

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

Nota introductoria: Algunos ejercicios solo los tiene que hacer uno de los dos grupos creados para la Actividad 1. En estos casos se indica en el ejercicio. Si no se indica, el ejercicio debe ser realizado para ambos grupos.

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:

- Entidad Provincia
 - Jefes Provinciales: Set<text>*
- Zona
 - Municipios: Set<text>*
- Distribuye
 - Cantidad: Float
 - Fecha: Date
- Provee
 - Cantidad: Float
 - Fecha: Date
- Subestación:
 - Capacidad: Float
- Línea:
 - Longitud: Float
- Distribución de red:
 - Longitud máxima: Float
- Productor
 - Media Producción: Float
 - Máximo Producción: Float

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

Parte 2: Programa Python de gestión de datos (80%)

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

Clases por entidad o relación (15%):

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:

- Ambos grupos
 - La entidad Provincia.

- La entidad Productores.

Creación de métodos de inserción de datos (25%)

Debe crear funciones para insertar instancias de entidades o relaciones en todas las tablas donde sea necesario para mantener la integridad de los datos, exceptuando la especificada en la Nota 1. En el caso de insertarse la instancia de una entidad, ésta debe insertarse en las tablas cuya clave primaria esté compuesta de columnas asociadas a atributos de dicha entidad. En el caso de las relaciones, la tabla deberá incluir columnas asociadas a atributos de, exclusivamente, las entidades relacionadas. Toda la información que insertar debe ser provista por el usuario a través de la consola.

Se piden funciones para insertar instancias de las siguientes entidades. Entre paréntesis se indica las consultas de las que se crearon las tablas a insertar los datos:

- Provincia en una tabla para grupo 1 (consulta 1)
- Productores en una tabla para grupo 1 (consulta 8), dos tablas para grupo 2 (consultas 1 y 8).

Se piden funciones para insertar las siguientes relaciones entre instancias:

- Grupo 1:
 - Relación Provee
 - Relación conjunta Consiste-Suple
- Grupo 2
 - Relación Distribuye
 - Relación conjunta Provee-Cabecera

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 (15%)

Se deberán crear funciones que realices las siguientes tareas:

- Actualizar el nombre de una provincia en la tabla que satisfaga la consulta 1 de la actividad 1 (grupo 1).
- Actualizar el origen de la energía que produce un productor en la tabla que satisfaga la consulta 8 de la actividad 1 (grupo 1).
- Actualizar el origen de la energía en la tabla que satisfaga la consulta 1 (grupo 2).
- Actualizar el país de un productor en la tabla que satisfaga la consulta 1 (grupo 2).

Nota: Se deberán realizar las operaciones necesarias para ello. Recuerde que en ocasiones no se puede usar la operación UPDATE cuando haya que actualizar el valor de una clave primaria de una tabla y que se requiere de la combinación REMOVE + INSERT.

Creación de funciones de consulta de información general (15%)

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. 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 el 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.

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 debe subir al campus virtual un fichero comprimido (zip o 7z) que contenga los siguientes dos 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.

La fecha de entrega es el 23 de noviembre a las 23:59h.