## **Creating Site Collections**

**Lab Time**: 45 minutes

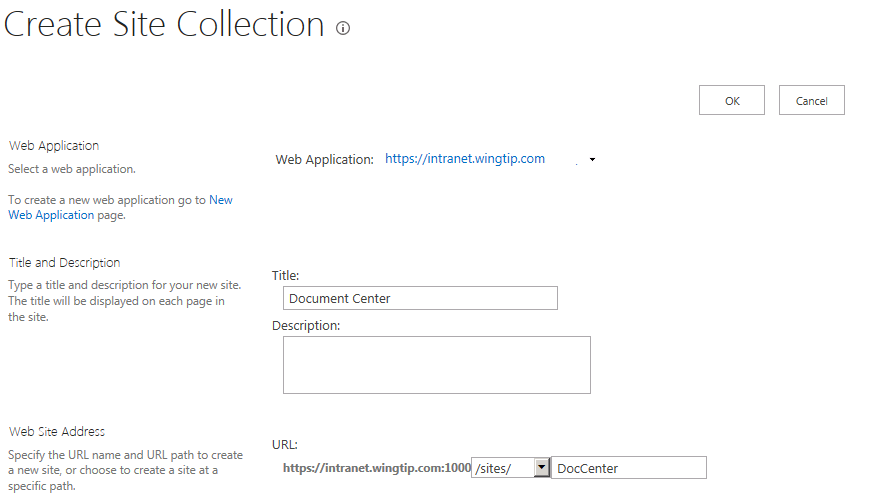
**Lab Folder**: C:\Student\Modules\SiteCollections\Lab

**Lab Overview**: In this lab you create several different types of site collections.

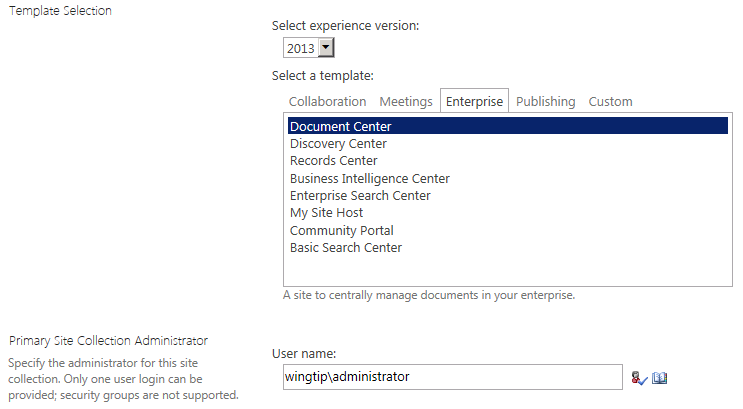
### Exercise 1: Create a Document Center Site in the Wingtip Intranet

In this exercise you will use Central Administration to create a new site collection with a top-site based on the Document Center site template.

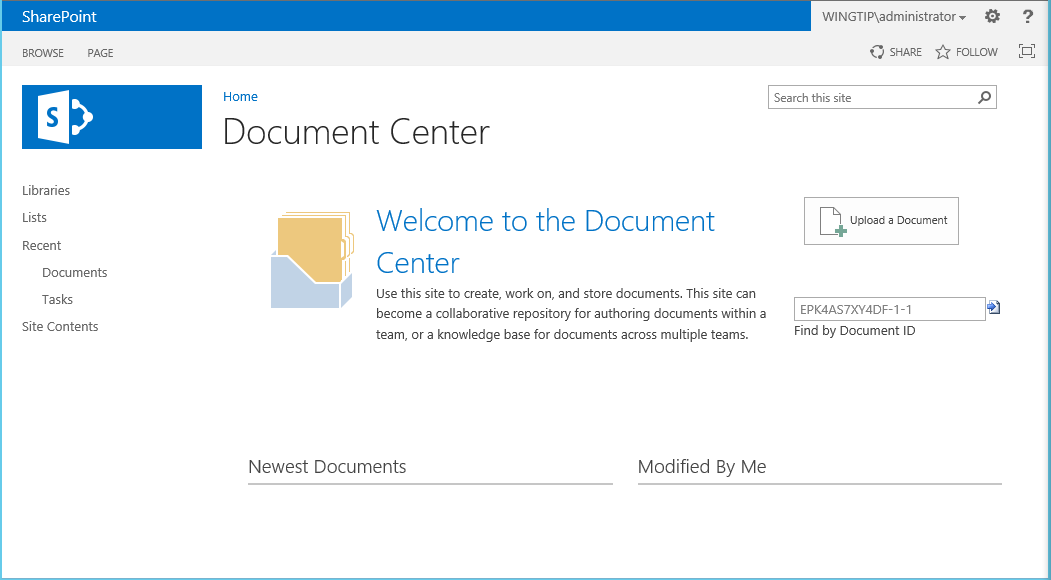
1. Ensure you are logged onto the **WingtipServer** server as **WINGTIP\Administrator**.
2. Navigate to **Central Administration** and click the **Create Site Collections** link.
3. On the **Create Site Collection** page, fill in the details to create a new **Document Center** site.
   1. Make sure you create the new site collection in the web application at **http//intranet.wingtip.com** and give it a title of **Document Center** and an URL of **http://intranet.wingtip.com/sites/DocCenter**.



* 1. Use the **Document Center** site template found in the **Enterprise** tab and assign the user account **WINGTIP\Administrator** as the **Primary Site Collection Administrator**.



* 1. Click the **OK** button to create the new site collection.
  2. Once the site collection has been created, navigate to it and then navigate around the site to see how it differs from a standard team site.



In this exercise, you created a Document Center site and saw that this type of site provides more functionality than a team site for dealing with a large number of documents.

### Exercise 2: Create a Blog Site as a Host Named Site Collection at http://blog.wingtip.com

In this exercise you will create a new Blog site at **http://blog.wingtip.com**. You will also work through the steps to configure this site to support anonymous access. Note that you must create this new blog site as a host named site collection so you cannot create it using Central Administration. Instead, you must create the new site collection using Windows PowerShell.

1. Create a PowerShell script named **CreateBlogSite.ps1** and save it to the lab folder at **C:\Student\Modules\SiteCollections\Lab**. Add code to **CreateBlogSite.ps1** to create a new host header site collection at **http://blog.wingtip.com**.
   1. Begin by loading the SharePoint PowerShell snap-in

Add-PSSnapin Microsoft.SharePoint.PowerShell

* 1. Next, create a variable named **$webapp** and initialize it by calling **Get-SPWebApplication** and obtaining a reference to the web application created by the Farm Configuration Wizard at a base URL of **http://WingtipServer**.

$webapp = Get-SPWebApplication -Identity "http://WingtipServer"

* 1. Create a set of variables to hold the data required to create a new web application using the following code listing.

$siteUrl = "http://blog.wingtip.com/"

$siteTitle = "Wingtip Blog Site"

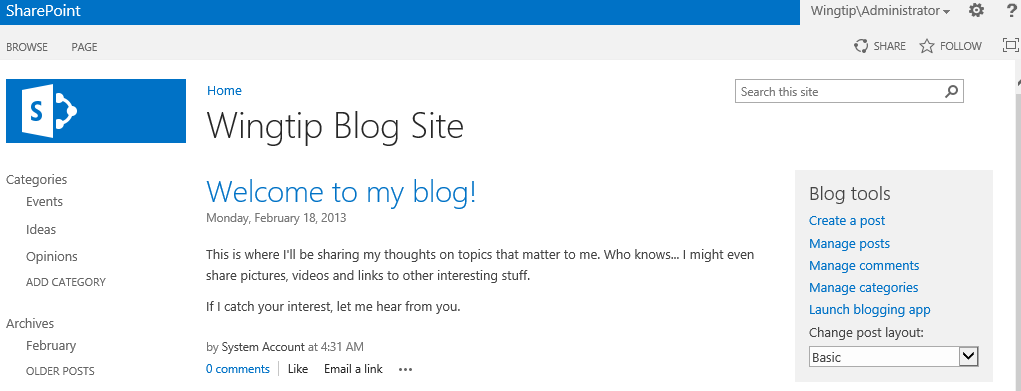
$siteOwner = "Wingtip\Administrator"

$siteTemplate = "BLOG#0"

* 1. Create the new host header site collection using a call to **New-SPSite** as shown in the following listing. Make sure you use the parameter named **–HostHeaderWebApplication** and pass the web application references by the **$webapp** variable as shown in the following listing. Also note the following listing uses line breaks in the parameter list to make the code readable. However, you cannot use line breaks when you add parameters to a cmdlet in an actual PowerShell script.

$site = New-SPSite -Url $siteUrl -Template $siteTemplate -OwnerAlias $siteOwner   
 -Name $siteTitle -HostHeaderWebApplication $webapp

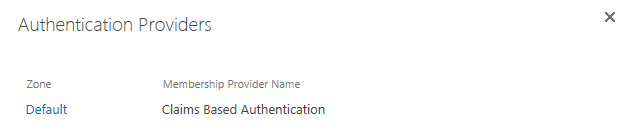
* 1. Once you have create the PowerShell script, execute it to create the new site collection at **http://blog.wingtip.com**.
  2. Test your work by navigating to the site at **http://blog.wingtip.com/** in the browser and making sure it was created properly.



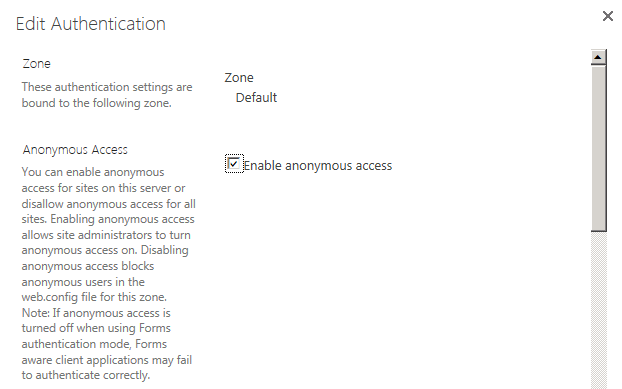
1. Experiment with the blog site by clicking the **Create a post** link and working through the steps to create a blog post.
2. Configure the host web application to support anonymous access.
   1. Navigate to **Central Administration**.
   2. Navigate to the **Manage** **Web Applications** page.
   3. On the **Web Application Management** page, select the web application with the name **SharePoint – 80**.
   4. Click the **Authentication Providers** button on the ribbon to display the **Authentication Providers** dialog.

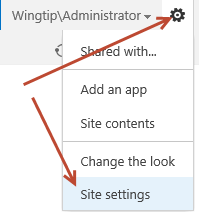


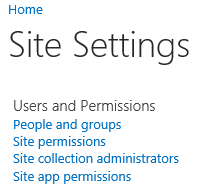
* 1. Inside the **Authentication Providers** dialog you should see a single authentication provider in the **Default** zone. Click on the Default link to display the **Edit Authentication** dialog for this zone.



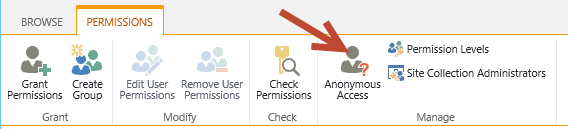
* 1. Inside the **Edit Authentication** dialog, enable the checkbox titled **Enable anonymous access**. Then scroll to the bottom of the dialog and click the **Save** button to save our configuration change for this web application. When the Edit Authentication dialog has closed, close the **Authentication Providers** dialog as well.



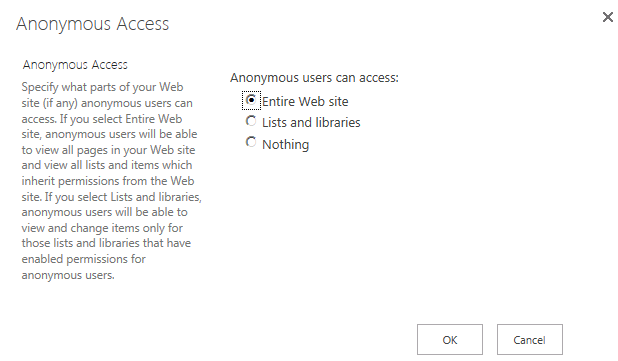
1. Now return to the blog site at **http://blog.wingtip.com**. In the top right hand corner of the screen, use the **Site Settings** menu item in the **Site Actions** menu to navigate to the **Site Settings** page.  
    
2. On the Site Settings page click the **Site permissions** link inside the **Users and Permissions** section.



1. On the Site permissions page, click on the **Anonymous Access** button in the ribbon to display the Anonymous Access dialog.



1. In the **Anonymous Access** dialog, select the option for anonymous users to access the entire web site and click **OK** to save your configuration changes.



1. Now shut down any open instance of the Internet Explorer.
2. Launch a new version of the Internet Explorer and navigate to the site at http://blog.wingtip.com. You should now be accessing the site using anonymous access. You can tell this is the case by looking at the top right corner of the page and making sure you can see the S**ign In** link.



1. Click the sign in link to authenticate and access the site as an authenticated user. If you want to access the site again using anonymous access, you need to shut down the Internet Explorer, launch it again and return to the site.

In this exercise you create a new blog site and configured it to support anonymous access.

### Exercise 3: Create a Publishing Site at http://www.wingtip.com

In this exercise you write a PowerShell script to create a new publishing site which will be accessible through the URL **http://www.wingtip.com**. However, unlike the previous exercise you will configure the new site collection to support anonymous access from within the same PowerShell script that is used to create the site itself.

1. In this exercise, you must create a new publishing site as s a host named site collection. Therefore, you cannot create the new site using Central Administration. Instead, you must create it using Windows PowerShell.
2. Create a PowerShell script named **CreatePublishingSite.ps1** and save it to the lab folder at **C:\Student\Modules\SiteCollections\Lab**. Add code to **CreatePublishingSite.ps1** to create a new host header site collection at **http://www.wingtip.com**.
   1. Begin by loading the SharePoint PowerShell snap-in

Add-PSSnapin Microsoft.SharePoint.PowerShell

Next, create a variable named **$webapp** and initialize it by calling **Get-SPWebApplication** and obtaining a reference to the web application created by the Farm Configuration Wizard at a base URL of **http://WingtipServer**.

$webapp = Get-SPWebApplication -Identity "http://WingtipServer"

Create a set of variable to hold the data required to create a new web application using the following code listing.

$siteUrl = "http://www.wingtip.com/"

$siteTitle = "Wingtip Toys"

$siteOwner = "Wingtip\Administrator"

$siteTemplate = "BLANKINTERNET#0"

Create the new host named site collection using a call to **New-SPSite** as shown in the following listing. Make sure you use the parameter named **–HostHeaderWebApplication** and pass the web application references by the **$webapp** variable as shown in the following listing. Also not the following listing uses line breaks in the parameter list to make the code readable. However, you cannot use line breaks when you add parameters to a cmdlet in an actual PowerShell script.

$site = New-SPSite -Url $siteUrl -Template $siteTemplate -OwnerAlias $siteOwner   
 -Name $siteTitle -HostHeaderWebApplication $webapp

* 1. In the previous exercise you enabled anonymous access on a site collection manually. In this exercise you will enable anonymous access on the new site collection using PowerShell. Add the following PowerShell code to the end of **CreatePublishingSite.ps1**.

# enable anonymous access for site collection

if($site.IISAllowsAnonymous){

Write-Host "Enabling anonymous access"

$web = $site.RootWeb

$web.AnonymousState = [Microsoft.SharePoint.SPWeb+WebAnonymousState]::On

$web.Update()

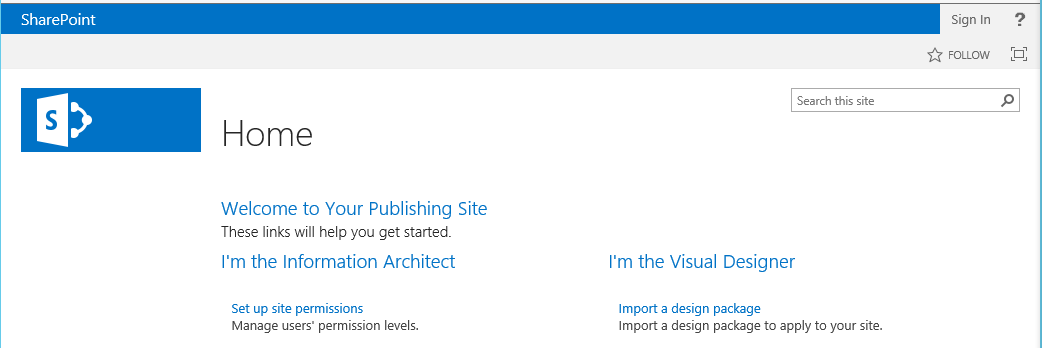
}

else{

Write-Host "Cannot enable anonymous access because Web Application does not allow"

}

* 1. Once you have create the PowerShell script, execute it to create the new site collection at **http://www.wingtip.com**.
  2. Test your work by navigating to the site at **http://www.wingtip.com/** in the browser and making sure it was created properly.



1. Note that you should be accessing the site using anonymous access. You can tell this is the case by looking at the top right corner of the page and making sure you can see the **sign in** link.



1. Click the sign in link to authenticate and access the site as an authenticated user. If you want to access the site again using anonymous access, you need to shut down the Internet Explorer, launch it again and return to the site.
2. If you'd like, spend a little time moving around inside the publishing site and exploring what's inside. If you are familiar with publishing sites from SharePoint 2010, try adding a few child sites and publishing pages to observe the differences in the user experience between SharePoint 2010 and SharePoint 2013.

In this exercise you create a publishing site and enable anonymous access for it using a PowerShell script.

### Exercise 4: Creating a Site Collection in a New Content Database

When you create a new site collection using Central Administration, you do not have very much control over how SharePoint 2013 selects the appropriate content database to host it. If you want to create a new site collection in a specific content database, you have to use PowerShell. In this exercise, you will learn how you can create a new site collection in its own private content database using a PowerShell script.

1. Create a PowerShell script named **CreateEngineeringSite.ps1** and save it to the lab folder at **C:\Student\Modules\SiteCollections\Lab**. Add code to **CreateEngineeringSite.ps1** to create a new content database and to create a site collection inside it which is accessible through the URL of **http://intranet.wingtip.com/sites/engineering**.
   1. Begin by loading the SharePoint PowerShell snap-in

Add-PSSnapin Microsoft.SharePoint.PowerShell

Next, create a new content database named **ContentDB\_WingtipEngineering** which is associated with the web application at **http://intranet.wingtip.com**.

$contentDatabaseName = "ContentDB\_WingtipEngineering"

$webappurl = "http://intranet.wingtip.com"

# create a new content database

New-SPContentDatabase ($contentDatabaseName) -WebApplication $webappurl

Create a set of variable to hold the data required to create a new site collection using the following code listing.

$siteUrl = "http://intranet.wingtip.com/sites/engineering"

$siteTitle = "Wingtip Engineering Site"

$siteOwner = "Wingtip\Administrator"

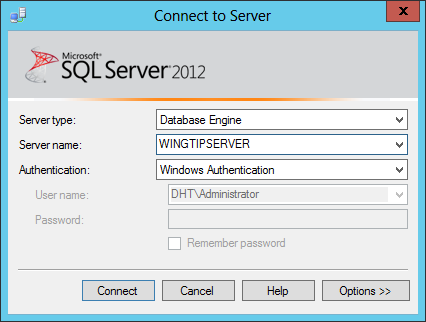
$siteTemplate = "STS#0"

Create the new site collection using a call to **New-SPSite** as shown in the following listing. Make sure you use the parameter named **-ContentDatabase** and pass the name of the content database create earlier in the script. Also note the following listing uses line breaks in the parameter list to make the code readable. However, you cannot use line breaks when you add parameters to a cmdlet in an actual PowerShell script.

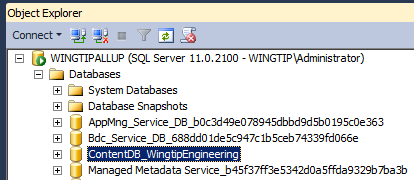
$site = New-SPSite -Url $siteUrl -Template $siteTemplate -ContentDatabase $contentDatabaseName  
 -OwnerAlias $siteOwner -Name $siteTitle

* 1. Save the PowerShell script and execute it to create the new site collection at **http://intranet.wingtip.com/sites/engineering**.
  2. Test your work by navigating to the site at **http://intranet.wingtip.com/sites/engineering/** in the browser and making sure it was created properly.

1. Now use **SQL Server Management Studio** and verify that the new content database has been created.
   1. Click **SQL Server Management Studio**
      1. **Start** » **All Programs** » **Microsoft SQL Server 2012** » **SQL Server Management Studio**.
   2. Select **Database Engine** for server type and **WINGTIPSERVER** for the server name. Leave the Authentication drop-down to **Windows Authentication**. Click **Connect** to log in.



* 1. Expand the Object Explorer tree through **WINGTIPSERVER** » **Databases**. Verify that a database has been created which is named **ContentDB\_WingtipEngineering**.



1. Now inspect the new content database using **SharePoint 2013 Central Administration**.
   1. Navigate to **Central Administration**.
   2. Navigate to **Application Management** page.
   3. Click the **Manage content databases** link to navigate to the **Content Databases** page.
   4. On the right-hand side of the page, use the drop down list to change the select **Web Application** to **intranet.wingtip.com**.
   5. Verify that there is a new content database named **ContentDB\_WingtipEngineering** associated with this web application.



In this exercise you learned how to create a new content database, attach it to a web application and create a new site collection using PowerShell.

### Exercise 5: Install and Begin to use Microsoft Office 2013 SharePoint Designer

In this section you will install Microsoft Office 2013 SharePoint Designer.

1. Ensure you are logged into the **WingtipServer** VM using the account **WINGTIP\administrator | Password1**.
2. Using Windows Explorer, look inside the **C:\Install** and locate the installation file for SharePoint Designer 2013 named **sharepointdesigner\_en-us\_x64.exe**.

If the SharePoint Designer 2013 installation file is not already on your VM, you can download it from the following page:

**http://www.microsoft.com/en-us/download/details.aspx?id=35491**

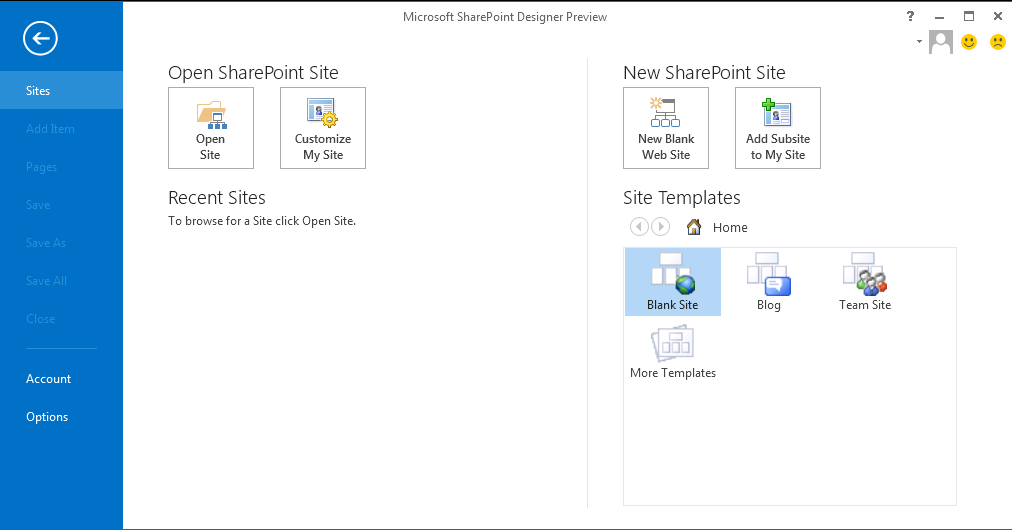
1. Install the **SharePoint Designer 2013**.
   1. Double click to **SharePointDesigner\_English\_x64.exe** to begin the installation.
   2. On the **Read the Microsoft Software License Terms** dialog, click the checkbox next to **I accept the terms of this agreement** and click **Continue**.
   3. On the **Choose the installation you want** dialog, click the **Install Now** button.
   4. The installer will proceed and install all SharePoint Designer 2013.
   5. When the installer completes, click the **Close** button.

At this point you have now installed SharePoint Designer 2013. Over the next few steps you will use this tool to open up the three sites you just created so you can get a feel for how SharePoint Designer 2013 works.

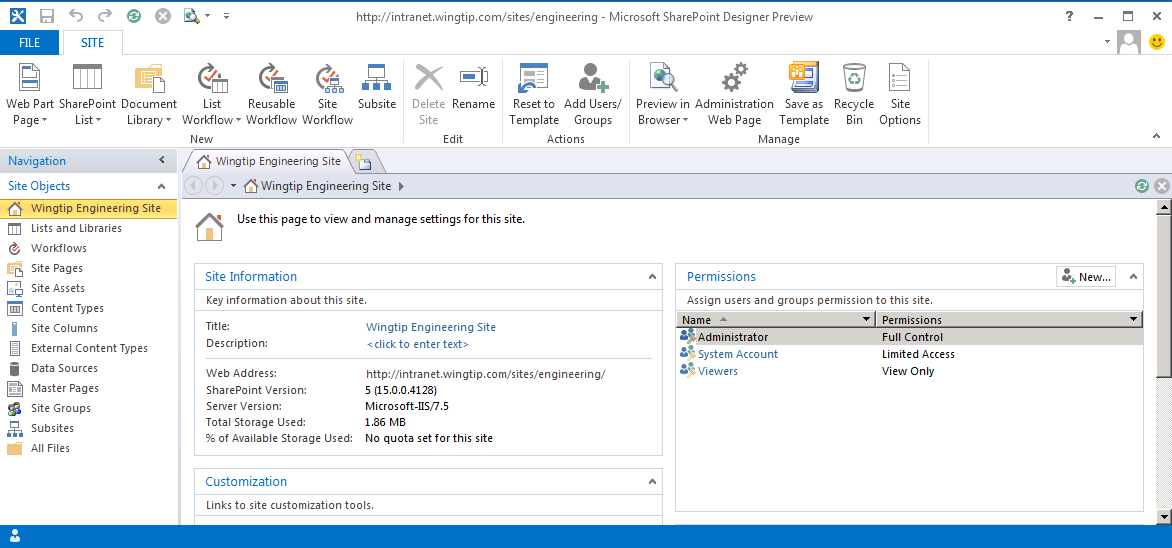
1. **Launch the SharePoint Designer 2013.**
   1. **Press the Windows key to display the Windows Start page.**
   2. **Locate and click the tile for SharePoint Designer 2013 to start this application.**

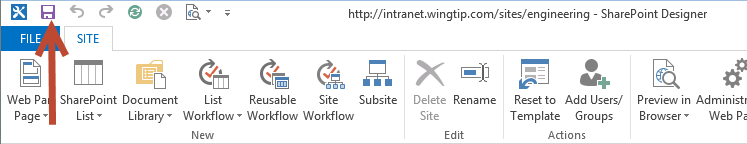
****

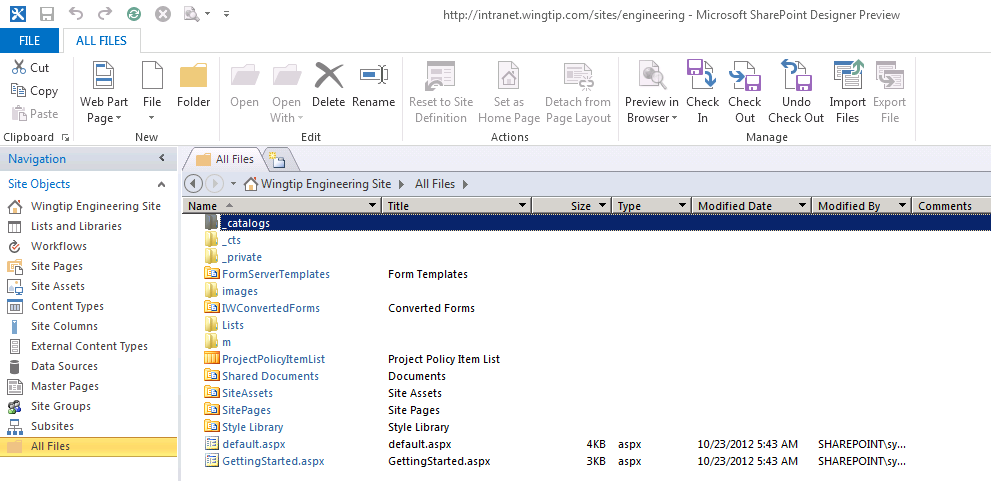
1. At this point SharePoint Designer 2013 should be running but there is no open site yet.



1. Click the **Open Site** button and fill out the URL http://intranet.wingtip.com/sites/engineering.
   1. If you are prompted for login credentials, use **WINGTIP\Administrator | Password1**.
2. **SharePoint** **Designer 2013** opens the site with a **Site Settings** page showing key site information and offers a wide range of possibilities going from links to quick customizations, general settings, site permissions, and sub sites. The ribbon has commands to create artifacts and manage the site as a whole.



1. Take a look at the **Site Information** section.
2. Click the **Wingtip Engineering Site** hyperlink next to **Title**. A text box appears to make it easy on you to change the title of the site. Change the site title from **Wingtip Engineering Site** to **Wingtip Engineering**.
3. Using Internet Explorer, open the Wingtip Engineering site (<http://intranet.wingtip.com/sites/engineering>)
4. Note how the site title is still **Wingtip Engineering Site**. Many changes in SharePoint Designer 2013 are not persisted until they are saved back to the site.
   1. In SharePoint Designer click on the **Save** Button   
      
   2. **Refresh** the Wingtip Engineering site in Internet Explorer. Note how this change has been persisted back to SharePoint.
5. Click the following object in the **Site Objects** navigation   
   pane and inspect what already exists inside a new team site.
   1. Lists and Libraries
   2. Workflows
   3. Content Types
   4. Site Columns
   5. Master Pages
   6. Site Groups
6. Look at the All Files view and inspect the logical collection of folders and files that exist within the virtual file system of this site.



In this exercise you installed the SharePoint Designer 2013 and used it to inspect the contents of a standard team site.