ACTIVIDAD 1 - DISEÑO Y OPERACIONES CRUD EN BASES DE DATOS NOSQL

Iván David Caviedes León.

Facultad de Ingeniería, Corporación Universitaria Iberoamericana.

202231: Ingeniería de Software

Mary Rubiano

10 de marzo 2021.

OBJETIVO GENERAL

Se desea desarrollar un modelo de base de datos **NOSQL** que permita la gestión de un torneo deportivo.

REGLAS DEL TORNEO

En esta ocasión el torneo será de Futbol entre diferentes escuelas, cada escuela debe registrar algunos datos de sus participantes y entrenadores. En el torneo Participarán la cantidad de 5 escuelas las cuales tendrán 1 solo encuentro con las otras escuelas al final del torneo los puntos se repartirán de la siguiente manera.

1. Si el equipo gana se dará 2 puntos
2. Si el equipo empata se dará 1 punto
3. Si el equipo pierde no se dará puntos

El ganador del torneo será el equipo con más puntos.

SOLUCION DE MODELO DE BASE DE DATOS

El modelo de datos tendrá que ser acorde con los datos de las siguientes colecciones **(Deportistas, Entrenadores, Árbitros, Encuentros Deportivos, Resultados)**

EQUIPO	
Nombre	**String**
Numero Integrantes	**Intenger**

DEPORTISTA	
Nombres	**String**
Apellidos	**String**
Altura	**Intenger**
Peso	**Intenger**
Edad	**Intenger**
Documento identificación	**Intenger**
Equipo	**ObjectId (Equipo)**

ENTRENADOR	
Nombres	**String**
Apellidos	**String**
Equipo	**ObjectId (Equipo)**
Documento	**Intenger**

ARBITROS	
Nombres	**String**
Apellidos	**String**

Documento	**Intenger**

ENCUENTROS	
Equipo A	**ObjectId (Equipo)**
Equipo B	**ObjectId (Equipo)**
Arbitro	**ObjectId (Arbitro)**
Nombre Encuentro	**String**

RESULTADOS	
Encuentro	**ObjectId (Encuentro)**
Resultado	**Intenger**
Puntos	**Intenger**

TABLA DE POSICIONES	
Equipo	**ObjectId (Equipo)**
Total Puntos	**Intenger**
Partidos Empatados	**Intenger**
Partidos Perdidos	**Intenger**
Partidos Ganados	**Intenger**

Después de realizado el torneo con los datos de resultado se podrá obtener la tabla de posiciones de cada encuentro y el ganador del torneo.

ALGUNOS COMANDOS BASICOS DE MONGODB

1. Para la inserción de un solo documento en la colección se utiliza el método **insertOne()** si dicha colección no existe el método también crea la colección, el documento no tiene un esquema de datos lo que significa que no tiene restricciones en la inserción.
   Sintaxis: nombrecollecion.insertOne({documento})
   Ejemplo:

   db.usuarios.insertOne({
   Nombre: “Nombre usuario”, Edad:20, Estado: “Activo”
   })

2. Para insertar varios registros en una colección se utiliza el método i**nsertMany()**

Sintaxis:
nombrecollecion.insertMany([{documento1},{documento2},{documento3}…{documento}])

Ejemplo:
db.usuarios.insetMany([
{
Nombre: “Nombre usuario1”, Edad:20, Estado: “Activo”
},
{
Nombre: “Nombre usuario2”, Edad:20, Estado: “Activo”
},
{
Nombre: “Nombre usuario3”, Edad:20, Estado: “Activo”
}
])

3. Para poder recuperar los datos de una colección se utiliza el método **find()** se utiliza para recuperar todos los documentos de la colección.

   Sintaxis: nombrecollecion.find()

   Ejemplo: db.usuarios.find()

   {Nombre: “Nombre usuario1”, Edad:20, Estado: “Activo”},
   {Nombre: “Nombre usuario2”, Edad:20, Estado: “Activo”},
   {Nombre: “Nombre usuario3”, Edad:20, Estado: “Activo”}

4. Para organizar la consulta del método **FIND** ya que es desorganizada se utiliza el método **pretty()** se nos proporciona una salida formateada para su mejor entendimiento
   Sintaxis: nombrecollecion.find().pretty()

   Ejemplo: db.usuarios.find().pretty()

   {
   Nombre: “Nombre usuario1”,
   Edad:20,
   Estado: “Activo”
   },
   {
   Nombre: “Nombre usuario2”,

Edad:20,
Estado:” Activo”

},

{
Nombre: “Nombre usuario3”,
Edad:20,
Estado: “Activo”
}

ENLACE DEL VIDEO: https://www.youtube.com/watch?v=rF1odcvyb-U