* Introduction: Coding Fundamentals practice with Python. Create a console based program that simulates an ATM system. The program must follow basic functions of an actual ATM and the students must have the program running on a Tkinter GUI.
* Learning Objectives
  + Apply coding fundamentals
  + Follow professional standards of syntax and etiquette in program.
  + Utilize version control
  + Test small and often
* Content
  + Basic Boolean logic examples in Python
    - <https://www.geeksforgeeks.org/python-logical-operators-with-examples-improvement-needed/>
    - <https://www.w3schools.com/python/python_operators.asp>
  + Basic List examples in Python
    - <https://www.tutorialspoint.com/python/python_lists.htm>
    - <https://www.geeksforgeeks.org/python-list/>
  + Basic Loop examples in Python
    - <https://www.learnpython.org/en/Loops>
    - <https://www.programiz.com/python-programming/for-loop>
    - <https://www.geeksforgeeks.org/loops-in-python/>
  + Basic Function examples in Python
    - <https://www.w3schools.com/python/python_functions.asp>
    - <https://www.datacamp.com/community/tutorials/functions-python-tutorial>
    - <https://www.tutorialspoint.com/python/python_functions.htm>
  + Basic Tkinter GUI examples
    - <https://www.codemy.com>
    - <https://www.geeksforgeeks.org/python-tkinter-tutorial/>
    - <https://www.tutorialspoint.com/python/python_gui_programming.htm>
    - <https://likegeeks.com/python-gui-examples-tkinter-tutorial/>
* Recommended Pacing Guide
  + Group - 10 minutes of review with the basic content from above
  + Individual - 20 minutes of adding in new feature
* Activities
  + Online – Building out the code
  + Un-plugged – Develop the dialog and ask your partner the questions. Recreate the user experience before coding anything
  + Go through basic list examples
* Assessment
  + You will build your Tkinter ATM throughout these assessments, make sure that each file is called lastnameATMrev#.py
  + Formative 1 – Can the program do the following:
    - Tkinter GUI looks similar to an actual ATM
    - All input must come from the Tkinter window like you would utilize on a touch screen or actual system.
  + Formative 2 – Can the program do the following:
    - Log in to the machine utilizing credit card number and pin
    - Print the user’s information to the screen including name, checking, and savings account information
  + Formative 3 – Can the program do the following:
    - Withdraw
    - Deposit
    - Transfer Balance
    - All 3 functions need to take in parameters and return values while following professional coding standards including
  + Summative 1 – The program must do all of the above plus:
    - Read and Write information to a text file which acts like the database
    - Utilize Classes for each user in the database
    - The program will not crash on any runtime errors
    - Code is professionally commented and developed
* Standards
  + CS
    - CSI-1.1,2,4
    - CSI-2.1,5
    - CSI-3.1,2,3,4,5,6
    - CSI-4.1,2,3,4,5,6,7,8,9,10,11,12,13
    - CSI-5.1,2,3,4,5,6,7
    - CSI-6.1,2,3,4,5,6
    - CSI-7.1,2
  + 1 Non-CS
    - Still Looking
* Scaffolding
  + Calculator.py
  + MediaLibrary.py
  + PLTW material
* Resources
  + <https://www.youtube.com/playlist?list=PLCC34OHNcOtoC6GglhF3ncJ5rLwQrLGnV>
  + <https://www.geeksforgeeks.org/reading-writing-text-files-python/>
  + PLTW material
* Appendix
  + Calculator.py
  + MediaLibrary.py
  + ATMGUI picture
  + exampleATMproject