

**Competency 230 – Python Excel**

*Plan*

**Leader**: L Shumlich

**Date:** 2019/05

**Overview & Purpose**

Python is a server language; one of its major strengths is working with data. In this set of exercises, we will start exploring that strength working with Excel.

The purpose of this exercise is to teach you about structuring a data model, validating input data and merging data sets using python.

**Prerequisite**

* 220 - File IO

**References**

* <https://www.python.org/>
* <https://openpyxl.readthedocs.io/en/stable/tutorial.html>

**Exercise - The word is out**

Do this exercise in teams of two or three. Use only one computer and close the lid of the computers you are not using.

The word is out of your great success in the EvolveU program. Every friend and friend of friends wants you to help them. Many of your friends and new friends are not sure what a full stack developer is but they sure know you can help.

(Big Note… If you have an exercise you would like to substitute for this exercise, you are welcome. It must have at least two, preferably three, subject areas / tables that relate. See Larry first if you are planning to substitute).

You have decided to help Billy. He is just starting a small business. He needs to keep track of his clients, invoices, items sold, along with his product list and inventory. He has asked that you design an excel worksheet for him to last until he is large enough for you to write a custom system for him. All the invoices are done by hand right now and that’s the way he wants to keep it. You just need to design a system for him to keep track of what is going on. He would like you to design the data so that it can be loaded directly into a database. i.e. structure your data model (Excel sheets) using good practices.  Design the data so it is “Normalized”

[Database normalization](https://www.studytonight.com/dbms/database-normalization.php)

Design the spreadsheet so that you can create invoices from the data. There will be 4 worksheets in this design. Design the data so it is “Normalized”. (If you have not seen Larry’s lecture on SQL, ask him). The sheets should be:

* customers (one row for each customer)
* invoices (one row for each invoice)
* invoice line items (one row for each product on the invoice)
* product (one item for each product that you sell)

Create a python program that will relate the data in the 4 tables and create an invoice. The invoice does not need to be fancy. Just start with minimal sample data at first.

**Before programming,**

Document your design. Review with the other groups and pick one of the designs to implement.

Review the rough design (paper / whiteboard / napkin) with Larry. You can review it several times. The design should include the following:

* invoice wireframe
* data model (Excel sheets and their relationships)
* sample data

Development Requirements

* load the data from the worksheet into memory
* use dictionaries to store the customers, invoices, line items, and products
* have your program ask the user for the invoice to print
* as always use the TDD approach

The design should allow for (but do not code or develop reports as follows:

* total invoiced amount to each client
* the invoiced amount each day
* invoices by client

Review with the other groups and pick one of the designs to implement.

**Only develop the code to create an Invoice given an existing invoice ID**. Use the KISS approach. No PANDAS but you can use other libraries to read the data. Hard code or enter an invoice to be generated. Your output can be in the easiest format for you, excel, text file.... Use dictionaries to solve this problem.

**Exercise - Populate the data**

Using the same design and template worksheet as the other students / groups, add more sample data that will represent one month (March – April 2019) of data. Each group takes a different month. There should be:

* 10 - 15 clients (use 5ish clients that are the same as another group)
* 3 - 4 invoices per client 30 invoices
* 1 - 5 items per invoice = 3 items per invoice
* about $15,000 of invoices per month

**Exercise - Validate and Merge the data**

**Validation**

Write python code and use the OpenPyXL package to validate that your input worksheet data is in the right format and reasonable for your system. Validate all the other group’s data as well.  Larry will come and modify your input spreadsheet to try and break your system.  Your validation code should catch these problems.

Think about how to validate your inputs from ANY source, i.e. Excel sheets, tables, etc.  You may not always be loading data from this spreadsheet.  Try to build your validation code to handle ANY input format.

**Merge**

Write python code to merge all the other groups’ data into a single well-formatted workbook that matches your existing data model. If you are the first group, create a second one and merge it.  Validate the new workbook and create some invoices.

https://lh4.googleusercontent.com/fbVu2cAjYdmxFkTQ6MlyxOtbiuqj3URDSV9dX7H9DgHNPv3N842xosfQoevs0KE_niYtGpzHlQ7TFTtdnxfMweSf-f2JchS0UmphvE9_ClZGrhQLWUHZoMadqrGilsULqrt2L-8R