

CONTEXTO

El **BORME** (Boletín Oficial del Registro Mercantil) es el órgano oficial de publicidad del Registro Mercantil de España, en el que **se publican los actos jurídicos** que, por disposición legal, deben hacerse públicos.

La información que se publica se estructura en Secciones:

- Sección Primera: Empresarios
 - o Actos inscritos (BORME-A)
 - o Otros actos (BORME-B)
- Sección Segunda: Anuncios y avisos legales (BORME-C)

Vamos a restringirnos a los BORME-A (sección primera, actos inscritos).

Éstos se publican, salvo excepciones, **todos los días laborales del año** en el BOE (https://www.boe.es/diario_borme/), de forma que **cada registro mercantil provincial recoge, en un PDF, toda la información que se debe hacer pública** en esta sección. Desde su inicio (enero de 2009), se han publicado más de 100.000 PDFs de tipo A.

Grosso modo, **la estructura de cada PDF es esta:**

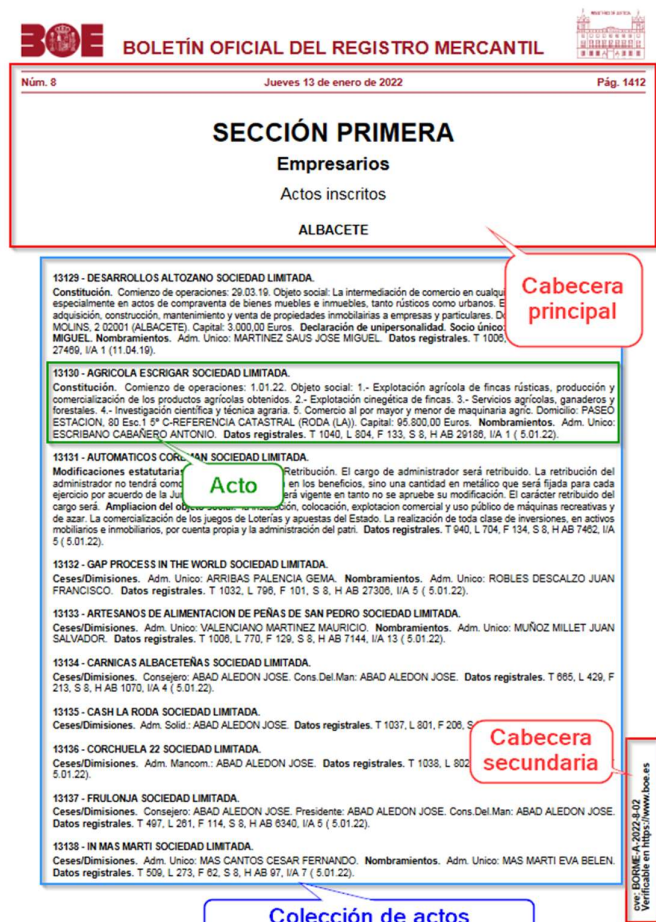
- Cabecera del PDF (en rojo)
- Colección de actos (en azul)

Cada acto (en verde), recoge la información que se publica para una determinada entidad jurídica.

Se han publicado, desde su inicio, más de 11.000.000 actos.

La información que se publica en cada acto se estructura en subactos, según la naturaleza de lo que se inscribe en el registro mercantil. Algunos tipos de subactos son:

- Constitución
- Nombramientos (de cargos)
- Ceses/Dimisiones (de cargos)
- Fusión (de empresas)
- Escisión (de empresas)



Aunque no hay una especificación clara, se sabe que cada acto sigue este formato:

- Identificador del acto, que NO es único dentro del BORME (verde).
- Razón Social de la entidad jurídica a la que se refiere (amarillo).
- Subactos (rosa).
- Hoja registral (naranja)
- Fecha de inscripción (gris)

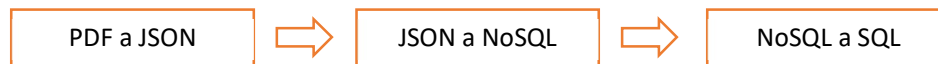
13130 - AGRICOLA ESCRIGAR SOCIEDAD LIMITADA.

Constitución. Comienzo de operaciones: 1.01.22. Objeto social: 1.- Explotación agrícola de fincas rústicas, producción y comercialización de los productos agrícolas obtenidos. 2.- Explotación cinegética de fincas. 3.- Servicios agrícolas, ganaderos y forestales. 4.- Investigación científica y técnica agraria. 5. Comercio al por mayor y menor de maquinaria agric. Domicilio: PASEO ESTACION, 80 Esc.1 5º C-REFERENCIA CATASTRAL (RODA (LA)). Capital: 95.800,00 Euros. **Nombramientos.** Adm. Unico: ESCRIBANO CABAÑERO ANTONIO. **Datos registrales.** T 1040, L 804, F 133, S 8, H AB 29186, I/A 1 (5.01.22).

Se han publicado, desde su inicio, más de 15.000.000 subactos

Para almacenar toda la información, vamos a suponer que se ha definido un modelo mixto (SQL + NoSQL), de forma que:

- La información sin tratar y parcialmente tratada se guarda en una base de datos NoSQL.
- La información tratada y estructurada se almacena sobre tablas relacionales de SQL.



Para los tres primeros ejercicios de la prueba, se partirá de este contexto y **se pedirá resolver elementos muy concretos** de un **hipotético ciclo de datos** basado en el procesamiento de **datos del BORME**.

En la medida de lo posible, se pide:

- **Justificar** aquello que se haga que no resulte trivial.
- **Ser concreto** resolviendo lo pedido.

Pregunta 1 [Python]

Subactos de constitución. Partiendo de algunos ejemplos de subactos de Constitución (json disponibles en la carpeta correspondiente a esta pregunta).

Se pide:

- **Crear una función** que recorra el json y extraiga la información que se publica en dicho tipo de subactos (ver figura abajo): Comienzo de operaciones (Fecha de constitución), Objeto social, Domicilio, Capital.
- **Explicar la lógica de la función.**

366623 - GRUPO MEXCALISTA SL.	
Constitución.	Comienzo de operaciones: 7.07.21. Objeto social: La explotación de todo tipo de cafeterías, bares , restaurantes, hoteles, terrazas y otros establecimientos de hostelería. Domicilio: C/ LEON 31 (MADRID). Capital: 3.010,00 Euros. Nombramientos.
Adm. Unico: PEREZ RUMEBE LUCIANO. Datos registrales. T 42286 , F 50, S 8, H M 748613, I/A 1 (27.07.21).	

Función:

- *Entrada: json del acto*
- *Salida: json con el formato siguiente:*

```
{
  "_id": "XXXX",
  "row": [
    {
      "key": "Constitución",
      "value": [
        {
          "key": "Comienzo de operaciones",
          "value": "7.07.21"
        },
        {
          "key": "Objeto social",
          "value": "La explotación de todo tipo de cafeterías, bares , restaurantes, hoteles, terrazas y otros establecimientos de hostelería"
        },
        {
          "key": "Domicilio",
          "value": "C/ LEON 31 (MADRID)"
        },
        {
          "key": "Capital",
          "value": "3.010,00 Euros"
        }
      ]
    }
  ],
  "header_RAW": "366623",
  "company_RAW": "GRUPO MEXCALISTA SL"
}
```

- Una vez extraídos los diferentes campos, con los datos de las fechas de constitución, se pide **crear una función** que valide y normalice dichas fechas para que todas tengan el formato YYYY-MM-DD.

Función:

- *Entrada: fecha de constitución en cualquier formato*
- *Salida: fecha de constitución en formato YYYY-MM-DD*

Pregunta 2 [SQL]

Supongamos las siguientes tablas en SQL que guardan información estructurada sobre el BORME:

- **BORME_Cabecera.** Guarda la información de la cabecera del PDF:
 - URL de publicación (BO_url)
 - Fecha de publicación (BO_date)
 - Provincia, usando la codificación del INE (BO_province)
 - Código de verificación (BO_cve)

Name	Value
BO_id	1
BO_cve	BORME-A-2022-8-02
BO_date	2022-01-13
BO_url	boe.es/borme/dias/2022/01/13/pdfs/BORME-A-2022-8-02.pdf
BO_province	02
BO_section	A

- **BORME_Actos.** Guarda la información de cada acto publicado:
 - Fecha del acto (BA_date)
 - Identificador del acto en el PDF (BA_header_RAW)
 - Nombre de la empresa (BA_company_RAW)
 - Hoja registral (BA_RD_sheet)
 - CIF de la empresa (BA_CIF)
 - ID que relaciona el acto con la cabecera (BA_BO_id)

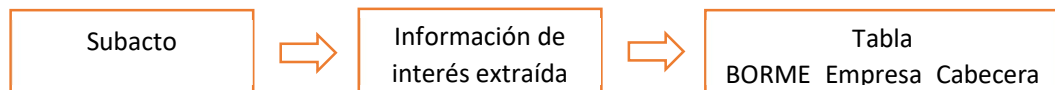
Name	Value
BA_id	1
BA_BO_id	1
BA_date	2022-01-05
BA_header_RAW	13130
BA_company_RAW	AGRICOLA ESCRIGAR SOCIEDAD LIMITADA
BA_RD_sheet	T 1040, L 804, F 133, S 8, H AB 29186
BA_CIF	B67918433

- **BORME_Empresa_Cabecera.** Aquí se guarda la información de referencia de una empresa, extraída de diferentes subactos.

En concreto, tiene información sobre:

- Fecha de constitución (BS_constitution_date)
- Publicación de una web (BS_web)
- Domicilio Social (BS_registered_office)
- Objeto social (BS_social_object)
- Además, guarda información del identificador del acto en el que se publica para poder relacionarlo con la tabla **BORME_Actos** (BS_BA_id)

La particularidad de esta tabla es que tiene un carácter transaccional sobre los subactos. Esto, en la práctica, significa que, para cada subacto publicado que tiene “algo” que ver con lo destacado (fecha de constitución, web, domicilio u objeto sociales), se genera una fila nueva. Y, sobre ésta, se guarda la información disponible en el subacto en cuestión.



Por ejemplo, en este acto de constitución, no aparece la web de la empresa. Pero sí información sobre la fecha de constitución, el domicilio social y el objeto.

13130 - AGRICOLA ESCRIGAR SOCIEDAD LIMITADA.

Constitución. Comienzo de operaciones: 1.01.22. Objeto social: 1.- Explotación agrícola de fincas rústicas, producción y comercialización de los productos agrícolas obtenidos. 2.- Explotación cinegética de fincas. 3.- Servicios agrícolas, ganaderos y forestales. 4.- Investigación científica y técnica agraria. 5. Comercio al por mayor y menor de maquinaria agríc. Domicilio: PASEO ESTACION, 80 Esc.1 5º C-REFERENCIA CATASTRAL (RODA (LA)). Capital: 95.800,00 Euros. **Nombramientos.** Adm. Único: ESCRIBANO CABAÑERO ANTONIO. **Datos registrales.** T 1040, L 804, F 133, S 8, H AB 29186, I/A 1 (5.01.22).

Y, por tanto, **el campo de la web en la tabla estará a NULL.** Y el resto, con la información trasladada.

Name	Value
BS_id	2242114
BS_BA_id	8018470
BS_company_RAW	AGRICOLA ESCRIGAR SOCIEDAD LIMITADA
BS_constitution_date	2022-01-01
BS_registered_office	PASEO ESTACION, 80 Esc.1 5º C-REFERENCIA CATASTRAL (RODA (LA))
BS_social_object	1.- Explotación agrícola de fincas rústicas, producción y ...
BS_date	20220105
BS_web	[NULL]

Al final, en esta tabla y para una empresa concreta, tendremos los actos que han modificado alguna dimensión de estudio.

Varias observaciones:

- La forma de unir empresas es por su CIF (el CIF está indicado en el BA_CIF de la tabla BORME_Actos)
- Como la tabla, en sí, tiene subactos, está relacionada con la tabla de BORME_Actos a través del campo **BS_BA_id**
- La fecha de inscripción del acto está disponible también en esta tabla, pero en formato entero, en lugar de como fecha (campo **BS_date**)

A modo de ejemplo, para la empresa con CIF A04753620, tenemos entre otras cosas diferentes domicilios sociales, objetos sociales, etc. El contenido de la tabla **BORME_Empresa_Cabecera** recogerá esta información:

BS_BA_id	BA_CB_CIF	BS_constitution_date	BS_registered_office	BS_social_object	BS_web	BS_date
9754046	A04753620	2012-10-29	C/ MIGUEL RUEDA 31 3 A - AGUADULCE (ROQUETAS DE MAR)	La compraventa, instalacion, mant...	NULL	20130201
10039351	A04753620	NULL	NULL	NULL	NULL	20130425
10039351	A04753620	NULL	NULL	NULL	http://www.ledvideoscreens.es	20130425
1252397	A04753620	NULL	C/ MIGUEL RUEDA 6 3 A - AGUADULCE (ROQUETAS DE MAR).	NULL	NULL	20170922
1252397	A04753620	NULL	NULL	El procesamiento de datos médicos, la...	NULL	20170922
6170007	A04753620	NULL	C/ NOGAL 2 - AGUADULCE (ROQUETAS DE MAR).	NULL	NULL	20190510

Se supone la siguiente volumetría:

- **BORME_Cabecera**: 100K rows
- **BORME_Actos**: 13M rows
- **BORME_Empresa_Cabecera**: 2M rows

Se quiere construir una última visión de la tabla BORME_Empresa_Cabecera, de forma que, **cada fila tenga la información más reciente de un determinado CIF.**

Para el ejemplo anterior, lo que se deberá obtener será esta información:

CIF	A04753620
BSU_constitution_date	29/10/2012
BSU_constitution_date_BS_BA_id	9754046
BSU_registered_office	C/ NOGAL 2 - AGUADULCE (ROQUETAS DE MAR).
BSU_registered_office_BS_BA_id	6170007
BSU_web	http://www.ledvideoscreens.es
BSU_web_BS_BA_id	10039351
BSU_social_object	El procesado de datos médicos, la compraventa, alquiler, mantenimiento, diseño, desarrollo y fabricación de equipos médicos electrónicos por cuenta propia o por un tercero.
BSU_social_object_BS_BA_id	1252397

Se pide:

- **Explicar qué paso/s** se implementarían (i.e., queries) para conseguir el resultado pedido.
- Indicar los **campos necesarios a indexar** en las tablas para poder ejecutar el proceso.
- Discutir si **el resultado** del proceso **debe materializarse** en una tabla **o si resulta aceptable su consulta al vuelo** (vista), filtrando por un CIF concreto.

Pregunta 3 [SQL]

Supongamos que disponemos de una tabla en SQL que guarda la siguiente información estructurada sobre las ampliaciones/reducciones de capital que una empresa realiza:

- CIF: CIF de la empresa
- BCU_fecha_valor: fecha en la que se realiza la variación de capital (constitución, ampliación de capital, reducción de capital)
- BCU_accion: qué variación de capital se realiza: constitución, ampliación de capital, reducción de capital
- BCU_capital_suscrito: importe en el que se modifica el capital
- BCU_capital_desembolsado: importe desembolsado en el que se modifica el capital
- BCU_resultante_suscrito: capital resultante que quedaría tras la variación de capital indicada en el campo BCU_capital_suscrito
- BCU_resultante_desembolsado: capital resultante desembolsado que quedaría tras la variación de capital indicada en el campo BCU_capital_suscrito_desembolsado

Por ejemplo, para la empresa con CIF B01000652 se registrarían 4 movimientos en su capital.

CIF	BCU_fecha_valor	BCU_accion	BCU_capital_suscrito	BCU_capital_desembolsado	BCU_resultante_suscrito	BCU_resultante_desembolsado
B01000652	2020-06-24	Reducción de capital	-650000	NULL	4350000	NULL
B01000652	2012-02-07	Ampliación de capital	1000000	1000000	5000000	5000000
B01000652	2011-01-04	Ampliación de capital	79440	79440	4000000	4000000
B01000652	2011-01-04	Ampliación de capital	1564303,44	1564303,44	3920560	3920560

A partir de la información de la tabla (ver archivo de la carpeta correspondiente a la pregunta),

Se pide:

- **Explicar qué pasos** se harían (query) para calcular la variación patrimonial, medida según la siguiente fórmula: $((V_2 - V_1) / V_1) \times 100$, donde V_1 representa al valor pasado o inicial y V_2 representa al valor presente o final. Usad la columna *BCU_resultante_suscrito* para los cálculos.
- **Explicar qué pasos** se implementarían para validar que la información publicada sobre las ampliaciones/reducciones es correcta. Es decir, para el ejemplo anterior una comprobación a realizar sería:
 - o A partir del valor inicial (aquel que la fecha indica en BCU_fecha_valor es la más antigua), comprobar que
$$BCU_resultante_suscrito_i = BCU_resultante_suscrito_{i-1} + BCU_capital_suscrito_i$$
$$BCU_resultante_desembolsado_i = BCU_resultante_suscrito_{i-1} + BCU_capital_desembolsado_i$$

Además, se pueden dar otros errores como los detallados a continuación:

- o Error de parseo: ha habido algún problema y el cambio de capital no ha sido registrado en la tabla (falta un registro en la tabla transaccional).
- o Error de ordenado: la información está completa, pero hacemos la comprobación por orden de actos, y los actos no están publicados en el orden correcto y la comprobación falla.

Qué comprobaciones harías para detectar estas situaciones.

Pregunta 4 [Python/API]

Se pide:

- **Crear una API en Flask/FastAPI** que compare dos documentos y devuelva como respuesta el resultado de la comparación. Para ello,
 - Se creará un endpoint que permita que el usuario envíe un documento json con los datos necesarios (ver documentos de la carpeta Pregunta4 - ejemplo A00000001.json).

Este endpoint se encargará de:

 - Dado el CIF que se envía en el json, buscar en el directorio ./data de la aplicación el json correspondiente a ese CIF (nombre del documento=CIF_YYYYMMDD.json).
 - Parsear los json para detectar si hay algún campo cuyo valor es distinto en ambos documentos.
 - Crear un json de salida que muestre el resultado de la comparación (ver ejemplo de documentos respuesta: response_A00000001.json).
 - Almacenar el json de salida en MongoDB*.

**Nota: añadir cómo se haría la interacción con base de datos pero no hace falta crear y conectarse a MongoDB.*
 - Además, la API deberá:
 - Validar que el formato de los campos del documento es el correcto.
 - Registrar el tiempo que tarda cada función en ejecutarse.
 - Gestionar los errores/excepciones que se puedan producir en cualquier punto de la API.
 - Añadir logs para tener traza de lo que va ocurriendo en las llamadas a la API.

Entrada:

Como entrada se espera un json que incluye los siguientes campos:

- **cif:** string con cif de la empresa
- **data:** diccionario con distintos datos:
 - **importe_propuesta:** float
 - **riesgo_vivo:** float
 - **scoring:** string (sólo puede tomar los siguientes valores: A, B, CC, C, D)
 - **importe_impagado:** float
 - **importe_pendiente:** float
 - **result_aeat:** bool
 - **date_created:** date (formato: YYYY-MM-DD)
 - **date_updated:** date (formato: YYYY-MM-DD)

Salida

La salida obtenida de la llamada vendrá en formato JSON y contendrá el resultado del procesamiento:

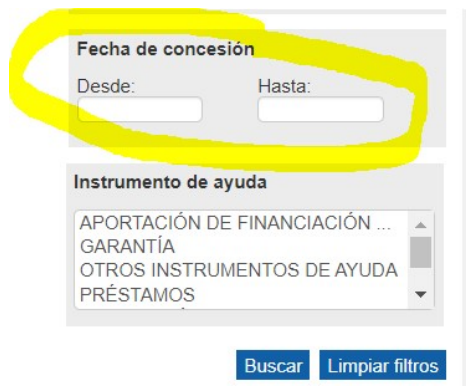
- **manual_validation:** bool indicando si se debe de realizar una validación manual; esto ocurre cuando algún campo de los documentos es diferente.

- **data:** diccionario que tiene la información extraída y el resultado de las comprobaciones hechas.
 - **cif_original:** string con cif del documento original.
 - **cif_uploaded:** string con cif del documento enviado.
 - **cif_chek:** bool que indica la coincidencia de “cif” de ambos documentos.
 - **importe_propuesta_original:** float con el importe de la propuesta del documento original.
 - **importe_propuesta_uploaded:** float con el importe de la propuesta del documento enviado.
 - **importe_propuesta_check:** bool que indica la coincidencia de “importe_propuesta” de ambos documentos
 - **riesgo_vivo_original:** float con el riesgo vivo del documento original.
 - **riesgo_vivo_uploaded:** float con el riesgo vivo del documento enviado.
 - **riesgo_vivo_check:** bool que indica la coincidencia de “riesgo_vivo” de ambos documentos
 - **scoring_original:** string con el scoring del documento original.
 - **scoring_uploaded:** string con el scoring del documento enviado.
 - **scoring_check:** bool que indica la coincidencia de “scoring” de ambos documentos
 - **importe_impagado_original:** float con el importe impagado del documento original.
 - **importe_impagado_uploaded:** float con el importe impagado del documento enviado.
 - **importe_impagado_check:** bool que indica la coincidencia de “importe_impagado” de ambos documentos
 - **importe_pendiente_original:** float con el importe pendiente del documento original.
 - **importe_pendiente_uploaded:** float con el importe pendiente del documento enviado.
 - **importe_pendiente_check:** bool que indica la coincidencia de “importe_pendiente” de ambos documentos
 - **result_aeat_original:** bool que indica el resultado de AEAT del documento original.
 - **result_aeat_uploaded:** bool que indica el resultado de AEAT del documento enviado.
 - **result_aeat_check:** bool que indica la coincidencia de “result_aeat” de ambos documentos

Pregunta 5 [Web Scraping]

Se pide:

- **Explicar** qué pasos darías para saber cómo extraer la información que se publica en la siguiente web: <https://www.infosubvenciones.es/bdnstrans/GE/es/concesiones>
- **Qué harías** para extraer la información publicada en una determinada fecha aplicando los filtros de fecha?



The image shows a web interface for filtering information. It has two main sections: 'Fecha de concesión' and 'Instrumento de ayuda'. The 'Fecha de concesión' section contains two input fields labeled 'Desde:' and 'Hasta:'. This entire section is highlighted with a yellow oval. Below it, the 'Instrumento de ayuda' section features a dropdown menu with the following options: 'APORTACIÓN DE FINANCIACIÓN ...', 'GARANTÍA', 'OTROS INSTRUMENTOS DE AYUDA', and 'PRÉSTAMOS'. At the bottom of the interface, there are two buttons: 'Buscar' and 'Limpiar filtros'.

- En el caso de que una web implemente mecanismos para evitar que se automatice la extracción de información, ¿**qué harías** para evitar dichos mecanismos?