

## Actividad 2: Puesta en marcha de BBDD Cassandra y gestión de datos básica en Python

### Parte 1: Creación de tablas (20%)

Cree una base de datos Cassandra que implemente las tablas diseñadas en la Actividad 1. El nombre del keyspace debe ser su nombre y su primer apellido en minúsculas. Por ejemplo, si usted se llama Juan Méndez, el nombre del keyspace será "juanmendez". Las columnas de las tablas serán todas de tipo text excepto las que se encuentren asociadas a los siguientes atributos del modelo conceptual:

- Ejemplar
  - nro: int
- Es\_prestado
  - fecha: date
- Libro
  - año: int
  - temas: set<text>
- Autor
  - cod: int
  - premios: set<text>
- Editorial
  - cod: Int
- Pais
  - cod: Int

\*Recuerde que cuando un atributo conjunto tenga que ser clave primaria de una tabla, este deberá implementarse como una columna de tipo individual.

Cada tabla deberá ser nombrada como Tabla#, siendo # el número de la consulta de la Actividad 1 que satisface. En el caso de nombrar a las tablas de otra forma, se descontará la mitad de lo obtenido en este apartado.

### Parte 2: Programa Python de gestión de datos (65%)

Se pide realizar un programa que tenga las siguientes funciones o código implementado:

#### Clases por entidad o relación (10%):

Por cada entidad o relación n:m se deberá tener una clase de Python tal y como se explicó en la Práctica 2. Se recomienda usar estas clases para guardar la información en el resto de la práctica.

#### Creación de métodos destinados a consultar la información de tablas soporte (5%)

Se deberán crear tablas soporte para consultar información de las siguientes entidades a través de su clave primaria:

- La entidad Libro
- La entidad Usuario

En el fichero cql a entregar (Parte 1) deberá constar la creación de estas tablas. El nombre de estas tablas será SoporteLibro y SoporteUsuario.

### Creación de métodos de inserción de datos (20%)

Debe crear funciones para insertar instancias de entidades o relaciones en las tablas que se especifiquen.

Se piden funciones para insertar instancias de las siguientes entidades. Se indican las consultas de las que se crearon las tablas donde se debe insertar la información:

- Libro en tabla de consulta 1 y soporte.
- Usuario en tabla de consulta 5 y soporte.
- Autor en tabla de consulta 7.

Se piden funciones para insertar las siguientes relaciones entre instancias en todas las tablas necesarias:

- Relación Es\_prestado
- Relación Corresponde-Es\_prestado

Para determinar dónde insertar la información de una relación se sigue el siguiente criterio:

**La tabla deberá incluir exclusivamente columnas asociadas a atributos de las entidades relacionadas.**

Nota 1: Pueden quedar tablas sin datos a insertar si tienen como objetivo almacenar información que no se requiere en este ejercicio.

Nota 2: Se deben usar consultas a tablas soporte cuando sea posible. En el resto de los casos toda la información podrá ser insertada por el usuario no siendo necesario el mantenimiento de la integridad de los datos en estas inserciones (inconsistencias entre la información almacenada en diferentes tablas). Se permite, si el alumno lo desea, la implementación de tablas soporte adicionales a las requeridas.

### Creación de funciones de actualización de datos (10%)

Se deberán crear funciones que realices las siguientes tareas:

- Actualizar el año de publicación de un libro en base a su ISBN. (Consulta 1) (10%)

Nota: Se deben usar consultas a tablas soporte cuando sea necesaria.

### Creación de funciones de borrado de datos (5%)

Se deberán crear funciones que realices las siguientes tareas:

- Borrar todos los autores que hayan ganado un premio en específico (consulta 7)

### Creación de funciones de consulta de información general (10%)

Realice funciones en las que se pueda consultar la información de las tablas y en los que se pueda filtrar la información a través de una cláusula WHERE según los requisitos originales de las consultas. Se debe recibir del usuario los valores por los que se quiere buscar según lo especificado en la consulta. Si el enunciado de la Actividad 1 no especifica la búsqueda de información a través de un atributo en concreto, se deberá buscar por la partition key de la tabla. En cada consulta se debe mostrar la información de al menos las columnas clave y al menos dos columnas no clave en el caso de que las hubiera.

### Creación de interfaz de interacción de usuario (5%)

Debe crear una interfaz de consola que le pida al usuario realizar alguna de las operaciones programadas en las funciones creadas en los distintos apartados o salir de la aplicación pulsando 0. La elección se realizará a través de un número introducido por el usuario empezando por el 1. Se debe explicar al usuario la operación que se realizará a través de un mensaje por pantalla.

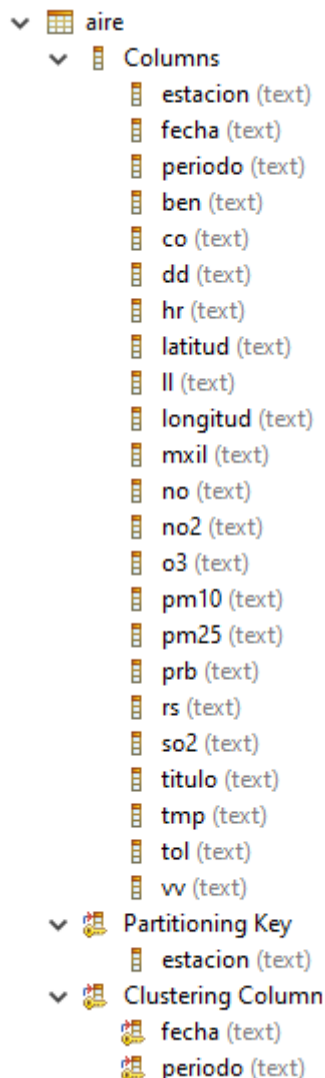
### Sobre penalizaciones con respecto a malos diseños de las tablas

En general no se penalizará que el código trabaje sobre tablas con un diseño incorrecto. Solo se penalizará en el caso de que las tablas diseñadas faciliten de manera excesiva la realización de la actividad 2 evitando el uso de técnicas de manipulación de datos vista en clase.

### Parte 3: Migración de datos (15%)

Descargue el fichero que se encuentra adjunto a este enunciado en el campus virtual denominado “calidadAire.csv”

Este fichero deberá ser migrado a la tabla con el siguiente diseño:



Para crear dicha tabla ejecute el siguiente código CQL:

```
CREATE TABLE aire (estacion text ,titulo text ,latitud text,longitud text,fecha text,periodo text,so2 text, no text, no2 text, co text, pm10 text, o3 text, dd text, vv text, tmp text, hr text,
```

prb text, rs text, ll text, ben text, tol text, mxil text, pm25 text, primary key (estacion, fecha, periodo));

Deberá migrar los datos del fichero a la tabla a través de DataStax Bulk Loader. Deberá entregar un script de las operaciones ejecutados y un archivo PDF en el que se detalle a través de capturas y explicaciones breves las operaciones realizadas.

**NOTA IMPORTANTE: Se comprobarán los códigos entregados por los alumnos usando un software anti plagio. Si se detecta que dos códigos son copiados, se procederá a la calificación de 0 suspenso de la actividad, además de las sanciones posteriores pertinentes.**

### Entrega

Se deben subir a la actividad adjunta el campus virtual los siguientes archivos:

1. Parte 1: Archivo cql que contenga la creación de la estructura de la base de datos con la creación del keyspace y de las tablas.
2. Parte 2: Programa python con las funciones necesarias implementadas y listo para su funcionamiento. **Es importante que la extensión del archivo entregado sea .py.** Si se entrega otro formato que necesite de un software específico para ser analizado, el profesor se reserva el derecho a no valorar la entrega. Se valorarán comentarios explicando el código.
3. Parte 3: Archivo PDF en el que se detalle a través de capturas y explicaciones breves las operaciones realizadas.

**La fecha de entrega es el 29 de octubre de 2024 a las 23:59h.**