

Trabajo práctico N°2 - Python

Para todos los ejercicios: siempre intentar dividir el problema en subproblemas para modularizar el código.

1. Escribir un procedimiento “reverso” que permita ingresar como parámetro una cadena, y devuelva la cadena invertida (“hola” se convierte en “aloh”). Escribir luego un programa que determine si una cadena de caracteres es un palíndromo (un palíndromo es un texto que se lee igual en sentido directo y en inverso, ej.: “radar”). Sugerencia: para evitar diferencias entre mayúsculas y minúsculas en las cadenas, utilice la función del tipo string `.upper()` ó `.lower()` en las cadenas, ya que Radar es distinto a radaR.
2. Escriba una función llamada `EsBisiesto` que permita ingresar un número de año y devuelva verdadero en caso que el año sea bisiesto, o falso cuando no lo es. Un año es bisiesto si: es divisible entre cuatro y (no es divisible entre 100 o es divisible entre 400). Utilizarlo en un programa que permita ingresar día, mes y año y muestre por pantalla si la fecha es válida o no.
3. Escriba un programa que permita cargar las notas de exámenes, primero debe permitir ingresar por teclado la cantidad de notas que desea cargar, y luego cargarlas en una lista, y posteriormente debe buscar la nota más alta, mostrarla, e indicar en qué índice del arreglo se encuentra.
4. Escriba un programa que permita ingresar un número, se debe validar que realmente se haya ingresado un número, y crear una lista para almacenar por separado los dígitos del número. Luego recorrer la lista y mostrar el índice que contiene el dígito mayor.
5. Escriba un programa que permita cargar una lista de alumnos junto con su nota del parcial. Seleccione la estructura de datos que mejor se adapte al problema. Luego de ingresados los datos debe generar una lista donde figure si aprobaron o no (se aprueba con 40 o más). El listado a mostrar por pantalla debe ser como el siguiente (el resultado no se almacena, se calcula):

ALUMNOS	PARCIAL	RESULTADO
Smith, Juan	70	Aprobado
Suárez, María	35	Desaprobado

6. Escriba un programa que permita leer de un archivo distancias.txt (cada renglón tiene una distancia válida) las distancias recorridas por el vehículo de una empresa, luego calcular cual es la distancia promedio, y mostrar por pantalla el promedio y todas las distancias mayores al promedio.

Ej del contenido del archivo:

150

120

50

34

250

Salida: "La distancia promedio de los viajes es ... y los viajes con distancia mayor son: ... , ... , , "

7. Un almacén guarda los códigos, los nombres de los productos y sus precios, respectivamente, separados por punto y coma (;) en el archivo productos.txt. Hacer un algoritmo y luego los procedimientos necesarios que permitan tomar los datos del archivo y buscar el precio de un artículo ingresado por teclado. Para probar el algoritmo crear un archivo "productos.txt" y cargarle datos al estilo:

100;arroz;10

102;fideos;5

135;lentejas;8

138;porotos;6

140;sal gruesa;5

201;aceite;20 (etc...)

8. El mismo almacén del punto anterior almacena los datos del stock de productos en el archivo stock.txt separados por punto y coma (;) con el formato "codigo de producto; stock mínimo; stock real". Escriba un programa, que a partir de información contenida en los archivos, genere otro archivo de texto, Compras.txt, conteniendo todos los productos cuyo stock se encuentra por debajo del mínimo. Utilizar el archivo productos.txt del punto anterior, y crear un archivo stock.txt y cargarle datos utilizando los códigos de los productos del archivo anterior. Ej:

100;50;60

102;50;20

135;20;15

138;20;20

140;10;8

201;20;30 (etc...)

9. Un profesor almacenó los datos de los alumnos de su materia en un archivo `alumnos.txt`. En cada línea guardó el número de legajo del alumno y sus tres notas finales (oral, escrito y trabajos prácticos). El archivo está ordenado por número de legajo.

En otro archivo, ordenado alfabéticamente por apellido, guarda por línea, número de legajo, apellido y nombre de cada uno.

En ambos archivos los datos están separados por punto y coma (;) .

Desea escribir un programa para generar un archivo `Promoción.txt` con los apellidos y nombres de los alumnos que promocionan la materia, esto es, alumnos que el promedio de las tres notas supere los 7 puntos.

El archivo debe quedar ordenado alfabéticamente