

Instrucciones

- Lea con detenimiento y desarrolle individualmente cada una de las actividades a realizar durante la experiencia.
- Cree un archivo con extensión **.py** con lo desarrollado. El nombre del archivo debe tener el siguiente formato: **TEL101_P5_Nombre_Apellido.py** (Ej. TEL101_P5_Nicolas_Galvez.py), sin incluir tildes.
- Enviar el archivo a través de la página de aula del ramo, sección “Práctico 5” hasta las 23:59:59 del día Domingo 21/06/2020 hora local continental de Chile (UTC-4).
- Utilice solo herramientas aprendidas en clases y presentes en las diapositivas del curso. **No utilice códigos genéricos obtenidos por búsqueda en la web.**
- Sea riguroso con las instrucciones de desarrollo.
- ¡Éxito!

1 Problemas

1. El Departamento de Estudios Humanísticos (DEH) de la PyTFSM está ad-ortas de su tradicional competición anual de deletreo: **Duelo a Muerte con Palabras**. Durante la competición se sortea aleatoriamente una cantidad de palabras para ser asignadas a los concursantes desde el archivo `palabras.txt`, el cual contiene las palabras seleccionadas por la organización del concurso. Un ejemplo de este archivo se muestra a continuación:

`palabras.txt`

```
obstuso
paralelepipedo
esmerimemelo
onomatopeya
luminiscencia
arcaico
colorido
pormenores
paralelo
quijada
aljibe
elegancia
displicencia
oportunismo
caballerosidad
reciproco
autonomia
autodidacta
agorafobia
incandescente
...
```

Note que el archivo puede contener más palabras.

El DHE le pide lo siguiente:

- Easy Mode/Modo Fácil: Desarrolle la función `get_palabras(archivo)`, la cual recibe como parámetro el nombre de un **archivo** que contiene una lista de palabras preseleccionadas para el concurso. La función debe retornar una lista con todas las palabras preseleccionadas presentes en el **archivo**. Note que cualquier caracter extra obtenido desde la lectura del **archivo** debe ser eliminado. Además, si no puede acceder al archivo debe lanzar la excepción correspondiente.

```
>>> get_palabras("palabras.txt")
['obstuso', 'paralelepipedo', 'esmerimemelo', 'onomatopeya', 'luminiscencia',
'arcaico', 'colorido', 'pormenores', 'paralelo', 'quijada', 'aljibe',
'elegancia', 'displidencia', 'oportunismo', 'caballerosidad', 'reciproco',
'autonomía', 'autodidacta', 'agorafobia', 'incandescente']

>>> get_palabras("palabritas.txt")
Atención el archivo no existe.
Asegurese de colocar bien el nombre o crearlo.
Traceback (most recent call last):
  File "p5.py", line 17, in <module>
    print(get_palabras("palabritas.txt"))
  File "p5.py", line 4, in get_palabras
    arch=open(archivo)
FileNotFoundError: [Errno 2] No such file or directory: 'palabritas.txt'
```

- Medium Mode/Modo Medium: Cree la función `sorteo(pal, num)` la cual recibe como parámetros una lista de palabras preseleccionadas, `pal`, y un número de palabras a seleccionar, `num`. La función debe seleccionar aleatoriamente, de una por una, las palabras a asignar y guardarlas una lista ordenada desde la primera hasta la última seleccionada. Note que las palabras seleccionadas **no se pueden repetir**. La función debe retornar una lista de listas dónde el primer componentes es la lista de palabras seleccionadas, y el segundo es la lista de palabras que quedaron sin escoger.

Además, la cantidad de palabras a escoger es mayor a las disponibles, no se realiza la selección y la función debe levantar una excepción. Lo mismo debe ocurrir si la cantidad de palabras a escoger no es positiva. Recuerde que en la librería `random`, existen variadas funciones (`randint`, `choice`, `sample`, `shuffle`, etc.) para la elección aleatoria.

```
>>> palabras = get_palabras("palabras.txt")
>>> res = sorteo(palabras, 5)
>>> print(res)
[['incandescente', 'aljibe', 'reciproco', 'luminiscencia', 'colorido'],
 ['obstuso', 'paralelepipedo', 'esmerimemelo', 'onomatopeya', 'arcaico',
  'pormenores', 'paralelo', 'quijada', 'elegancia', 'displicencia',
  'oportunismo', 'caballerosidad', 'autonomía', 'autodidacta', 'agorafobia']]

>>> res = sorteo(palabras, -1)
Traceback (most recent call last):
  File "p5.py", line 46, in <module>
    res = sorteo(palabras, -1)
  File "p5.py", line 24, in sorteo
    raise ValueError("El numero de palabras a seleccionar no puede ser cero...")
ValueError: El numero de palabras a seleccionar no puede ser cero o negativo

>>> res = sorteo(palabras, 100)
Traceback (most recent call last):
  File "p5.py", line 48, in <module>
    res = sorteo(palabras, 100)
  File "p5.py", line 26, in sorteo
    raise ValueError("Existen menos palabras que las solicitadas")
ValueError: Existen menos palabras que las solicitadas
```

- **Hard Mode/Modo Difícil:** Desarrolle un programa que solicite realizar N sorteos de M palabras para el concurso a partir de un conjunto de palabras disponibles. Los resultados de cada sorteo deben ser registrados en el archivo `sorteo.txt` y debe seguir el formato que se adjunta en el ejemplo. Además, las palabras escogidas en todos los sorteos no deben repetirse. El programa debe finalizar al alcanzar los N sorteos o al momento que no se pueden generar más sorteos (si es que la cantidad de palabras no es suficiente.). Si ocurriese el caso de no poder realizar un sorteo, un mensaje de aviso debe ser escrito en `sorteo.txt`

Ejemplo 1:

```
Ingrese un número de sorteos: 5
Ingrese número de palabras por sorteo: 4
Sorteo 1 efectuado correctamente
Sorteo 2 efectuado correctamente
Sorteo 3 efectuado correctamente
Sorteo 4 efectuado correctamente
Sorteo 5 efectuado correctamente
```

sorteo.txt

```
Sorteo 1
autonomía - quijada - paralelepipedo - caballerosidad

Sorteo 2
arcaico - autodidacta - incandescente - esmerimemelo

Sorteo 3
onomatopeya - oportunismo - reciproco - paralelo

Sorteo 4
agorafobia - colorido - displicencia - elegancia

Sorteo 5
obstuso - luminiscencia - aljibe - pormenores
```

Ejemplo 2:

```
Ingrese un número de sorteos: 5
Ingrese número de palabras por sorteo: 6
Sorteo 1 efectuado correctamente
Sorteo 2 efectuado correctamente
Sorteo 3 efectuado correctamente
Fin de sorteos, no se puede realizar un sorteo por error de tipo: 'ValueError'
```

sorteo.txt

```
Sorteo 1
displicencia - elegancia - autodidacta - paralelepipedo - arcaico - agorafobia

Sorteo 2
obstuso - colorido - quijada - esmerimemelo - aljibe - onomatopeya

Sorteo 3
autonomía - reciproco - oportunismo - incandescente - luminiscencia - paralelo

Fin de sorteos, no se puede realizar un sorteo por error de tipo: 'ValueError'
```