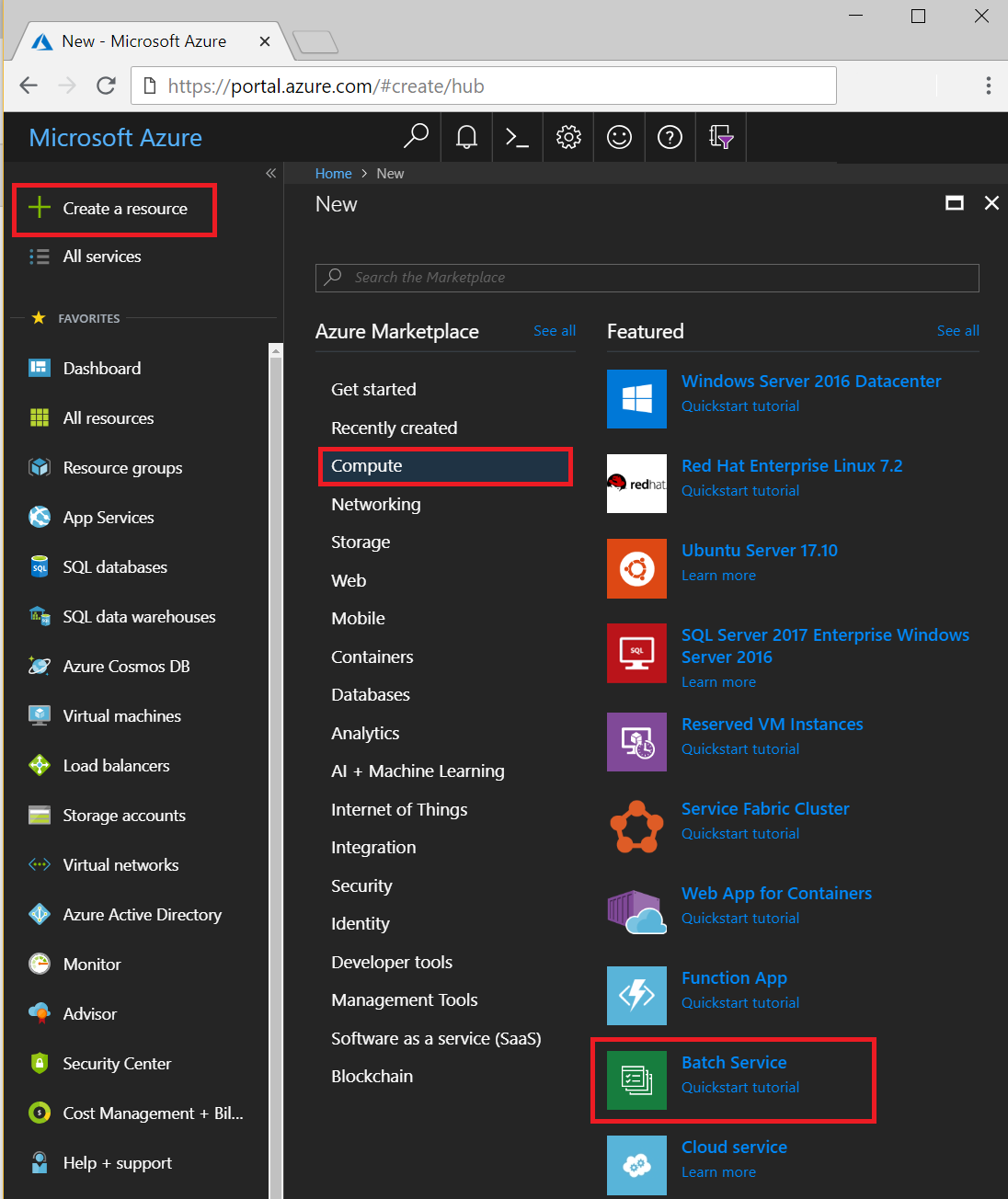
**Sign in to Azure**

Sign in to the Azure portal at [https://portal.azure.com](https://portal.azure.com/).

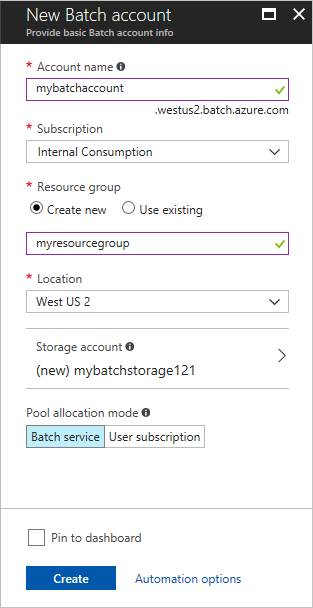
**Create a Batch account**

Follow these steps to create a sample Batch account for test purposes. You need a Batch account to create pools and jobs. As shown here, you can link an Azure storage account with the Batch account. Although not required for this quickstart, the storage account is useful to deploy applications and store input and output data for most real-world workloads.

1. Select **Create a resource** > **Compute** > **Batch Service**.



1. Enter values for **Account name** and **Resource group**. The account name must be unique within the Azure **Location** selected, use only lowercase characters or numbers, and contain 3-24 characters.
2. In **Storage account**, select an existing storage account or create a new one.
3. Keep the defaults for remaining settings, and select **Create** to create the account.



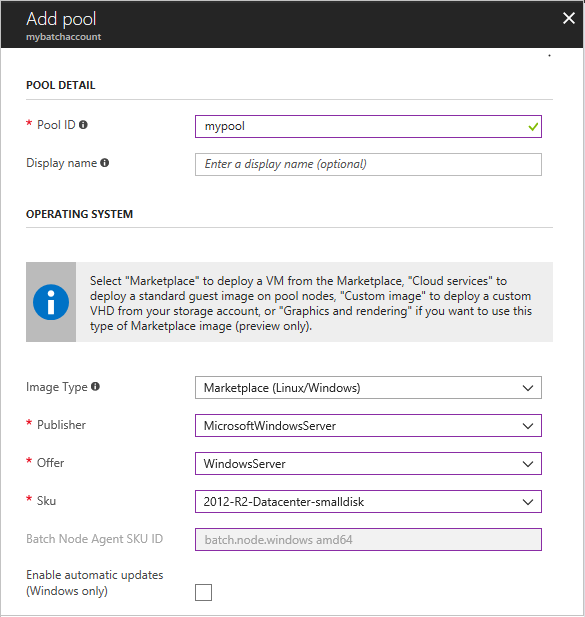
When the **Deployment succeeded** message appears, go to the Batch account in the portal.

**Create a pool of compute nodes**

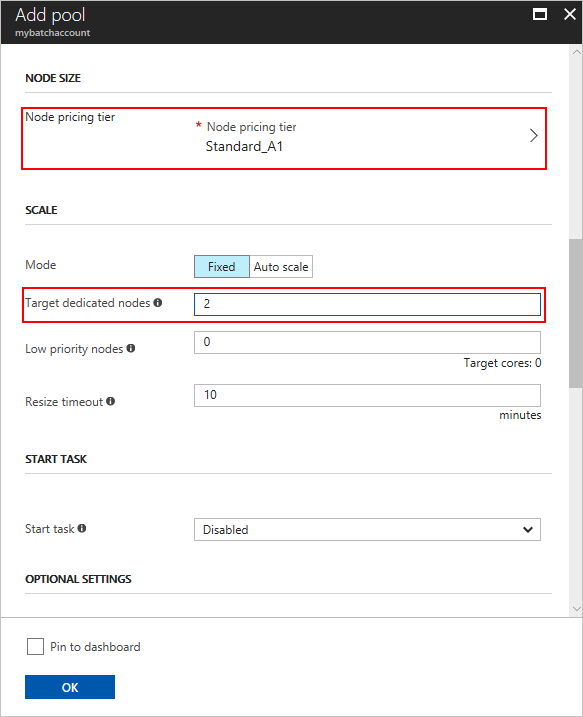
Now that you have a Batch account, create a sample pool of Windows compute nodes for test purposes. The pool for this quick example consists of 2 nodes running a Windows Server 2012 R2 image from the Azure Marketplace.

1. In the Batch account, select **Pools** > **Add**.
2. Enter a **Pool ID** called *mypool*.
3. In **Operating System**, select the following settings (you can explore other options).

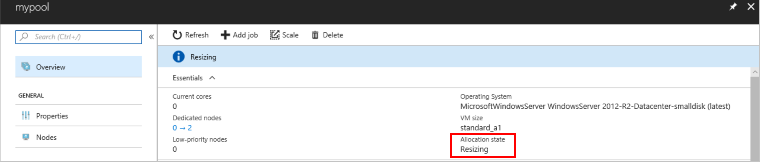
|  |  |
| --- | --- |
| **Setting** | **Value** |
| **Image Type** | Marketplace (Linux/Windows) |
| **Publisher** | MicrosoftWindowsServer |
| **Offer** | WindowsServer |
| **Sku** | 2012-R2-Datacenter-smalldisk |

1. 
2. Scroll down to enter **Node Size** and **Scale** settings. The suggested node size offers a good balance of performance versus cost for this quick example.

|  |  |
| --- | --- |
| **Setting** | **Value** |
| **Node pricing tier** | Standard\_A1 |
| **Target dedicated nodes** | 2 |

1. 
2. Keep the defaults for remaining settings, and select **OK** to create the pool.

Batch creates the pool immediately, but it takes a few minutes to allocate and start the compute nodes. During this time, the pool's **Allocation state** is **Resizing**. You can go ahead and create a job and tasks while the pool is resizing.

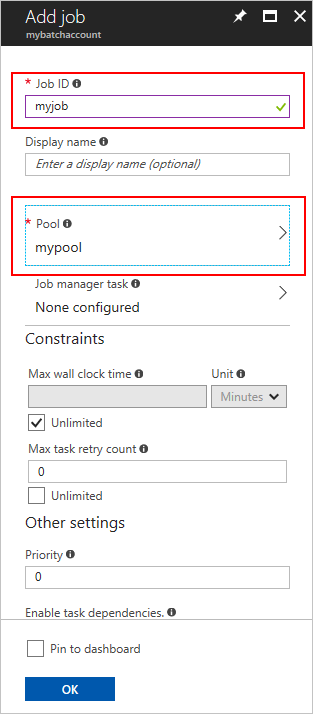


After a few minutes, the state of the pool is **Steady**, and the nodes start. Select **Nodes** to check the state of the nodes. When a node's state is **Idle**, it is ready to run tasks.

**Create a job**

Now that you have a pool, create a job to run on it. A Batch job is a logical group for one or more tasks. A job includes settings common to the tasks, such as priority and the pool to run tasks on. Initially the job has no tasks.

1. In the Batch account view, select **Jobs** > **Add**.
2. Enter a **Job ID** called *myjob*. In **Pool**, select *mypool*. Keep the defaults for the remaining settings, and select **OK**.



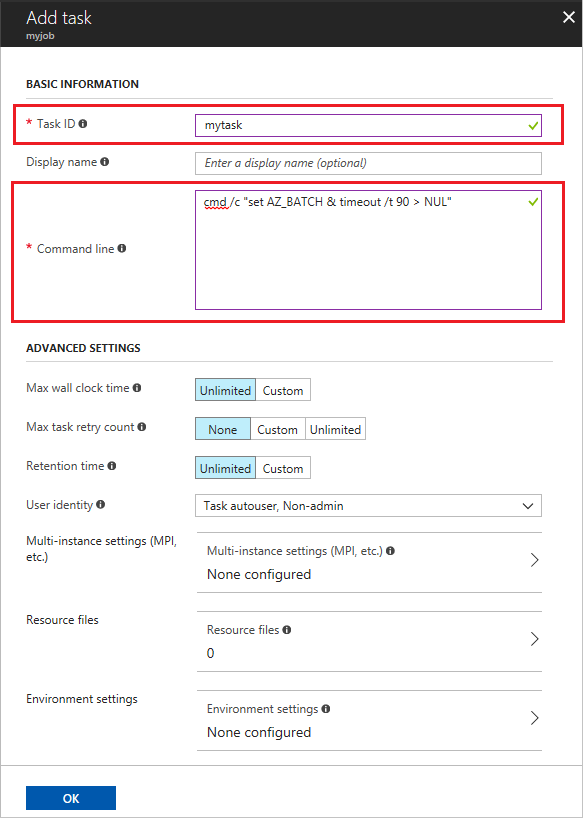
After the job is created, the **Tasks** page opens.

**Create tasks**

Now create sample tasks to run in the job. Typically you create multiple tasks that Batch queues and distributes to run on the compute nodes. In this example, you create two identical tasks. Each task runs a command line to display the Batch environment variables on a compute node, and then waits 90 seconds.

When you use Batch, the command line is where you specify your app or script. Batch provides a number of ways to deploy apps and scripts to compute nodes.

To create the first task:

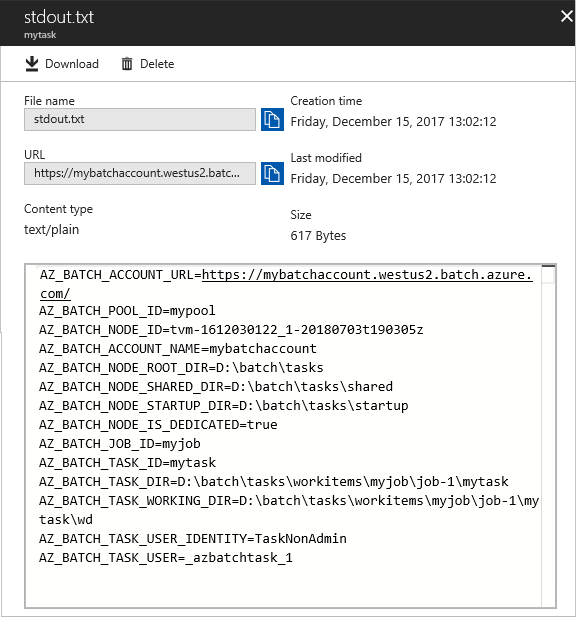
1. Select **Add**.
2. Enter a **Task ID** called *mytask*.
3. In **Command line**, enter cmd /c "set AZ\_BATCH & timeout /t 90 > NUL". Keep the defaults for the remaining settings, and select **OK**.

After you create a task, Batch queues it to run on the pool. When a node is available to run it, the task runs.

To create a second task, go back to step 1. Enter a different **Task ID**, but specify an identical command line. If the first task is still running, Batch starts the second task on the other node in the pool.

**View task output**

The preceding task examples complete in a couple of minutes. To view the output of a completed task, select **Files on node**, and then select the file stdout.txt. This file shows the standard output of the task. The contents are similar to the following:



The contents show the Azure Batch environment variables that are set on the node. When you create your own Batch jobs and tasks, you can reference these environment variables in task command lines, and in the apps and scripts run by the command lines.

**Clean up resources**

If you want to continue with Batch tutorials and samples, use the Batch account and linked storage account created in this quickstart. There is no charge for the Batch account itself.

You are charged for the pool while the nodes are running, even if no jobs are scheduled. When you no longer need the pool, delete it. In the account view, select **Pools** and the name of the pool. Then select **Delete**. When you delete the pool, all task output on the nodes is deleted.

When no longer needed, delete the resource group, Batch account, and all related resources. To do so, select the resource group for the Batch account and select **Delete resource group**.