```
      __future__           print_function
        random
        colorama
      colorama          Fore, Back, Style

colorama.init()

# This is the list of words that will be used in the hangman game. They will
# be chosen at random and assigned to the variable "secret"

secretwords = []

secretfile = open("secretwords.txt", "r")
filewords = secretfile.readlines()

    word    filewords:
    word = word.strip("\n")
    secretwords.append(word)

secretfile.close()

alphabet = 'abcdefghijklmnopqrstuvwxyz'
already played =
totalscore = 0
name = ""

    hangman_display(guessed, secret):

    '''

    <--- Hangman display function --->


    Takes arguments "guessed" and "secrets", which are both strings.

    This function takes the "guessed" and "secret" strings and makes a
    comparison as to which letters in the "guessed" string are in the "secret"
    string. If the guessed letters (represented by the variable "letter") are
    in the secret string, that letter is appended to "displaylist", which keeps
    track of the letters to display. If the guessed letter is NOT in the
    "secret" string, a "-" is appended to the displaylist instead. If there is a
    space in the "secret string, the space is not replaced by anything, and
    instead, a space is appended to the displaylist. After the letters of the
    "secret" string have been iterated, a string called "displaystring" will
    obtain the join()-ed list of "displaylist".

    This function returns "displaystring".

    '''

    displaylist = []

        letter    secret:

          letter    guessed:
           displaylist.append(letter)
            letter == " ":
           displaylist.append(letter)
            :
           displaylist.append("-")

    displaystring = ''
    displaystring = displaystring.join(displaylist)

            displaystring

     restart():

    '''

    <--- Game Restart Function --->


    This function takes no arguments.

    This function is called at the end of the game and asks the player if they
    would like to try again. If they say yes, the main game function is called
    again.
```

```
    This function returns no values, but calls the hangman() function.

    '''

          (Fore.YELLOW)
          ("Your score is only saved when you quit the game.")
    restartin = raw_input(Fore.GREEN + "Do you want to restart? (Y/N) ")

      "y"    restartin.lower():
       hangman()

        :
       scoreboard = open("scoreboard.txt", "a+")
       scoreboard.write(name + ": " + str(totalscore) + "pts\n")
            ()
            (Fore.WHITE + "Score saved.")
       scoreboard.close()

            ()
       scoredisp = raw_input(Fore.RED + "Would you like to see the \
scoreboard? (Y/N) ")

         "y"    scoredisp.lower():
                (Fore.CYAN)
          scoreboard = open("scoreboard.txt", "r")

                (scoreboard.read())
          scoreboard.close()


     hangman():

    '''

    <--- Main Game Function (Hangman) --->


    This function takes no arguments.

    This is the main hangman function. In this function, the player goes into
    a loop which allows them to make their guesses. The guess is put into the
    variable "guessed", which becomes an argument in the display_list()
    function. The loop then analyzes the result and makes sure that the input
    is valid, then calls display_list(). The loop continues while tries is not
    0.

    This function returns no values, but calls restart() at the end.

    '''

          alreadyplayed
          totalscore
          name

    score = 0

      alreadyplayed ==       :

            (Fore.WHITE)
            ("Welcome to Computer Science Hangman!")
            ()
       name = raw_input("What is your name?: " + Fore.YELLOW)
            (Fore.WHITE)
            ("Welcome, " + Fore.YELLOW + name + Fore.WHITE + "!")
            ()
       play = raw_input("Press enter to continue... ")

    maximumtries = 8
    tries = maximumtries
    recentguessed = ''
    guessed = ''
    secret = random.choice(secretwords)

          (Fore.WHITE)
          ("<---- " + Fore.YELLOW + "WELCOME TO HANGMAN! "+Fore.WHITE+ "---->")
          (Fore.YELLOW)
          ("How to play:")
          (Fore.RED + "Type letters to guess them!")
          ("Type the secret word if you think you've got it!")
```

```
        (Fore.BLUE)
        ("Your score is calculated by multiplying (tries left) by (length of \
the secret word).")
        ("Your scores are saved under your name ONLY when you quit the game.")
        (Fore.YELLOW)
        (hangman_display(guessed, secret))

         tries > 0:

            (Fore.GREEN)
            ("Guess the secret word!")
            ("Tries left = " + Fore.RED + str(tries))


      # developer mode
      #print()
      #print("Answer =", Fore.WHITE, secret)

            (Fore.BLUE)
      recentguessed = raw_input("Type a letter or guess the word here: "
+ Fore.WHITE)
      recentguessed = recentguessed.lower()

        recentguessed == secret:

              ()
              (Fore.YELLOW + "You got it!")


           recentguessed == "":

              ()
              ("Please actually type something in, mate.")

           len(recentguessed) > 1:

            recentguessed == recentguessed.upper():
                  ()
                  ("There's no need to yell.")

            len(recentguessed) == len(secret):
                  ()
                  ("Seems incorrect to me.")

              len(recentguessed) > 3:
                  ()
                  ("There's no need to spam.")

              :
                  ()
                  ("One letter at a time, please.")

          recentguessed = ""

          recentguessed         alphabet:

              ()
              ("That's not even part of the alphabet...")

          recentguessed    guessed:

              ()
              ("You already guessed that.")
          tries = tries + 1

          recentguessed         secret:

              ()
              ("Incorrect!")

          recentguessed    secret:

              (Fore.CYAN)
              ("You got a letter!!")
          tries = tries + 1

      guessed = guessed + recentguessed
            (Fore.YELLOW)
            (hangman_display(guessed, secret))
```

```
238             ()
239
240         # Increment tries
241         tries = tries - 1
242
243           "-"         hangman_display(guessed, secret):
244
245                   (Fore.YELLOW + "You Win!")
246
247
248     score = tries*len(secret)
249     totalscore = score + totalscore
250
251       tries == 0:
252
253             ("You lose!")
254
255         ("*************************")
256
257
258         (Fore.RED)
259         ("Points gained:" + Fore.WHITE, score)
260         (Fore.RED + name + "'s total score:" + Fore.WHITE, totalscore)
261     already played =
262
263     restart()
264
265 ''' <---- Call game functions ----> '''
266
267 hangman()
```