Create various graph and chart for Earning Software with Python

Hello and welcome back, in this chapter we will continue to develop the previous earning application which shows the shoe and shirt sales figure from the input database.

If you want to understand what is going on, then do read the previous post about this topic. In this chapter, I am going to improve the previous application by including a combo box that allows the user to select the type of graph or chart he or she wants to view.

This is the updated version of the user interface program.

Creating a Portable Python Environment from Imports

Python environments provide sandboxes in which packages can be added. Conda helps us deal with the requirements and dependencies of those packages. Occasionally we find ourselves working in a constrained remote machine which can make development challenging. Suppose we wanted to take our exact dev environment on the remote machine and recreate it on our local machine. While conda relieves the package dependency challenge, it can be hard to reproduce the exact same environment.

How to Comment in Python

When writing Python code, it is always a good practice to make your code clean and easily understandable. Organizing the code, giving variables and functions descriptive names are several ways to do this.
Another way to improve the readability of your code is to use comments. A comment is a human-readable explanation or annotation that is used to explain the code.

Mike Driscoll: Python 101 ? Launching Subprocesses with Python

There are times when you are writing an application and you need to run another application. For example, you may need to open Microsoft Notepad on Windows for some reason. Or if you are on Linux, you might want to run grep. Python has support for launching external applications via the subprocess module.

The subprocess module has been a part of Python since Python 2.4. Before that you needed to use the os module. You will find that the subprocess module is quite capable and straightforward to use.

PyCoder?s Weekly: Issue #427 (June 30, 2020)

Python 3.8.4rc1

The Python 3.8 series is the newest major release of the Python programming language, and it contains many new features and optimizations.

Python 3.8.4rc1 is now ready for testing

Assuming no critical problems are found prior to 2020-07-13, the scheduled release date for 3.8.4, no code changes are planned between this release candidate and the final release. That being said, please keep in mind that this is a pre-release and as such its main purpose is testing. Maintenance releases for the 3.8 series will continue at regular bi-monthly intervals, with 3.8.5 planned for mid-September 2020.

 [...] 

The Python 3.8 series is the newest feature release of the Python language, and it contains many new features and optimizations. See the ?What?s New in Python 3.8? document for more information about features included in the 3.8 series.

This is the first bugfix release that is considerably smaller than the previous three. There?s
20% less changes at 130 commits than the average of previous three releases. Detailed information about all changes made in version 3.8.4 specifically can be found in its change log.

- **Episode 6 - Where Does the Data Go?** [9]

  On this episode, we will learn about storing data and how Django manages data using models.

  [...]

  A relational database is like a collection of spreadsheets. Each spreadsheet is actually called a table. A table has a set of columns to track different pieces of data. Each row in the table would represent a related group. For instance, imagine we have an employee table for a company. The columns for an employee table might include a first name, last name, and job title. Each row would represent an individual employee.

- **Unicode in Python: Working With Character Encodings** [10]

  Python's Unicode support is strong and robust, but it takes some time to master. There are many ways of encoding text into binary data, and in this course you'll learn a bit of the history of encodings. You'll also spend time learning the intricacies of Unicode, UTF-8, and how to use them when programming Python. You'll practice with multiple examples and see how smooth working with text and binary data in Python can be!

- **PSF GSoC students blogs: Week 4 Check-in** [11]

- **PSF GSoC students blogs: Week 5 Checkin!** [12]

  This week I worked on the PR. The code was not exactly python ready. So I along with my mentors worked on making the code ready for use in python. Giving PUBLISHED access to exposed functions and members and especially, debugging while compiling the code was challenging. I was stuck many times while compiling the code to make it python ready.