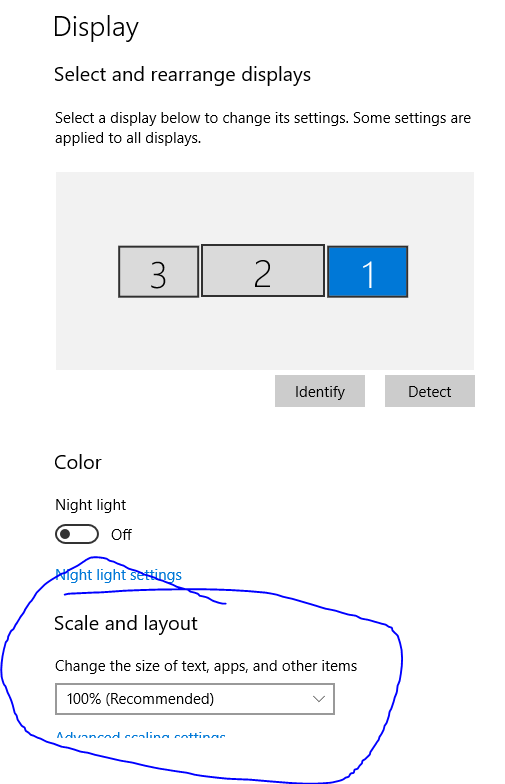
**PIEthon Guide**

Hi welcome to my PIEthon guide. Sorry its not cool with a table of contents and whatever else.

**Purpose of PIEthon**  
PIEthon is a tool that serves two purposes. The first is that is allows you to download data from PIE as a CSV. Pretty much everything in Pie can be downloaded. Or is meant to be downloaded. So there’s that. The other function is to automatically generate reports. This uses a lot of the same code to download the data, but instead of saving it as a file, it is processed and the output tables/graphs are saved as an html. And that’s pretty much it.

**Getting it set up**Grab this whole folder <https://iu.box.com/s/3zy3fg6ddccw9muf7fhfi21wz43jawzx> . This is the working directory for PIEthon. You need privileges for the folder to access it. Email [brfunk@iu.edu](mailto:brfunk@iu.edu) to request it. (NOTE: DO NOT STORE IN BOX LIKE I DID. I AM DUMB THIS IS A BAD IDEA). Okay. So this folder is your working directory. You’ll need to do the following things to ensure it runs correctly. Yes this is a lot of garbage steps but I didn’t anticipate others working on this so here we are. A lot of these problems are thing I will be adding to the list of stuff to do!

1. Change line the variable *pather* on line 14 in ‘functions.py’ to match your directory path for the chromedriver.exe. It is in the include PIEthon directory folder.
2. Make sure your resolution is 1920x1080p. I hard coded all of the form controls for the GUI so it’ll look weird/wrong otherwise. Also make sure the scaling for windows is set to 100%



1. Install the following Python packages.
   1. Selenium
   2. Pandas
   3. PyQt5
   4. numpy
   5. matplotlib
   6. mpld3
   7. Jinja2
2. Run ‘testzone.py’ and pray to the old and new Gods.

**How it works**  
Buckle up goobers cause this is gonna be a fun one. I’m gonna layout the general functionality at a very high level and then move in to more detail in a later section

1. The file that should be run to initialize everything else is ‘testzone.py’. It kicks off with a call to create the login gui
2. The login gui (from ‘loginGui.py’) should then pop up on the screen with text boxes to input your CAS credentials. You can also choose whether you want to run reports (select which one from drop down), or pull data.
3. No matter what you chose, get ready for a DUO push notification. If you don’t have automatic duo pushes on then go do that otherwise you can’t use it lol sorry.
4. If you chose a report (I think only lab breakdown works right now tbh), then everything will run and an HTML output of the report should open automatically. Note that this can take a while because PIE : /
5. If you pull data, a second window will pop up. This is the mainGui window (from ‘mainGui.py’. Here you can select what type of data you want to pull, and apply any optional filters. The calendars change the start and end date and the max returns is the max number of records returned. Click the submit button to go grab the data.
6. The next window has all of the data that was pulled

**PIEthon Backlog (in no particular order --- for now)**

* Make the variable *pather* in ‘functions.py’ not hardcoded. Dynamically change to project directory
* Make all Gui elements dynamic with screen setting (resolution, scaling, etc)
* Push everything to Github instead of Box because yikes
* Change the all data pulls to use the requests package instead of Selenium for better resource management (still would need Selenium to get past Duo though)
* loginGui changes – add option to use Duo call instead of push notifications, let user know if password was correct before sending Duo push message.
* Change output of lab breakdown report to not look like trash – html/css. Maybe switch from using matplotlib to plotly and have a dashboard?
* Get all reports working – add more?
* Make it so PIEdata variable don’t have to be added to list. Instantiate the list with them in there – should take like 10 minutes
* Make it so each report does not have to be manually added to the dropdown menu on the login page. Instead there is something more dynamic in place.
* ‘Select all’ button for the columns in data preview. Also a ‘deselect all’. Just so you don’t manually have to do that when there are 100+ attributes.
* Have all columns be default checked to be included in the export in the preview Gui
* Add input label in preview gui so that the file can be named whatever you want. Maybe make the default name something dynamic too, like ‘Piedata var name-startdate-enddate.csv’ IE ‘Contacts-021316-05318.csv’
* Make all guis look less like garbage. Add some colors, maybe even some images?
* Add confirmation when data export is complete
* Add navigation buttons to guis. Like a back button that reopens the previous gui. Also make it so the mainGui and previewGui don’t open on top of each other. IE one opens when the other is closed.
* Clean up Inventory Report Pie Data Var because it is bad.
* Maybe finish the DMNX handler if you want
* Comment out code to make it as accessible as possible because my lord is it bad.
* Make this documentation also less bad
* Wrap everything up in an executable and make it so updates can be add by installing new version (deletes old)
* Make it so you can hit ‘enter’ on the login screen to submit credentials