QR Toolbox Installation Guide

Step 1. Download the QR Tool files

1. Download QR Tool files from https://github.com/USEPA/QR_Tool/tree/master by selecting the green "Clone or download" button on the right side, then clicking "Download ZIP".
2. Then, in the desired location, extract the ZIP files contents. You now have all of the files necessary to run the software.

Step 2. Install Python and pip packages

The QR Toolbox requires Python 3+ and some Python packages to run, along with other setup commands. In order to install this software:

1. First locate the Setup.bat file in the QR_Tool folder, then double-click it to run it. The command line and Python installer windows will appear on your screen similar to the example shown below.
2. During the installation of Python, select the option to "Add Python to PATH":
   a. **If you already have Python installed, look at the second screenshot below. Select 'Cancel', and then, when prompted if you are sure, click 'Yes':**
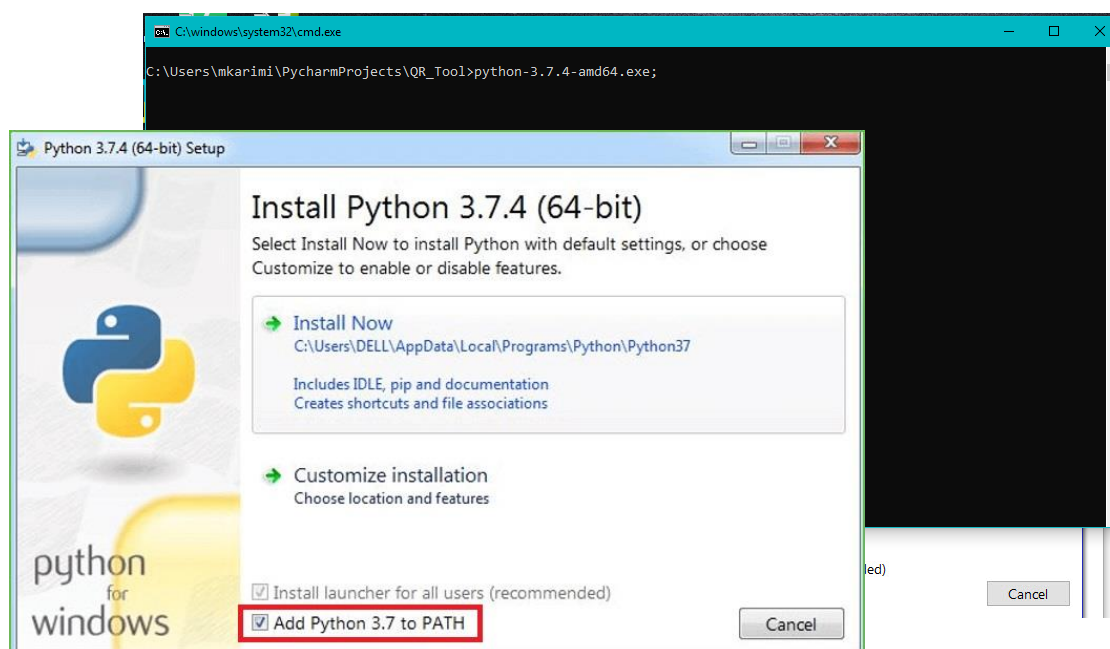

Figure 1 Python Installation Windows – If Python not already installed

Note: EPA users may not have permission to install Python and need to ask IT support to give permission. The installation steps after Python is installed <u>do not</u> require IT support permission.
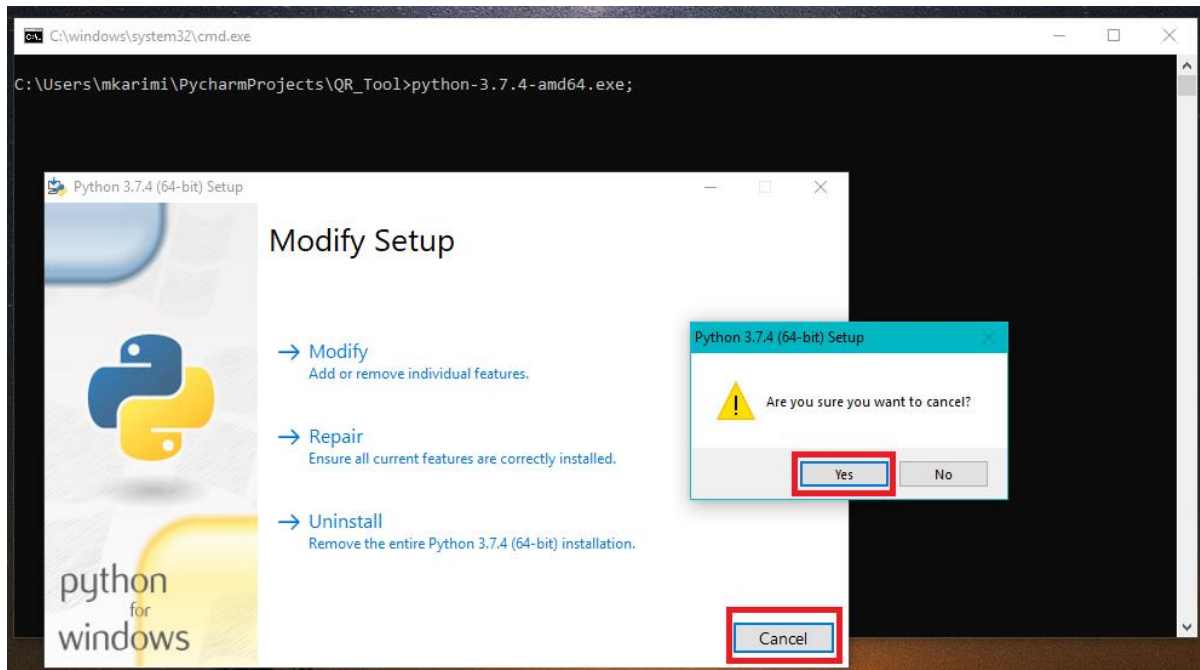
Figure 2 Python Installation Windows - If Python already installed

3. Complete the Python installation by following the prompts from the Python Installer. If you already had Python installed, after clicking 'Yes' the Setup.bat will continue as normal.

Once the Python installation is done, the Setup.bat will automatically install the following required Python packages: pyzbar, imutils, qrcode, Pillow, opencv-python, office365. This should look similar to the image below:
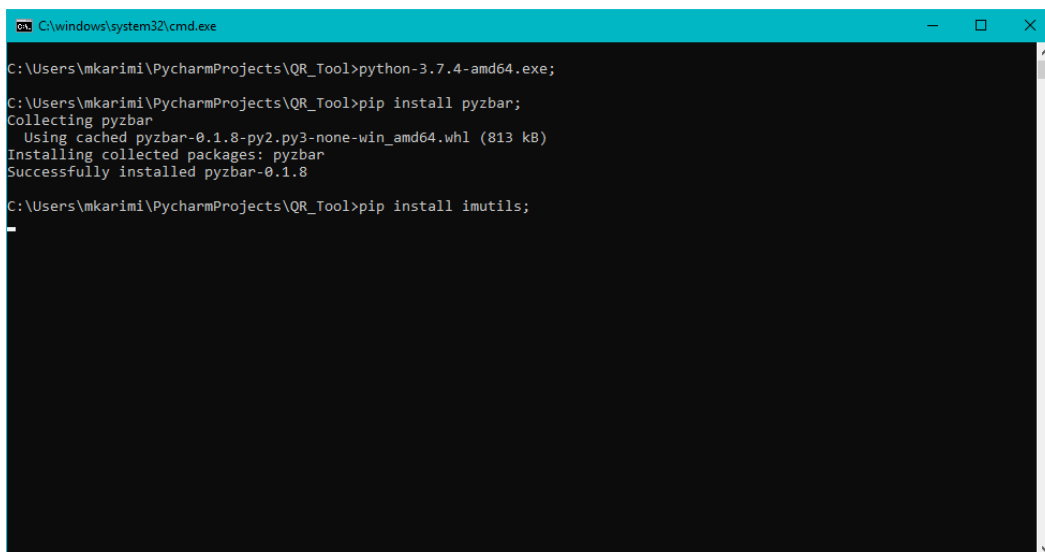

Figure 3 Python Package Installation Window

Note: With Python 3+ installed, 'pip' is also installed automatically.

Step 3. Create Application Key for SharePoint Site
If you need to write a QR Code(s) to a SharePoint Site, you need to create an Application Key for the site so the Python script can access the SharePoint site. You'll need to have admin rights to create application key.

The instructions for setting up the application key are located here:
https://docs.microsoft.com/en-us/sharepoint/dev/solution-guidance/security-apponly-azureacs

Here is the example SharePoint site URL:
https://usepa.sharepoint.com/sites/Emergency%20Response/EOCIncident/

To create an application key, run the URL (example):
https://usepa.sharepoint.com/sites/Emergency%20Response/EOCIncident/_layouts/15/AppRegNew.aspx

The following screen will show:

App Information

The app's information, including app id, secret, title, hosting url and redirect url.

Client Id:

[   ] Generate

Client Secret:

[   ] Generate

Title:

[   ]

App Domain:

[   ]

Example: "www.contoso.com"

Redirect URI:

[   ]

Example: "https://www.contoso.com/default.aspx"

Create   Cancel

Click "Generate" button to generate Client Id and Client Secret. Write down Client Id and Client Secret to use in the next step. Enter a Title that is meaningful for you.
Title: your title
App Domain: www.localhost.com
Redirect URI: https://www.localhost.com

Click "Create" button to create.

Then run the following URL (example):
https://usepa.sharepoint.com/sites/Emergency%20Response/EOCIncident/_layouts/15/AppInv.aspx

App Id and Title

The app's identity and its title.

App Id:

| | Lookup |

Title:

App Domain:

Example: "www.contoso.com"

Redirect URL:

Example: "https://www.contoso.com/default.aspx"

App's Permission Request XML

The permission required by the app.

Permission Request XML:

Enter the Client id and click 'Lookup' button.

In the 'Permission Request XML:' box, enter the following:

```
<AppPermissionRequests AllowAppOnlyPolicy="true">
<AppPermissionRequest Scope="http://sharepoint/content/sitecollection/web"
Right="FullControl"/>
</AppPermissionRequests>
```

Click the 'Create' button to finish.

Step 4. Modify settings.py

Locate and modify the settings.py file using the Client Id and Client Secret generated from the last step and enter the SharePoint site URL for 'url'.

```
settings = {
    'url': '<Your URL>',
    'client_id': '<Your Client Id>',
    'client_secret': '<Your Client Secret>'
}
```

The following lines in QRToolbox.py may need to be modified:

- *listTitle = "QR Timestamps" (this is the SharePoint list name for writing scanned entry to)*
- *qrfolder = "QRCodes" (this is the folder name to store QR Code image files)*
- *bkcsvfolder = "HXWTEST" (this is the folder name to store backup csv files)*
- *qrbatchfile = "names.csv" (this is file name for batch generate QR Code images)*

- *relative_url =*
  "/sites/Emergency%20Response/EOCIncident/EOC%20Documents/QRCodes/names.csv" (this is
  location that names.csv resides)

QR Toolbox User Guide

Run the QR-Toolbox
To run the QR-Toolbox, simply double-click the QR-Toolbox.py to run.

** If the user has two versions of Python installed (Python 2 and Python 3), they may need to specify running QR-Toolbox.py in Python 3.

The following is one way to do that:

1. Create a .bat file to run the QR-Toolbox by opening a new notepad file and saving it as a .bat. Add the following line (all on one line) to the .bat file:
   a. *<python 3 exe location>/python.exe <QR Tool location>/QRToolbox.py*
   b. *Example: C:\Users\<user>\AppData\Local\Programs\Python\Python37\python.exe*
      C:\Users\<user>\Desktop\QR_Tool\QR-Toolbox.py
2. (Optional) Create a shortcut of the .bat file on the desktop by right-clicking and selecting "Create Shortcut", and then dragging that shortcut to the desktop or anywhere else.
3. Double-click the .bat file or the shortcut to run the QR-Toolbox.

Startup Questions
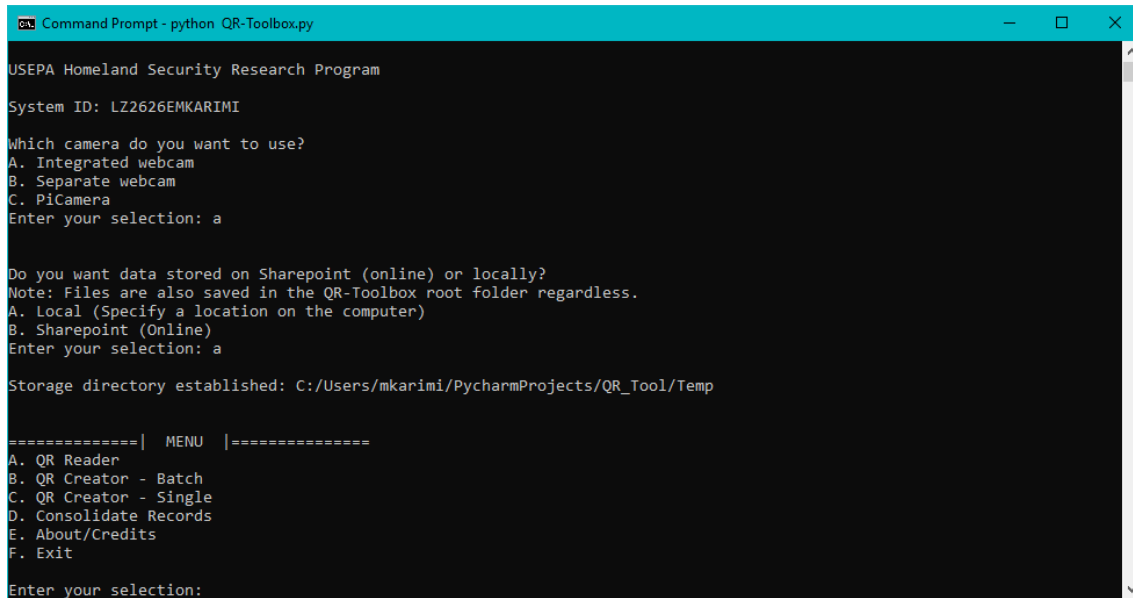There are 2 questions that are asked when the QR-Toolbox is started:

Figure 4 *Startup Questions*

1. Which camera do you want to use?
    a. This question is asking, if the user intends to read QR Codes, which camera should the QR-Toolbox to use. The user can choose from the following options:
        i. Integrated webcam (built in camera, such as in a laptop)
        ii. Separate webcam (camera the user connects via USB port or similar)
        iii. PiCamera (a camera specifically for Raspberry Pis)
    b. *Note: These mappings do assume that the computer has default settings regarding webcams. If those settings are different or it is not a Windows computer that is being used, these mappings above may not be accurate. Example: Plugged in USB Logitech on one laptop, where Option A (should be integrated webcam) was the separate webcam/Logitech, and Option B (should be the separate webcam) was the integrated webcam.*
2. Do you want data stored on SharePoint (online) or locally?
    a. When any data is created by the QR-Toolbox, it is stored in the same folder as the QR-Toolbox.py by default. However, the user has the option of also telling the software to store that data either online on SharePoint, or somewhere else on their computer (locally).

Functions in the QR-Toolbox

There are 6 functions in the QR-Toolbox, each with a local and SharePoint version (the version shown to the user depends on the answer given for the second question):

Figure 5 *Local Version Menu*

Figure 6 *SharePoint Version Menu*

1. QR Reader: Scan the QR Code and the entry will be stored.

    a. SharePoint version: QR Code entry is stored in the same folder as QR-Toolbox.py, and on the SharePoint site.

    b. Local version: QR Code entry is stored in root, and in the location the user specified.

    c. *Notes: Program will reject QR codes created by other sources when it detects a compatibility issue.*

2. QR Creator – Batch: Generate QR Code image file for each entry defined in names.csv.

    a. SharePoint version:

        i. Example entries in names.csv:

            1. *Lastname.firstname@epa.gov*

            2. *FirstName LastName*

        ii. The names.csv must be available in the SharePoint site defined in the QRToolbox.py.

            1. *relative URL = "/sites/Emergency%20Response/EOCIncident/EOC%20Documents/QRCodes/names.csv"*

            2. The user can modify the above line in QRToolbox.py to point to another SharePoint site.

        iii.   *Note: if there are any special characters (ex. 'È', 'è', etc.), the program cannot store these on SharePoint without replacing those characters with their regular equivalents ('è' = 'e'). The program will ask the user if they would like to do that, and if they respond 'No', that text/entry will be skipped. This will occur for every entry that has special characters.*

    b.   Local version:

        i.   Same examples as above

        ii.   The names.csv must be in the same folder as the QR-Toolbox.py file.

        iii.   The generated QR Codes will be stored in the root folder and in the folder the user specified.

3.   QR Creator – Single: Generate QR Code for single entry. This option will prompt to enter email or other ID to generate QR Code image file for the entered value.

    a.   SharePoint version: The QR Code image file will be stored in the SharePoint Site defined in settings.py and the folder name as defined in QR-Toolbox.py: qrfolder = "QRCodes", and also stored in the same folder as QR-Toolbox.py.

        i.   *Note: if there are any special characters (ex. 'È', 'è', etc.), the program cannot store these on SharePoint without replacing those characters with their regular equivalents ('è' = 'e'). The program will ask the user if they would like to do that, and if they respond 'No', that text/entry will be skipped.*

    b.   Local version: QR Code image file stored in the root folder and in the location the user specified.

4.   Consolidate Records (Local version only): This option is used to consolidate records into one file.

5.   About/Credits: Display information about the QR Code Tool.

6.   Exit: Exit the QR Code Tool

Access the QR Records in SharePoint

A SharePoint workflow will automatically aggregate and consolidate the QR data from all input devices.