

Universidad Nacional de La Matanza



Realtime Database

Intro

- ¿Qué conocimientos previos tienen sobre bases de datos?

¿Qué es una base de datos?

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

No es lo mismo:

- Dato: Valor.
- Información: Dato al que se le da un contexto, el cual lo hace útil.

Operaciones de una BBDD

- Insertar
- Borrar
- Modificar
- Consultar

¿Cómo se usan en desarrollo *mobile*?

Existen dos formas de almacenar información en BBDD cuando desarrollamos apps:

- Bases de datos remotas (en un servidor).
- Bases de datos locales (en el teléfono/tablet/dispositivo del cliente).

¿Dónde deberían almacenarse los datos?

Depende del uso que se les quiera dar...

Tipos de Bases de Datos

- Relacionales
 - La información se almacena en tablas, columnas y registros.
 - Tienen un lenguaje de consulta potente que permite obtener datos a través de relaciones entre sí.
 - Permiten evitar redundancia de datos.
 - Ejemplos: MySQL, PostgreSQL.
- No relacionales
 - La información se guarda en *documentos* u *objetos*.
 - Generalmente más veloces.
 - Ejemplos. SQLite, MongoDB.

Ejemplo de SQL

```
SELECT * from Partidas WHERE retador = 'F2gLlt1992cFgZBMuVnl43UHPwk2';
```


Ejemplo Firebase Realtime Database

```
FirebaseDatabase.getInstance().reference.child(Constantes.TABLA_PARTIDAS);
```

Tablas vs Documentos

PARTIDAS			
id	ganador	oponente	retador
-L-NoOmdPD8rvHmPzDG	F2gLlt1992cHgZBMuVnl43UHPwk2	0zMtfxzbOzR5Tvby6Uz6HoMpkUN2	F2gLlt1992cHgZBMuVnl43UHPwk2
MOVIMIENTOS			
partida_id	jugador	posicion	
-L-NoOmdPD8rvHmPzDG	F2gLlt1992cHgZBMuVnl43UHPwk2	5	
-L-NoOmdPD8rvHmPzDG	0zMtfxzbOzR5Tvby6Uz6HoMpkUN2	6	
-L-NoOmdPD8rvHmPzDG	F2gLlt1992cHgZBMuVnl43UHPwk2	7	
-L-NoOmdPD8rvHmPzDG	0zMtfxzbOzR5Tvby6Uz6HoMpkUN2	9	
-L-NoOmdPD8rvHmPzDG	F2gLlt1992cHgZBMuVnl43UHPwk2	3	

```
"PARTIDAS" : {
  "-L-NoOmdPD8rvHmPzDG" : {
    "ganador" : "F2gLlt1992cHgZBMuVnl43UHPwk2",
    "movimientos" : [ {
      "jugador" : "F2gLlt1992cHgZBMuVnl43UHPwk2",
      "posicion" : 5
    }, {
      "jugador" : "0zMtfxzbOzR5Tvby6Uz6HoMpkUN2",
      "posicion" : 6
    }, {
      "jugador" : "F2gLlt1992cHgZBMuVnl43UHPwk2",
      "posicion" : 7
    }, {
      "jugador" : "0zMtfxzbOzR5Tvby6Uz6HoMpkUN2",
      "posicion" : 9
    }, {
      "jugador" : "F2gLlt1992cHgZBMuVnl43UHPwk2",
      "posicion" : 3
    } ],
    "oponente" : "0zMtfxzbOzR5Tvby6Uz6HoMpkUN2",
    "retador" : "F2gLlt1992cHgZBMuVnl43UHPwk2"
  }
}
```

JSON

- JavaScript Object Notation
- Formato de texto sencillo usado para la serialización de objetos
- Serializar: Transformar un objeto a algo que se pueda transmitir.
- Deserializar: Transformar algo, cómo ser un texto, en un objeto.
- Es uno de los tantos estándares posibles para transmitir datos entre cliente y servidor.

JSON: Tipos de datos permitidos

- Objetos: Entre llaves { }
- Arrays: Entre corchetes []
- Atributos
 - Booleans: true/false
 - Strings: entre comillas “
 - Números enteros
 - Números con decimal

Queda en cada uno definir si admite **null** o no cada tipo.

Intro



Casos de Uso

- Almacenar datos en la nube.
- Mantener sincronizada una base de datos local con una remota.
- Recibir actualizaciones en tiempo real de los datos de la BBDD.

Características

- Almacena los datos en la consola de Firebase.
- Auto-escalable.
- No relacional.
- Caché automático.
- Los datos se sincronizan cuando hay conexión.
- Opciones de seguridad por roles de usuario, etc.
- Se accede a los datos con una ruta similar a una URL.

Integración

- Integrar Firebase Realtime Database modificando los archivos `build.gradle` y agregando el `google-services.json`, de acuerdo a lo visto en clases anteriores.

En el archivo build.gradle a **nivel app**

```
dependencies {  
    // Dependencia de Firebase  
    implementation platform('com.google.firebase:firebase-bom:27.1.0')  
  
    // Dependencia de Realtime Database  
    implementation 'com.google.firebase:firebase-database:19.7.0'  
    ...  
  
    // Otras dependencias  
    ...  
}
```

En el código accedemos a Firebase Realtime Database de la siguiente manera:

```
FirebaseDatabase.getInstance().reference
```

```
FirebaseDatabase.getInstance()  
    .reference  
    .child(Constantes.TABLA_PARTIDAS)  
    .addChildEventListener(listenerTablaPartidas)
```

Integración - Leyendo un nodo

```
private val listenerTablaPartidas: ChildEventListener = object : ChildEventListener {  
    override fun onChildAdded(dataSnapshot: DataSnapshot, s: String?) {  
        ...  
    }  
  
    override fun onChildChanged(dataSnapshot: DataSnapshot, s: String?) {  
        ...  
    }  
  
    override fun onChildRemoved(dataSnapshot: DataSnapshot) {  
        ...  
    }  
  
    override fun onChildMoved(dataSnapshot: DataSnapshot, s: String?) {  
        ...  
    }  
  
    override fun onCancelled(databaseError: DatabaseError) {  
        ...  
    }  
}
```

Objetos clave

- FirebaseDatabase
- DatabaseReference
- ChildEventListener
- DataSnapshot

Bibliografía

- [Documentación Firebase Realtime Database](#)

¿Preguntas?