

Lenguajes de marcas y sistemas de gestión de información

UT05. Act03. JSON

Francisco José García Cutillas | 2FPGS_DAM

Índice

Ejercicio 1 3

Ejercicio 2 4

Ejercicio 3 5

Ejercicio 4 6

Ejercicio 5 7

Ejercicio 1

Vas a crear un fichero JSON que ofrezca un listado de una o más editoriales, donde cada editorial presenta la siguiente información: nombre, dirección, email, página web, teléfono, fax y una lista de libros donde para cada libro queremos guardar los siguientes datos: Título, Autor, Fecha e ISBN. Debes crear al menos datos de dos editoriales con dos libros cada una.

```
{
  "editoriales": [
    {
      "nombre": "Paraninfo",
      "direccion": "Calle José Abascal 41 (Madrid)",
      "email": "info@paraninfo.es",
      "paginaWeb": "www.paraninfo.es",
      "telefono": "914463350",
      "fax": "914585666",
      "libros": [
        {
          "titulo": "Abuelo, mándame un email",
          "autor": "Patricia Fuente Rico",
          "fecha": "05/06/2021",
          "isbn": "978842833177"
        },
        {
          "titulo": "Ciberseguridad. Manual práctico",
          "autor": "José Manuel Ortega Candel",
          "fecha": "25/09/2017",
          "isbn": "9788413661162"
        }
      ]
    },
    {
      "nombre": "Mcgraw Hill",
      "direccion": "Calle Basauri 17 ARAVACA (Madrid).",
      "email": "incidencias@mheducation.com",
      "paginaWeb": "www.mheducation.es",
      "telefono": "911803240",
      "fax": "911803055",
      "libros": [
        {
          "titulo": "Programación multimedia y dispositivos móviles",
          "autor": "José Alfredo Murcia Andrés",
          "fecha": "25/05/2022",
          "isbn": "97884448626754"
        },
        {
          "titulo": "Desarrollo de interfaces",
          "autor": "Javier Catalá Jiménez",
          "fecha": "22/05/2022",
          "isbn": "97884448626747"
        }
      ]
    }
  ]
}
```

Ejercicio 2

Crea un fichero JSON que contenga datos de al menos 3 personas, estos deberán tener los siguientes:

- a) Nombre
- b) Apellidos
- c) Fecha de nacimiento
- d) Edad
- e) Dirección: Estará compuesta por “tipo” en el que se indicará si es principal o secundaria, “nombre” que contendrá el nombre de la calle, “numero” para indicar el número y un “ampliación” que podrá tener texto de todo aquello que se considere. Una persona puede tener más de una dirección (deberá haber al menos un ejemplo que tenga más de una dirección)
- f) Población
- g) Provincia
- h) Idiomas que habla (deberá haber al menos un ejemplo de persona que hable más de un idioma).

```

1  {}
2  {
3    "personas": [
4      {
5        "nombre": "Evaristo",
6        "apellidos": "Muñoz Mondéjar",
7        "fechaNacimiento": "08/05/1963",
8        "edad": 60,
9        "direccion": [
10         {
11           "tipo": "secundaria",
12           "nombre": "falsa",
13           "numero": 123,
14           "ampliacion": "planta baja"
15         },
16         {
17           "tipo": "principal",
18           "nombre": "Mayo",
19           "numero": 1,
20           "ampliacion": "escalera derecha"
21         }
22       ],
23       "poblacion": "Elche",
24       "provincia": "Alicante",
25       "idiomas": [
26         "inglés",
27         "francés"
28       ],
29     },
30     {
31       "nombre": "Candelaria",
32       "apellidos": "Rodríguez Giménez",
33       "fechaNacimiento": "12/08/1968",
34       "edad": 55,
35       "direccion": [
36         {
37           "tipo": "principal",
38           "nombre": "barcelona",
39           "numero": 23,
40           "ampliacion": "1ºE"
41         }
42       ],
43       "poblacion": "Tobarra",
44       "provincia": "Albacete",
45       "idiomas": [
46         "italiano"
47       ],
48     },
49     {
50       "nombre": "Miroslav",
51       "apellidos": "Lukaschenko Ivanovich",
52       "fechaNacimiento": "25/12/1983",
53       "edad": 40,
54       "direccion": [
55         {
56           "tipo": "secundaria",
57           "nombre": "quinto pino",
58           "numero": 25,
59           "ampliacion": "sótano"
60         }
61       ],
62       "poblacion": "Los Infiernos",
63       "provincia": "Murcia",
64       "idiomas": [
65         "ruso",
66         "búlgaro"
67       ],
68     }
69   ]
70 }

```

Ejercicio 3

A partir del documento adjunto cartelera.xml obtén usando una web de conversión on-line el equivalente en formato JSON.

Vamos a utilizar el conversor online disponible en la página <https://codebeautify.org/xmltojson>.

- a) Analiza el mismo e indica aquello que te resulte relevante (pequeño resumen de la información que obtiene)

Al copiar el XML en la pantalla izquierda, hace la conversión automáticamente en el lado derecho a JSON.

Podemos observar que hay varios tipos de tabular el JSON, desde 2 espacios hasta 8.

Con respecto a la información del JSON, podemos observar que tiene la información de una cartelera, la cual contiene un array de objetos película. El objeto película tiene los atributos título, título original, nacionalidad, género, clasificación, sinopsis, director, reparto, el cual contiene un objeto de tipo actor. Dentro del objeto actor tenemos un array de actores. Para terminar con el objeto película también tiene web y cartel.

Podemos observar que los atributos de los elementos no los muestra de ninguna manera en el JSON, por lo que hemos perdido información en la conversión.

- b) Número de caracteres que ocupan ambos ficheros

Podemos observar que el XML ocupa 2.31KB, mientras que el JSON ocupa 1.79KB.

Ejercicio 4

Convierte utilizando alguna web de conversión on-line el documento JSON que has obtenido en el ejercicio a XML.

Volvemos a utilizar la misma herramienta que en el ejercicio anterior <https://codebeautify.org/jsontoxml>

- a) Analiza el mismo e indica aquello que te resulte relevante (pequeño resumen de la información que obtiene)

The screenshot shows the 'JSON to XML Converter' tool. The input JSON on the left is as follows:

```

1 {
2   "cartelera": {
3     "película": [
4       {
5         "título": "AQUELLAS JUERGAS
6           UNIVERSITARIAS",
7         "título_original": "Old School",
8         "nacionalidad": "Estados Unidos",
9         "género": "Comedia",
10        "clasificación": "",
11        "sinopsis": "Mitch, Frank y Beanie son
12          tres amigos treintañeros \n
13          cuyas vidas no son
14          exactamente lo que esperaban.
15          Mitch tiene \n
16          una
17          novia ninfómana que se mete en la
18          cama con el primero que \n
19          agarra. Frank se ha casado y
20          su matrimonio nada tiene que ver
21          con las juergas
22          salvajes que organizaban años
23          atrás.",
24        "director": "Todd Philips",
25        "reparto": {
26          "actor": [
27            "Luke Wilson",
28            "Will Farrel",
29            "Vince Vaughn"
30          ]
31        },
32        "web": "http://www.uip.es",
33        "cartel": "caratulas/Aquellas juergas
34          .jpg"
35      }
36    ],
37    {
38      "título": "EL ORO DE MOSCÚ",
39      "nacionalidad": "España",
40      "género": "Comedia",
41      "sin_clasificar": "",
42      "sinopsis": "Por una extraña
43        coincidencia del destino, alguien
44        ..."
45    }
46  }
47 }

```

The resulting XML output on the right is:

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <root>
3   <cartelera>
4     <película>
5       <título>AQUELLAS JUERGAS
6         UNIVERSITARIAS</título>
7       <título_original>Old School
8         </título_original>
9       <nacionalidad>Estados Unidos
10        </nacionalidad>
11       <género>Comedia</género>
12       <clasificación></clasificación>
13       <sinopsis>Mitch, Frank y Beanie
14         son tres amigos treintañeros
15         cuyas vidas no son exactamente lo
16         que esperaban. Mitch tiene
17         una novia ninfómana que se mete en
18         la cama con el primero que
19         agarra. Frank se ha casado y su
20         matrimonio nada tiene que ver
21         con las juergas salvajes que
22         organizaban años atrás.
23       </sinopsis>
24       <director>Todd Philips</director>
25       <reparto>
26         <actor>Luke Wilson</actor>
27         <actor>Will Farrel</actor>
28         <actor>Vince Vaughn</actor>
29       </reparto>
30       <web>http://www.uip.es</web>
31       <cartel>caratulas/Aquellas juergas
32         .jpg</cartel>
33     </película>
34     <película>
35       <título>EL ORO DE MOSCÚ</título>
36       <nacionalidad>España</nacionalidad>
37       <género>Comedia</género>
38       <sin_clasificar></sin_clasificar>
39       <sinopsis>Por una extraña
40         coincidencia del destino,
41         alguien
42       </sinopsis>
43     </película>
44   </cartelera>
45 </root>

```

Podemos observar, que en este caso no nos ofrece la opción de tabularlo. En este caso, en el XML, nos genera el elemento raíz <root>, dentro del cual está contenido el elemento cartelera, que a su vez contiene varios elementos película. Película está compuesto por título, título original (puede no aparecer), nacionalidad, género, clasificación o sin clasificar, sinopsis, director, reparto (formado por elementos actor), web y cartel.

Como partimos del JSON obtenido en el ejercicio anterior, obviamente al volver al XML los atributos de los elementos originales de cartelera.xml se han perdido.

- b) Número de caracteres que ocupan ambos ficheros

En este caso el JSON ocupa 1.87KB y el XML 1.83KB.

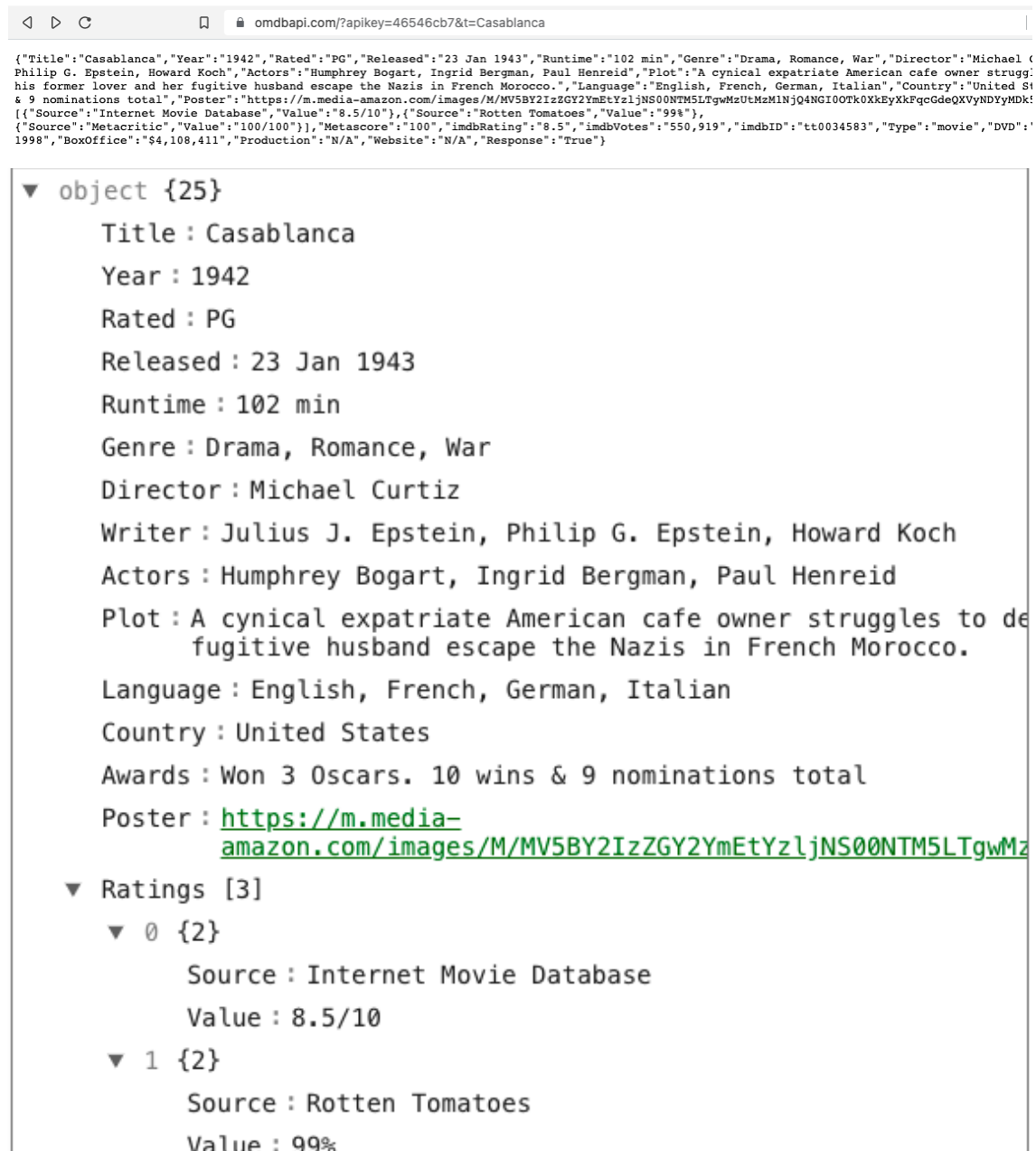
Ejercicio 5

Busca alguna API que pueda devolverte información sobre alguna cuestión que te resulte de interés y que devuelva el resultado de tus peticiones en formato JSON.

Ejemplo de API en este sentido es **OMDBAPI** (<https://www.omdbapi.com/>) a la que se le pueden hacer peticiones con diferentes parámetros que allí nos indica.

Por ejemplo, si quisiera obtener información en formato JSON de la película Casablanca podría hacer la siguiente petición: <https://www.omdbapi.com/?apikey=46546cb7&t=Casablanca>

NOTA: He tenido que obtener un apikey que en el caso de esta web nos permite de manera gratuita hacer 1000 consultas al día.

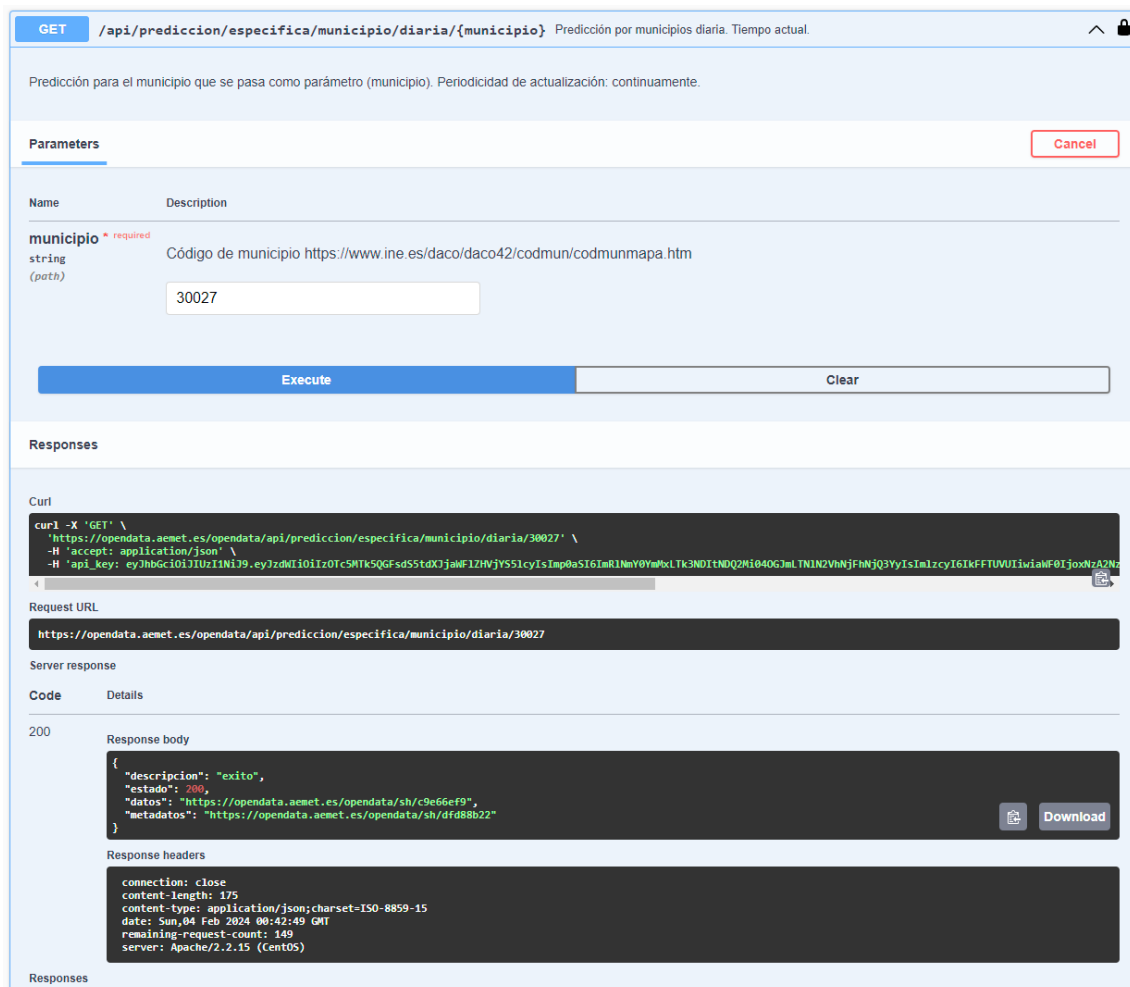
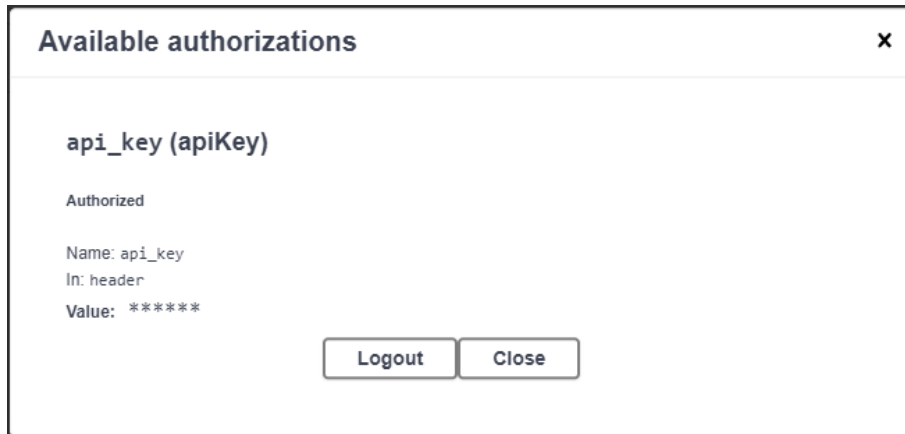


En mi caso vamos a utilizar la API que nos ofrece la AEMET (Agencia Estatal de Meteorología) en <https://opendata.aemet.es/dist/index.html?#/>

Aquí nos aparecen varias formas de hacer las peticiones a su servidor. Nosotros vamos a tomar por ejemplo, de las predicciones normalizadas, este tipo de petición:

```
/api/prediccion/especifica/municipio/diaria/{municipio}
```

Para utilizar este servicio, debemos solicitar una `apiKey`, la cual te mandan a tu correo y tiene una validez de 5 días.



Si ejecutamos con el código 30027 que es el que corresponde con Molina de Segura, nos muestra la URL de petición. En el cuerpo de la respuesta, un fichero JSON, podemos observar que en el elemento “datos” es donde se encuentra en realizad el JSON con toda la información.

```
[
  {
    "origen" : {
      "productor" : "Agencia Estatal de Meteorología - AEMET. Gobierno de España",
      "web" : "https://www.aemet.es",
      "enlace" : "https://www.aemet.es/es/eltiempo/prediccion/municipios/molina-de-segura-id30027",
      "language" : "es",
      "copyright" : "© AEMET. Autorizado el uso de la información y su reproducción citando a AEMET como autora de la misma.",
      "notalegal" : "https://www.aemet.es/es/nota_legal"
    },
    "elaborado" : "2024-02-03T22:41:08",
    "nombre" : "Molina de Segura",
    "provincia" : "Murcia",
    "prediccion" : {
      "dia" : [ {
        "probPrecipitacion" : [ {
          "value" : 0,
          "periodo" : "00-24"
        }, {
          "value" : 0,
          "periodo" : "00-12"
        }, {
          "value" : 0,
          "periodo" : "12-24"
        }, {
          "value" : 0,
          "periodo" : "00-06"
        }, {
          "value" : 0,
          "periodo" : "06-12"
        }, {
          "value" : 0,
          "periodo" : "12-18"
        }, {
          "value" : 0,
          "periodo" : "18-24"
        } ],
        "cotaNieveProv" : [ {
          "value" : "",
          "periodo" : "00-24"
        }, {
          "value" : "",
          "periodo" : "00-12"
        }, {
          "value" : "",
          "periodo" : "12-24"
        }, {
          "value" : "",
          "periodo" : "00-06"
        }, {
          "value" : "",
          "periodo" : "06-12"
        }, {
          "value" : "",
          "periodo" : "12-18"
        }, {
          "value" : "",
          "periodo" : "18-24"
        } ],
        "estadoCielo" : [ {
          "value" : "",
          "periodo" : "00-24",
          "descripcion" : ""
        }, {
          "value" : "",
          "periodo" : "00-12",
          "descripcion" : ""
        }, {
          "value" : "",
          "periodo" : "12-24",
          "descripcion" : ""
        }, {
          "value" : "",
          "periodo" : "00-06",
          "descripcion" : ""
        }, {
          "value" : "",
          "periodo" : "06-12",
          "descripcion" : ""
        }, {
          "value" : "",
          "periodo" : "12-18",
          "descripcion" : ""
        }, {
          "value" : "",
          "periodo" : "18-24",
          "descripcion" : ""
        } ]
      } ]
    }
  }
]
```

En la página aparece también la información de las posibles respuestas que puedes obtener del servidor al realizar esta petición.

Responses		
Code	Description	Links
200	respuesta con éxito	No links
<div>Media type</div> <div>application/json</div> <div>Controls Accept header</div> <div>Example Value Schema</div> <div><pre>{ "descripcion": "Éxito", "estado": 0, "datos": "string", "metadatos": "string" }</pre></div>		
401	petición no autorizada	No links
<div>Media type</div> <div>application/json</div> <div>Example Value Schema</div> <div><pre>{ "descripcion": "Unauthorized", "estado": 0 }</pre></div>		
404	petición sin datos	No links
<div>Media type</div> <div>application/json</div> <div>Example Value Schema</div> <div><pre>{ "descripcion": "Not Found", "estado": 0 }</pre></div>		
429	petición que sobrepasa los límites del servicio	No links
<div>Media type</div> <div>application/json</div> <div>Example Value Schema</div> <div><pre>{ "descripcion": "Too Many Requests", "estado": 0 }</pre></div>		