When a tool does not already exist in ArcGIS Pro to perform a specific and necessary task, the user can easily create the tool to serve the desired function by writing a script. While I am not an experienced Python coder, I am familiar with the ArcGIS Pro interface. Assignment 5 provided experience and practice that taught me how to effectively utilize the Python language to make additions to the ArcGIS pro interface. This assignment taught me how to create a tool to sample a random point(s) in an input dataset in two different ways. The first way was creating a tool(script) in the existing database toolbox. The second way was to create a Python toolbox that contained the required tool ("Random Sampling Tool"). Both methods allow the user to edit and execute the script from a code editor.

In my opinion, the second method is the more efficient way of creating this tool. Creating a Python toolbox that contains the tool all in one script that is easily edited and modified means that you can add multiple tools containing customized parameters and messages for the user interface all contained within the same script. However, I can also see the advantage of having a separate tool script that is not contained within a Python toolbox script. Having a separate script for every tool could make it easier to share these functions with other users if you are collaborating with other users who do not want to edit and execute entire toolboxes.