DSC 430: Python Programming
Assignment 0801: Plot Generator

Please look at the DieRollExample on the D2L.

A story generator or plot generator is a tool that generates basic narratives or plot ideas….The tool may allow the user to select elements for the narrative, or it may combine them randomly, a specific variation known as a random plot generator. Such tools can be created for virtually any genre, although they tend to produce formulaic and hackneyed situations. – Wikipedia, 2019

Create a class called **SimplePlotGenerator** that, when queried for a plot, returns "Something happens".

```
>>pg = SimplePlotGenerator()
>>pg.generate()
Something happens
```

Create a class called **RandomPlotGenerator**, which extends SimplePlotGenerator, such that, when queried for a plot, returns a random plot produced from the seven files found on the D2L in the form <plot_names>, a <plot_adjectives> <plot_profesions>, must <plot_verbs> the <plot_adjectives_evil> <plot_villian_job>, <plot_villains>. RandomPlotGeneratormust extend SimplePlotGenerator.

```
>>pg = RandomPlotGenerator()
>>pg.generate()
Aaliyah Mosley, a abiding alabasterer, must acknowledge the
assuming assassin, Acheron Redwood.
```

Create a class called **InteractivePlotGenerator,** which extends SimplePlotGenerator such that, when queried for a plot, offers the user a list of five random plot_names. After the user selects one, the system will offer the user a list of five random plot_adjectives, etc. After the user has made all seven selections, InteractivePlotGenerator should return the final plot.

IMPORTANT NOTE: Give special attention to how you query the user. Nothing in any of the plot generators should assume the form of the I/O. That means no 'print'ing and no 'input'ing. Instead, each of the plot generators should (**create** and) **register a plot viewer**. You will do all **user interactions** through the viewer. When it needs to query the user, it should do something like:

```
result = self.pv.queryUser("Enter a number: ")
```

Ensure to provide a sufficient amount of documentation/comments in the code. Also write answers to these in a comment section at the end of the code file:

- how you randomly produce a plot in **RandomPlotGenerator**.
- how you manage user input in **InteractivePlotGenerator** when you are connected to a view/controller.

Submission: Submit the source file (.ipynb) and the exported html file (.html) to the D2L. Do not zip or archive the file. Your code must include comments at the top including your name, assignment number, and the honor statement, "I have not given or received any unauthorized assistance on this assignment." Also, each function must include a docstring.