

Examen de Prácticas de Programación Septiembre 2019 (Grados en Física y Matemáticas)

Se dispone de un fichero con datos de las plantas solares instaladas en diversos estados de EEUU y se desea hacer parte del software que sirva para su análisis. Se dispone de la clase Estado ya realizada, que guarda los datos de las plantas solares de un estado concreto y responde al diagrama de clases que se muestra.

Estado	ListaEstados
-nombre: str -numero_plantas: int -capacidad_instalada: float -generacion_anual: float	-lista: List[Estado] -nombre_fichero: str -fuente: str
+__init__(nombre: str, numero_plantas: int, capacidad_instalada: float, generacion_anual: float) +get_nombre(): str +get_numero_plantas(): int +get_capacidad_instalada(): float +get_generacion_anual(): float	+__init__(nombre_fichero: str) {raises FicheroNoExiste} +lee() +resumen() +eficiencia(estado: str): float {raises EstadoNoEncontrado} +mas_plantas(): str

Los atributos de la clase se indican a continuación. La clase dispone también de un constructor al que se pasan los valores de los atributos y de un método observador para cada atributo:

- nombre: el nombre del estado
- numero_plantas: el número de plantas solares del estado
- capacidad_instalada: la potencia instalada del estado, en MW
- generacion_anual: valor de la energía total generada en el estado en un año, en GWh

Se pide hacer parte de la clase ListaEstados que responde al diagrama de clases que se muestra arriba. La clase almacena en un atributo una lista de objetos de la clase Estado. Dispone también de un atributo para guardar el nombre del fichero del que proceden los datos y otro para la fuente de los datos. También dispone de diversos métodos para analizar las plantas solares:

- *constructor*: crea la lista vacía guardándola en el atributo lista, anota en el atributo nombre_fichero el nombre que se pasa como parámetro y lee el fichero para encontrar la fuente de los datos, que se guardará en el atributo fuente. Si el fichero no existe se pone un mensaje de error y se lanza la excepción FicheroNoExiste ya definida en el mismo módulo.

La fuente de datos está en la segunda línea del fichero, tras los caracteres "Source: " situados al inicio de la línea. Ejemplo de esta segunda línea (la fuente sería "Energy.gov"):
Source: Energy.gov

Si la segunda línea del fichero no comienza por "Source: " se pone el atributo fuente al valor "Fuente desconocida".

- lee(): (Método ya hecho) lee los datos de las plantas solares del fichero cuyo nombre se halla en el atributo nombre_fichero y los mete en la lista. Si este no existe, el método no hace nada.

- `resumen()`: (Método ya hecho) muestra en pantalla un resumen de los datos de la lista.
- `eficiencia()`: calcula y retorna la eficiencia de las plantas solares del estado cuyo nombre se pasa como parámetro. La eficiencia de un estado es

$$eficiencia = \frac{g \cdot 1000}{c \cdot 12 \cdot 365}$$

siendo g su generación anual y c su capacidad instalada. Si el estado indicado no estuviese en la lista se lanza `EstadoNoEncontrado` ya definida en el mismo módulo.

- `mas_plantas()`: retorna el nombre del estado que tiene mayor número de plantas solares o `None` si la lista está vacía.

Finalmente, se pide hacer un programa principal en el mismo módulo, que haga lo siguiente:

- a. Crea un objeto de la clase `ListaEstados` con el nombre de fichero `solar_plants.csv`
- b. Invoca al método `lee()`
- c. Pone en pantalla un resumen de la lista invocando al método `resumen()`
- d. Muestra en pantalla la eficiencia de las plantas solares de California
- e. Muestra en pantalla el nombre del estado que tiene mayor número de plantas solares.

Tratamiento de errores a realizar en el programa principal:

- Si en el paso a) se lanzase `FicheroNoExiste` se abandonan los restantes pasos y se escribe el mensaje "El programa no se puede ejecutar"
- Si en el paso d) se lanzase `EstadoNoEncontrado` se pone un mensaje de error y luego se continúa normalmente con el paso e)

Entregar en moodle:

- código fuente
- informe en pdf con una captura de la ejecución de los resultados del main

Valoración: Se valorarán principalmente los métodos o partes de métodos que demuestren un funcionamiento correcto a través de la captura de los resultados del main

- encabezamiento de la clase y constructor: 2 puntos
- método `eficiencia()`: 2 puntos
- método `mas_plantas()`: 2 puntos
- programa principal: 2 puntos
- informe pdf: 2 puntos