**Title**: How to enable Python on IIS.

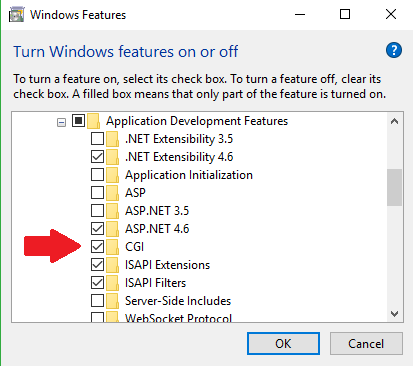
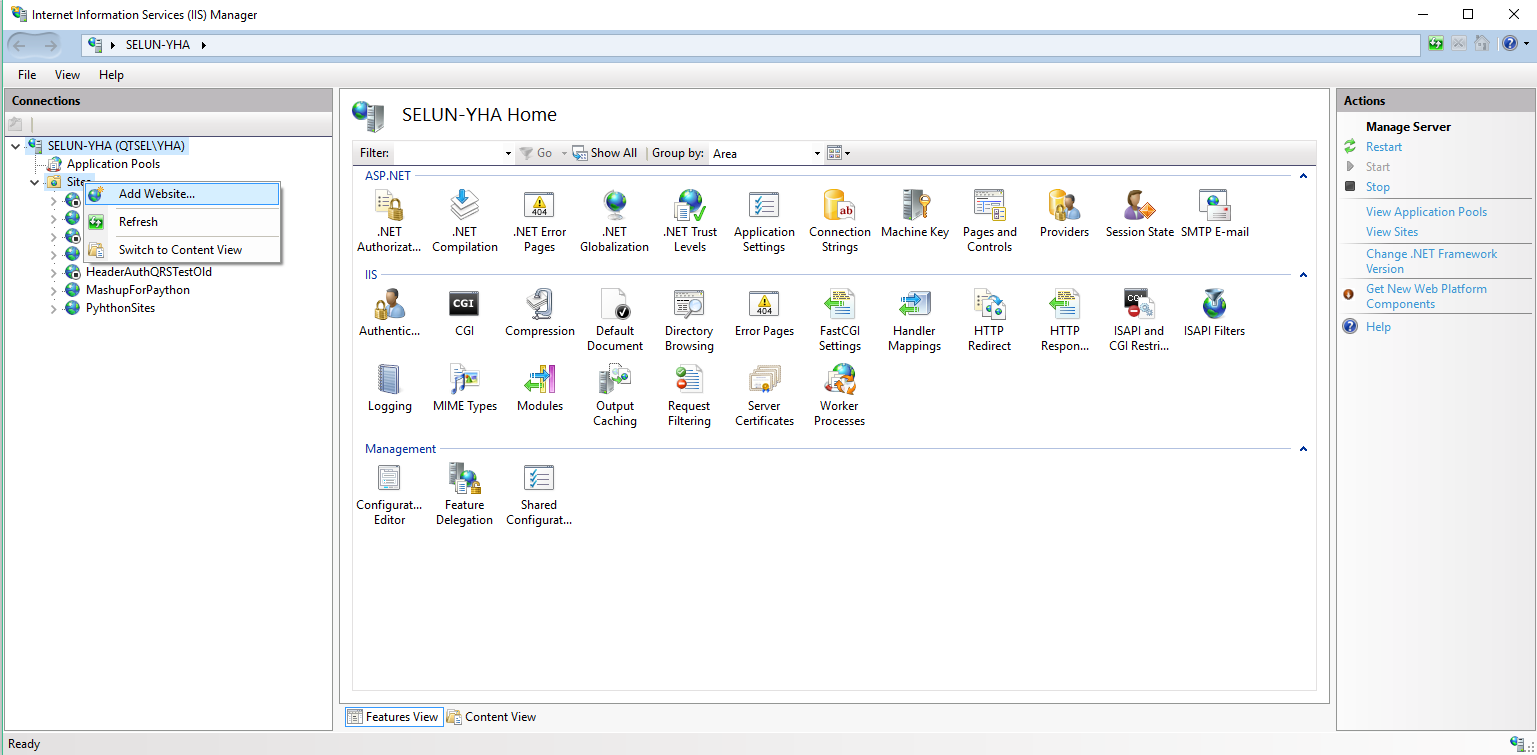
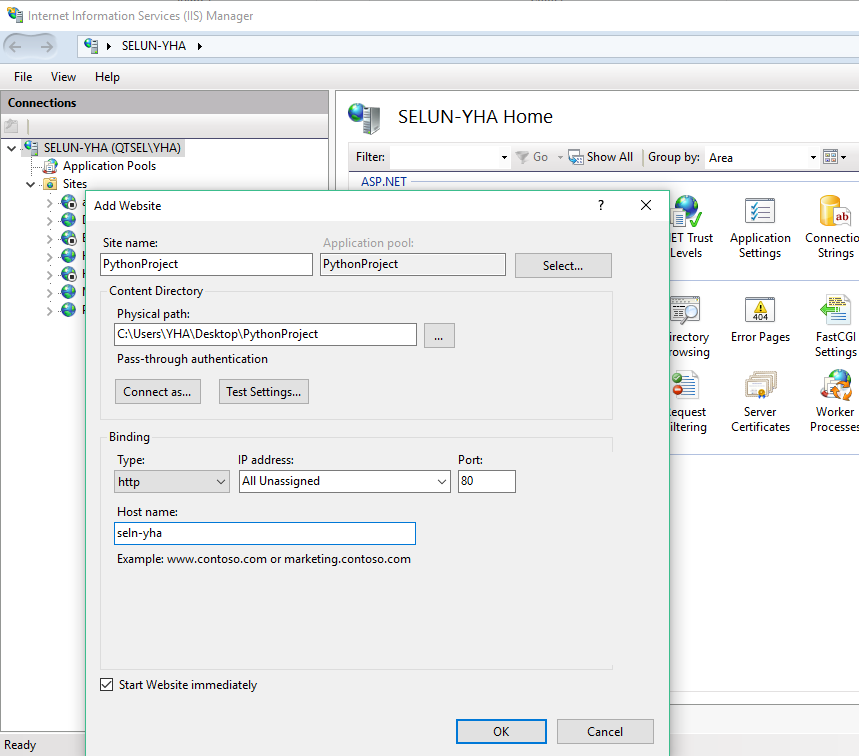
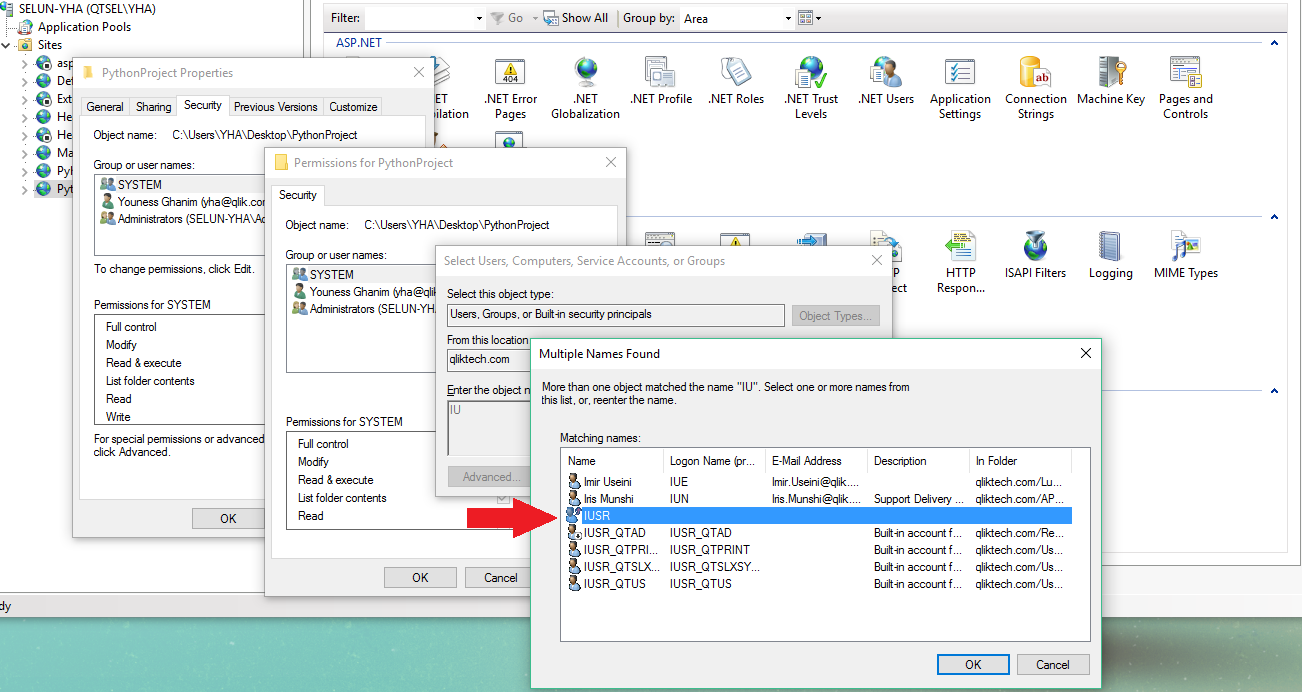
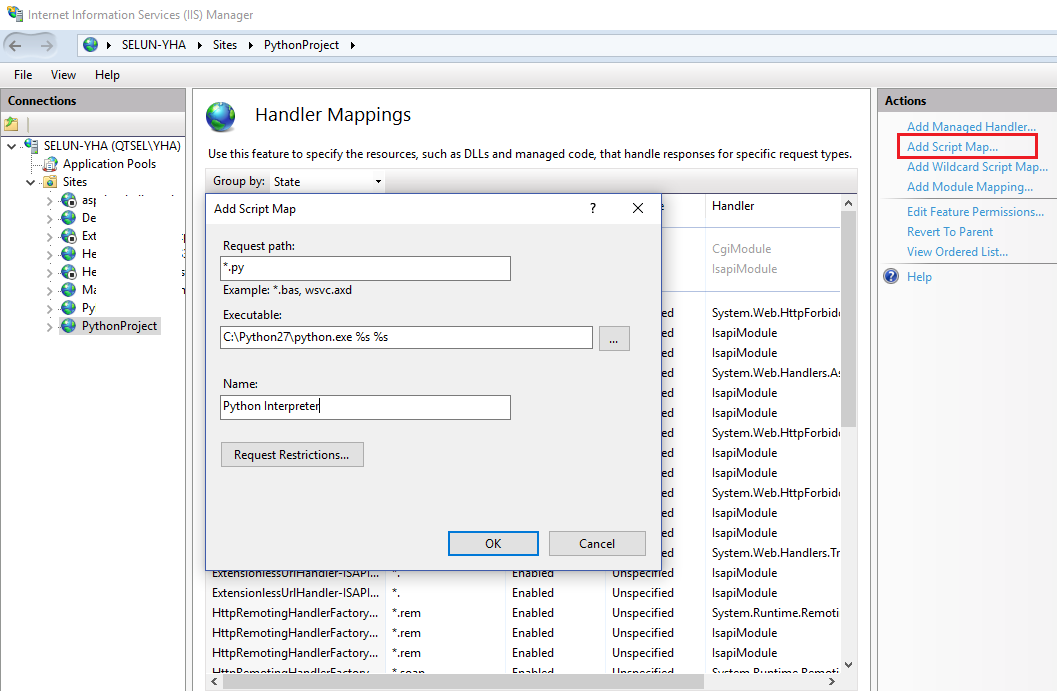
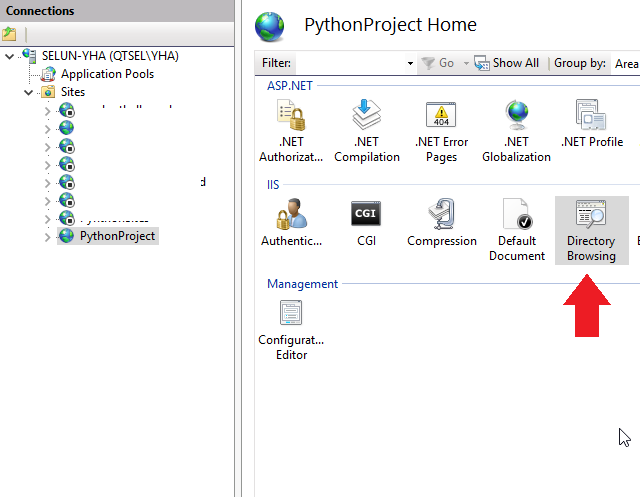
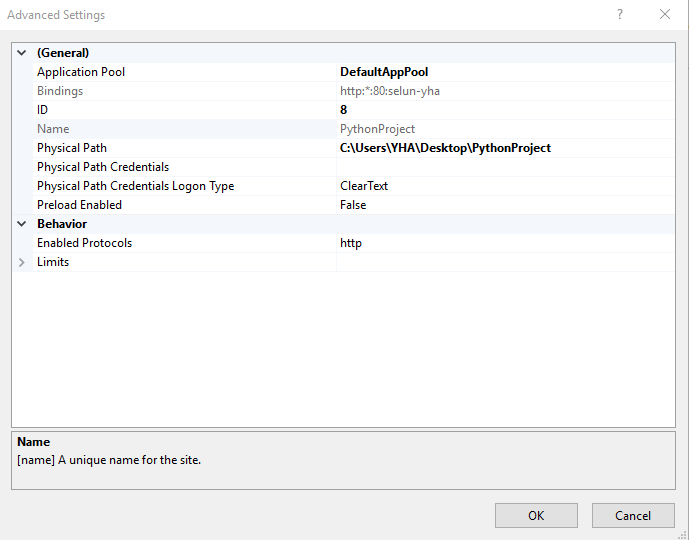
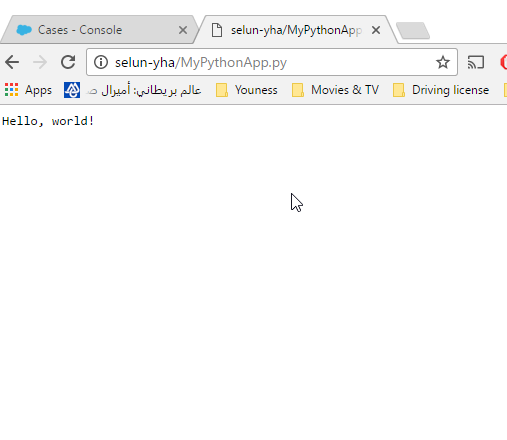
**Summary:**

This article explains how to enable Python on IIS version 1607 for Windows 10. The Python version tested in this example is 2.7.

**Description:**

Sometimes developers may want to make their mashups hosted on a site being added into IIS but can interpret Python code as well. Like in some scenarios a developer may run an Ajax call to query a Python file, but then for this communication to work fine the \*.py file needs to be interpreted by IIS, hence this article will explain how to enable this for IIS, the IIS version used in this test is 1607 running on Windows 10.

**Solution:**

1. Download and install Python form <https://www.python.org/>
2. Install the 64x version if you have a 64-bit machine.
3. Unpack and install the Python installation file, you may choose the C:\ directory so you can easily find the Python exe file later when working with IIS setup. The default installation path anyway will be something like C:\Python27.
4. Make sure the IIS feature is turned on for your Windows OS, you can do this by opening the ***Windows Features*** menu. In the list scroll until you find ***Internet Information Services***. Expand this option and check ***Web Management Tools*** and ***World Wide Web Services > IIS Management Console***. Then press Ok, this will enable IIS on your system.
5. Make sure ***CGI*** is also installed, you can find this under ***IIS > Application Development Features > CGI***. Mark this option.  
     
     
     
   CGI option.png
6. On your desktop create a folder ***PythonProject*** for example and place a Python file in it, in my case I created ***MyPythonApp.py***. All it does is to print a hello world message in the browser.
7. In the windows search bar type IIS, then in the search result you will get the ***Internet Information Services (IIS) Manager*** as the best match. Run it, if you face issues later in the prepress with permissions then re-run IIS with administrator rights. If you never done this before then simply right click on the application and click on ***Run as administrator*** from the pop up menu.
8. Expand the ***Sites*** as you see in the image below and select ***Add Website..***  
     
     
     
   *IIS add a site.png*
9. Fill in the necessary information needed by your site, see image below as a guide:  
     
     
     
   *IIS add site details.png*
10. Right click on your newly added site and set the permissions on the Python project directory and its content. To do that select the ***Edit Permissions***… option from the pop up menu. In the project properties menu go to the ***Security*** tab, press on the ***Edit*** button and then add the following user names (note in my case as an internal person working at Qlik I had to add these ***Group and user names)***: ***IUSR*** and give it ***Full control*** rights in the ***Permissions For IUSR*** section.  
      
      
    IIS site add user 1.png  
      
      
    Then I had to also add ***SELUN-YHA\IIS\_IUSRS*** and give it ***Full control*** rights in the ***Permissions For IIS\_IUSRS*** section.
11. Now we need to map between the site in IIS and the CGI / Python interpreter. The steps for that are:
    1. Select / highlight the site from the list of Sites.
    2. In the UI options to the right, there is an icon under ***IIS***, the icon says ***Handler Mappings***. Double click on it so that a new menu will appear to the left, with a title saying ***Actions***.
    3. Click on ***Add Script Map…*** after this a pop up will appear with some fields you need to fill in.
    4. The ***Request path*** is for specifying the type of files the interpreter will be able to process, in this case we should put ***\*.py*** for Python files.
    5. ***Executable***, this is the path to the Python.exe file (installed in the previous steps in the C:\). In my case, it put ***C:\Python27\python.exe %s %s***
    6. In the ***Name*** field, you can type a name like Python Interpreter. The image below illustrates my setup:   
         
         
         
         
       IIS site add python interp.png
12. Enable ***Directory Browsing*** for your project, this way when the site is launched you can see a listing of the files that are part of your web application and you will be able to click on which one you wish to execute.  
      
      
      
    IIS site enable directory browsing.png
13. One last step before we can begin to see any result, right click on your site, select ***Manage Website > Advanced Settings > Application Pool >*** Set this to ***DefaultAppPool.***  
      
      
    IIS add defaultAppPool.png
14. Now you may check if your site is in the running state. Right click on your site, select ***Manage Website > Browse***, then your default browser will open with a list of files you have in your web project, in this case it is PythonProject as indicated above.
15. Since we want to test Python so from the list simply click on the file with \*.py extension, in this case it is MyPythonApp.py. If everything was configured properly then you will get a result like this:  
      
      
    Files Directory.png
16. Finally, if you click on MyPythonApp.py, you should get the following result in the end:  
      
      
    Hello world as a result.py
17. Source code for the Python application is attached to the article.

**Tags**: IIS, Python, CGI, API