**05 – Expresiones regulares – ejercicios**

1. **Extracción de imágenes**

Diseña una expresión regular que capture y extraiga el nombre de un archivo de imagen (esto es, *.img,* .jpg, \*.png) de una url. Por ejemplo, en las siguientes:

* http://www.domain.com/static/js/imagen1.jpg
* http://www.domain.com/index.php?a=imagen2.img
* http://www.domain.com/index.php?a=/imagen3.png
* http://www.domain.com/?v=http://domain.com/static/imagen4.mpg
* http://www.domain.com/?v=http://domain.com/static/?v=http://www.domain/src/este.js&var=lol

Detalles:

* El nombre del archivo estará precedido por el carácter "/" o "=".
* Luego, caracteres distintos del carácter "/" o "=", seguidos de un punto y una extensión válida. Ésta será la cadena de caracteres que nos interesa capturar

1. **Coloreado de un fragmento de texto, en un archivo HMTL**

He aquí un fragmento de un archivo html:

|  |
| --- |
| <p>  'Este fragmento de html es negro, <font color="00ff00">ahora verde</font> y de nuevo negro...'  </p> |
|
|
|

Viendo cómo queda en la pantalla con mi navegador, resulta lo siguiente:

|  |
| --- |
| Este fragmento de html es negro, ahora verde y de nuevo negro... |

Te pido una expresión regular que identifique, en una cadena de caracteres, un fragmento de texto coloreado y el código del color que tiene:

|  |
| --- |
| >>> cadena = 'Este fragmento de html es negro, <font color="00ff00">ahora verde</font> y de nuevo negro...'  >>> obtener\_texto\_y\_color(cadena)  ('00ff00', 'ahora verde') |

Obviamente, lo más interesante de la función *“obtener\_texto\_y\_color”* es el patrón usado, en el que se define la expresión regular adecuada. Pero no es lo único.