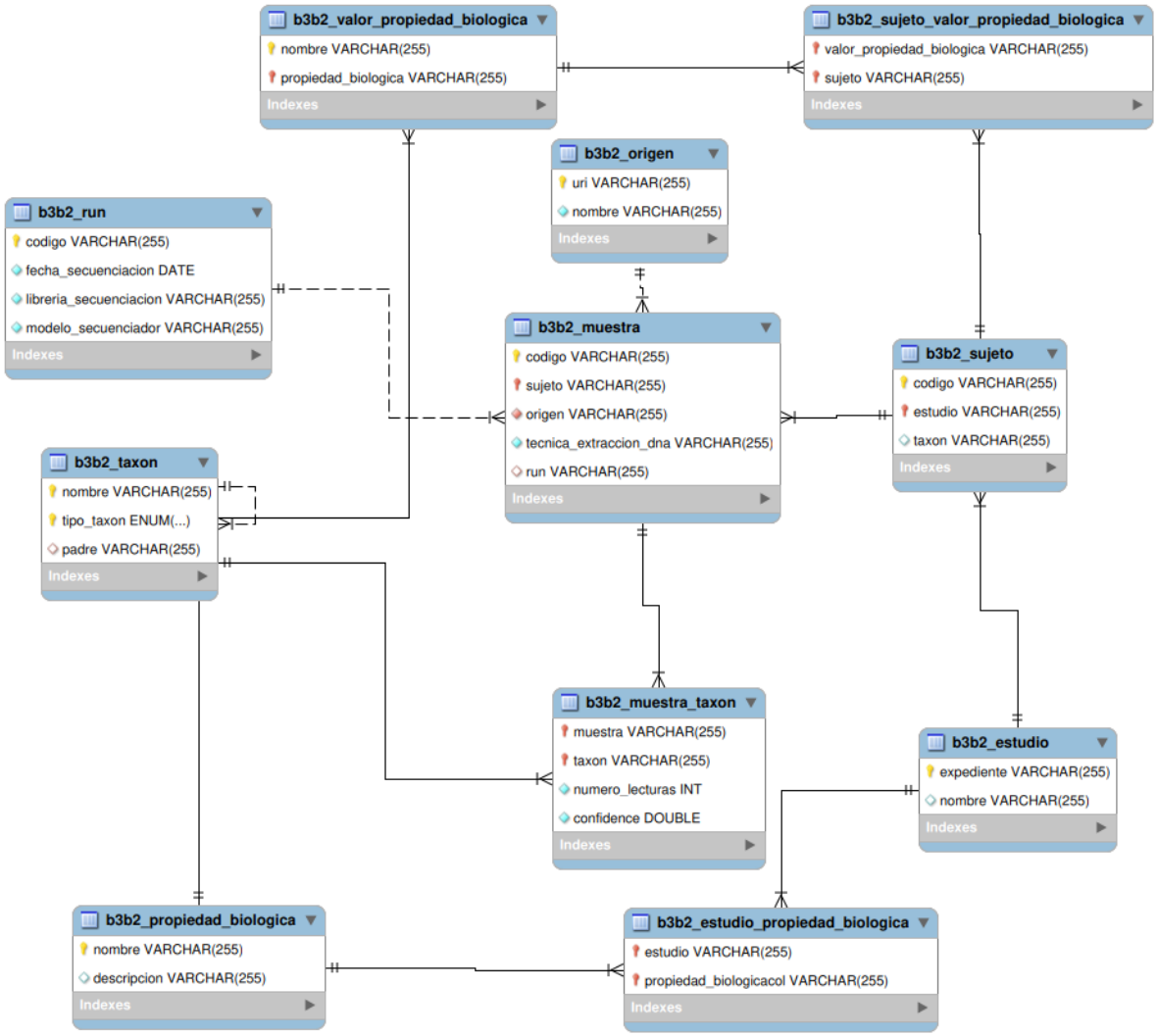
Práctica 2. Gestión y explotación de datos

En esta práctica se evaluará de forma conjunta el uso de lenguajes de Programación (bloque 3) con el uso de lenguajes de modificación y consulta de bases de datos (bloque 2).

Esta práctica constará de 3 ejercicios. Uno de carga de datos y dos de explotación de esos datos. El estudiante se basará en un modelo del entregable uno del bloque 2. En el siguiente diagrama se representa el esquema lógico de dicha base de datos:



En el fichero adjunto a este enunciado podéis encontrar la base de datos que debéis usar en esta práctica con algunas tablas ya rellenas. Las tablas que no están rellenas se deberán rellenar en los siguientes ejercicios.

Ejercicio 1 (10 puntos). Carga de los resultados de experimentos de DNA-SEQ.

En este ejercicio, el estudiante deberá completar las tablas `b3b2_origen`, `b3b2_muestra`, `b3b2_taxon` y `b3b2_muestra_taxon` con los resultados de los ficheros que se usaron en la

primera tarea. Hay algunos aspectos que no hace falta cargar como son las columnas `Species_exact` ni `sequence` ni `confidence` (por defecto le daremos el valor 1). No es obligatorio pero se valorará positivamente que la tabla `b3b2_taxon` se complete el campo padre usando los resultados como referencia.

Para completar la tabla b3b2_origen se va usar un servicio REST del European Bioinformatics Institute (EBI). Os pongo un ejemplo de llamada para obtener todos los posibles orígenes:

```
https://www.ebi.ac.uk/ols4/api/ontologies/Gold/terms/
https%253A%252F%252Fw3id.org%252Fgold.path%252FAtomicElement/children?lang=en
```

Por simplicidad, asumiremos que las muestras CT tiene como origen `Saliva` y que las muestras TR tienen como origen `Fecal`. Pero hay que cargar todos los posible orígenes usando el servicio que tenemos arriba. En la respuesta del servicio guardaremos como `uri` el campo `iri` y como `nombre` el campo `label`. La técnica de extracción de DNA de todas las muestras es la misma y la denominaremos "Extracción estándar DNA".

Se valorará con **8 puntos** la parte de programación en para realizar la carga y con **2 puntos** la parte de inserts, updates, manejo de transacciones, y cierre de conexiones. (8 puntos serán del bloque 3 y 2 puntos del bloque 3).

Ejercicio 2 (8 puntos bloque 2). Consultas para explotar la información

El estudiante usará el lenguaje de consulta SQL para responder a las siguientes preguntas sobre la base de datos completa resultado del Ejercicio 1.

1. Tabla resumen de sujetos en la que tengamos el código, y las propiedades biológicas sexo, edad, patologia y tratamiento. Debe aparecer una única línea por sujeto.
2. Quiero saber qué sujetos no tienen ninguna patología asociada y por lo tanto son controles.
3. Número de muestras por RUN agregando por librería y modelo de secuenciador. Es decir, si dos o más RUN han usado la misma técnica y secuenciador sólo debe aparecer una línea con todas las muestras.
4. Número de taxones por tipo identificados en nuestra base de datos.
5. Diferencia de abundancia diferencial (log de fold change) entre las muestras de los sujetos hombres y mujeres. Por ejemplo de 0.01 en valor absoluto.
6. Tabla resumen con los fármacos (propiedad biológica tratamiento) que se toman por cada patología. Debe salir una línea por fármaco y agregadas las patologías en las que se toman.
7. Listado de sujetos por sexo y edad de aquellos que sean mayores de 40 años.
8. Cuál es el RUN con más muestras y taxones identificados.

Ejercicio 3 (2 puntos bloque 3). Programando la explotación a base de datos

En este ejercicio, el estudiante hará un programa que devuelva mediante una interfaz de texto el

resultado de las 8 consultas anteriores. Se valorará que la consulta 5 sea parametrizable. Es decir, que pueda seleccionar el valor de fold change.

Observaciones

La práctica deberá funcionar en el servidor del máster sin la necesidad de instalar nada adicional.

Se penalizará que el código no esté comentado. No hace falta comentar línea a línea, pero si lo que significa cada variable que usamos y el objetivo de las estructuras de control. También se deberá comentar el código de bloques de código que saquemos de otros programas o de Internet.

Se valorará el control de errores y que el programa no falle por datos que no están correctamente introducidos por el usuario.

Se valorará que no haya código repetitivo.

Se valorará que la salida del programa por consola esté correctamente tabulada y sea legible.

Se deberán entregar dos archivos .ypynb (ejercicio 1 y 3) y sql (con la consultas del ejercicio 2) para hacer la práctica. **NO SON NECESARIOS LOS ARCHIVOS DE EXPERIMENTOS.**