text.txt是03.py所使用的文件、 README.md是10.py所使用的文件

我製作的LinkedList.py，可以任意更改text.txt裡面的內容，可以新增、修改刪除，都是以列為單位，使用課本上的物件導向(如下):

while current != None and current.row <= ptr.row: # 把ptr(目前文字)放入鏈結裡面  
 prev = current # 走到下一個節點  
 current = current.next  
# 建立鏈結  
ptr.next = current # TextRow().next = None  
prev.next = ptr # head.next = TextRow()

建立一個鍊結，頭尾相連

可以使用while迴圈，把一個一個叫出來:

while current != None:  
 print('%-10d %-15s' % (current.row, current.context,))  
 count = count + 1  
 current = current.next # 前進到下一個節點

再加上python 裡面的 open功能，對存在文件進行一系列操作，一開始程式會直接讀取txt檔案裏面的內容。

如果對程式裡面的資料做修改，就利用open("text" + ".txt", 'r+')去修改檔案

如下:

def text\_writer():  
 file = open("text" + ".txt", 'w') # w 會 over\_write, r+會疊加字的內容, r 只能讀不能寫  
 text\_dict = {}  
 pre\_k = 1  
 text = ""  
 current = head.next  
 while current != None: # 走過鏈結  
 text\_dict[current.row] = current.context  
 current = current.next  
  
 for k, v in text\_dict.items():  
 for i in range(k - pre\_k):  
 text += "\n"  
 text += v  
 pre\_k = k  
  
 file.write(text)  
 file.close()

可以讀取文件的資料，把文字和行數(使用\n判斷)放到字典裏面，之後再丟到結點裡面(class TextRow)

詳細的python程式詳解:

# 鏈結串列 -- 加入、刪除、修改及輸出  
# File Name: SingleLinkedList.py  
# Version 4.0 (updated on May 8, 2021)  
  
import sys  
  
  
def text\_writer():  
 file = open("text" + ".txt", 'w') # w 會 over\_write, r+會疊加字的內容, r 只能讀不能寫  
 text\_dict = {}  
 pre\_k = 1  
 text = ""  
 current = head.next  
 while current != None: # 走過鏈結  
 text\_dict[current.row] = current.context  
 current = current.next  
  
 for k, v in text\_dict.items():  
 for i in range(k - pre\_k):  
 text += "\n"  
 text += v  
 pre\_k = k  
  
 file.write(text)  
 file.close()  
  
def add\_text\_to\_link():  
 file = open("text" + ".txt", 'r+')  
 s = file.read(999)  
 context = ""  
 count = 0  
 for i in s:  
 if i == "\n":  
 count += 1  
 list\_insert(count, context)  
 context = ""  
 else:  
 context += i  
 count += 1  
 list\_insert(count, context)  
  
  
  
class TextRow:  
 def \_\_init\_\_(self):  
 self.row = 0  
 self.context = ""  
 self.score = 0  
 self.next = None  
  
  
head = TextRow()  
head.next = None  
  
  
# 按照分數的高低加入  
def insert\_f():  
 ptr = TextRow()  
 ptr.next = None  
 prev = head  
 current = head.next  
 while True:  
 try:  
 ptr.row = eval(input('加入行數: '))  
 ptr.context = input('輸入文字內容: ')  
 break  
 except Exception:  
 print("輸入錯誤!!")  
  
 while current != None and current.row >= ptr.row:  
 if ptr.row == current.row or ptr.row <= 0:  
 if (ptr.row <= 0):  
 print("輸入 列 必須大於0")  
 else:  
 print("此行有文字，請重新輸入")  
 while True:  
 try:  
 ptr.row = eval(input('加入行數: '))  
 ptr.context = input('輸入文字內容: ')  
 prev = head  
 current = head.next  
 break  
 except Exception:  
 print("輸入錯誤!!")  
 else:  
 prev = current # 走到下一個節點  
 current = current.next  
  
 # ptr.score = eval(input('Student score: '))  
 print()  
  
 while current != None and current.row <= ptr.row: # 把ptr(目前文字)放入鏈結裡面  
 prev = current # 走到下一個節點  
 current = current.next  
 # 建立鏈結  
 ptr.next = current # TextRow().next = None  
 prev.next = ptr # head.next = TextRow()  
  
  
def list\_insert(row, context):  
 ptr = TextRow()  
 ptr.next = None  
 ptr.row = row  
 ptr.context = context  
 prev = head  
 current = head.next  
 while current != None and current.row <= ptr.row:  
 prev = current  
 current = current.next  
 ptr.next = current  
 prev.next = ptr  
  
  
# 刪除某一特定的節點  
def delete\_f():  
 if head.next == None:  
 print(' 文件中無資料\n')  
 else:  
 del\_row = eval(input('要刪除的列數: '))  
 prev = head  
 current = head.next  
 while current != None and del\_row != current.row:  
 prev = current  
 current = current.next  
 if current != None:  
 prev.next = current.next  
 current.next = None  
 print('\n 已經刪除 第%d列 \n' % (del\_row))  
 else:  
 print('\n 沒有找到 第%d列 \n' % (del\_row))  
  
  
# 修改某一節點的分數  
def modify\_f():  
 if head.next == None:  
 print(' 文件中無資料\n')  
 else:  
 modify\_row = eval(input(' 需要修改的列: '))  
 prev = head  
 current = head.next  
 while current != None and modify\_row != current.row:  
 prev = current  
 current = current.next  
 if current != None:  
 while True:  
 try:  
 print(' 修改列數: %s' % (current.row))  
 print(' 文字內容: %s\n' % (current.context))  
 break  
 except Exception:  
 print("輸入錯誤!!")  
  
 # 先把舊的資料刪除  
 prev.next = current.next  
 current.next = None  
  
 # 再重新加入新的資料  
 new\_text = input(' 請輸入修改文字內容: ')  
 list\_insert(current.row, new\_text)  
 print('成功修改資料 !\n')  
 else:  
 print('\n沒有找到 第%d列 \n' % (modify\_row))  
  
  
# 顯示鏈結串列的所有節點資料  
def display\_f():  
 count = 0  
 if head.next == None:  
 print(' 文件中無資料\n')  
 else:  
 print('%-10s %-15s' % ('Row', 'Context'))  
 for i in range(32):  
 print('-', end='')  
 print()  
 current = head.next  
 while current != None:  
 print('%-10d %-15s' % (current.row, current.context,))  
 count = count + 1  
 current = current.next # 前進到下一個節點  
 for i in range(32):  
 print('-', end='')  
 print()  
 print('Total %d record(s) found\n' % (count))  
  
  
def main():  
 add\_text\_to\_link()  
 while True:  
 print('\n\*\*\*\*\*\* Single list operation for editing text.tx \*\*\*\*\*\*')  
 print(' <1> 插入text.tx檔案裡面的文字列 ')  
 print(' <2> 刪除text.tx檔案裡面的文字列 ')  
 print(' <3> 修改text.tx檔案裡面的文字列 ')  
 print(' <4> 顯示text.tx檔案裡面的內容 ')  
 print(' <5> Exit ')  
 print('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')  
  
 try:  
 option = int(input(' Choice : '))  
 except ValueError:  
 print('Not a correct number.')  
 print('Try again\n')  
  
 print()  
 if option == 1:  
 insert\_f()  
 elif option == 2:  
 delete\_f()  
 elif option == 3:  
 modify\_f()  
 elif option == 4:  
 display\_f()  
 elif option == 5:  
 sys.exit(0)  
 else:  
 print('不正確的選項')  
 text\_writer()  
  
  
main()