Define main

Set keepGoing to True

While keepGoing

Set choice equal to the function getMenuChoice

If choice equals 0

Set keep going to false

Elif choice equals 1

Set game equal to the function getDefaultGame

Print default game loaded

Elif choice equals 2

Set game equal to the function loadGame

print game loaded

Elif choice equals 3

Run the function saveGame

Print game saved

Elif choice equals 4

Set game equal to the function editNode

Elif choice equals 5

Run the function playGame

Define getMenuChoice

Print the menu screen

Set choice equal to input of user’s choice

Return choice

Define getDefaultGame

Set game equal to a one node dictionary

Return game

Define playGame

Set keepGoing equal to true

Set current node equal to start

While keepGoing

If the current node is quit, set keepGoing equal to false

Else, set currentNode equal to the function playnode

Define playNode

Set keepGoing equal to true

While keepGoing

Print the description and both menus

Set menuChoice equal to user input

If menuChoice is equal to 1

Set currentNode equal to nodeA

Set keepGoing equal to false

Elif menuChoice is equal to 2

Set currentNode equal to nodeB

Set keepGoing equal to false

Return currentNode

Define saveGame

With game.json as the outfile

Save the game

Define loadGame

Open game.json

Set game equal to the game file

Return the game

Define editFields

Gather inputs for desc, menuA, nodeA, menuB, nodeB and format them in a list

Return the new node

Define editNode

Print the current nodes and gather input for new node or node to replace

If the nodeToEdit is in the game key list

Print replacing node

Set the node inside game equal to the function editFields

Elif nodeToEdit is not equal to nothing

Print creating new node

Create a new node in game equal to editFields

Return game

Run main