**Create character plot with given text files implementing Model-View-Controller (MVC) design pattern.**

A story generator or plot generator allow the user to select elements for the narrative, or it may combine them randomly, a specific variation known as a random plot generator.

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 in the form <plot\_names>, a <plot\_adjectives> <plot\_profesions>, must <plot\_verbs> the <plot\_adjectives\_evil> <plot\_villian\_job>, <plot\_villains>.

RandomPlotGenerator must 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.

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: ”)

Write a class called PlotViewer serves as the both the view and the controller for the plot generator classes. The PlotViewer class will be able to register any plot generator.

>>pv = PlotViewer()

>>pv.registerPlotGenerator( SimplePlotGenerator() )

>>pv.generate() Something happens

>>pv.registerPlotGenerator( RandomPlotGenerator() )

>>pv.generate()

Aaliyah Mosley, an abiding alabasterer, must acknowledge the assuming assassin, Acheron Redwood.

>>pv.registerPlotGenerator( InteractivePlotGenerator() )

>>pv.generate()

Chose a hero’s name from the following list: …

A critical concern is how the generator classes handle input/output. When they are connected to a view/controller they must use the i/o of the view/controller. The generator (model) should not know or care.