

## Unit 2

# Python Essentials: an object oriented approach

### 2.1 Exercises

**Exercise 2.1.** Un archivo de texto contiene algo de texto (nada inusual) pero necesitamos saber con qué frecuencia aparece cada letra en el texto. Tal análisis puede ser útil en criptografía, por lo que queremos poder hacerlo en referencia al alfabeto latino. Escribe un programa que:

- Pida al usuario el nombre del archivo de entrada.
- Lea el archivo (si es posible) y cuente todas las letras latinas (las letras mayúsculas y minúsculas se tratan como iguales).
- Imprima un histograma simple en orden alfabético (solo se deben presentar recuentos distintos de cero).

Crea un archivo de prueba para tu código y verifica si tu histograma contiene resultados válidos. Suponiendo que el archivo de prueba contiene solo una línea con:

```
aBc
```

el resultado esperado debería verse de la siguiente manera:

```
a -> 1
b -> 1
c -> 1
```

**Exercise 2.2.** El código anterior necesita ser mejorado. Está bien, pero tiene que ser mejor. Realiza algunas enmiendas, que generen los siguientes resultados:

- El histograma de salida se ordenará en función de la frecuencia de los caracteres (el contador más grande debe presentarse primero).

- El histograma debe enviarse a un archivo con el mismo nombre que el de entrada, pero con la extensión '.hist' (debe concatenarse con el nombre original).

Suponiendo que el archivo de prueba contiene solo una línea con:

```
cBabAa
```

El resultado esperado debería verse de la siguiente manera:

```
a -> 3
b -> 2
c -> 1
```

**Exercise 2.3.** El profesor Jekyll dirige clases con estudiantes y regularmente toma notas en un archivo de texto. Cada línea del archivo contiene 3 elementos: el nombre del alumno, el apellido del alumno y el número de puntos que el alumno recibió durante ciertas clases.

Los elementos están separados con espacios en blanco. Cada estudiante puede aparecer más de una vez dentro del archivo del profesor Jekyll. El archivo puede tener el siguiente aspecto:

John	Smith	5
Anna	Boleyn	4.5
John	Smith	2
Anna	Boleyn	11
Andrew	Cox	1.5

Tu tarea es escribir un programa que:

- Pida al usuario el nombre del archivo del profesor Jekyll.
- Lea el contenido del archivo y cuenta la suma de los puntos recibidos para cada estudiante.
- Imprima un informe simple (pero ordenado), como este:

Andrew Cox	1.5
Anna Boleyn	15.5
John Smith	7.0

Nota:

- Tu programa debe estar completamente protegido contra todas las fallas posibles: la inexistencia del archivo, el vacío del archivo o cualquier falla en los datos de entrada; encontrar cualquier error de datos debería causar la terminación inmediata del programa, y lo erróneo deberá presentarse al usuario.

- Implementa y usa tu propia jerarquía de excepciones: la presentamos en el editor; la segunda excepción se debe generar cuando se detecta una línea incorrecta y la tercera cuando el archivo fuente existe pero está vacío.

**Exercise 2.4.** Escriba en un archivo una cierta cantidad de notas generadas de forma aleatoria. Luego calcule y genere la varianza y la desviación estándar de los números en el archivo además de su promedio. Si  $\bar{x}$  denota el promedio de  $x_1, x_2, \dots, x_n$ , la **varianza** es el promedio de los cuadrados de las desviaciones de los números del promedio:

$$var(x_1, \dots, x_n) = \frac{1}{n} \cdot \sum_{j=1}^n (x_j - \bar{x})^2.$$

y la **desviación estándar** es la raíz cuadrada de la varianza.