



Unidade 4

Seção 2



# Desenvolvimento para Dispositivos Móveis

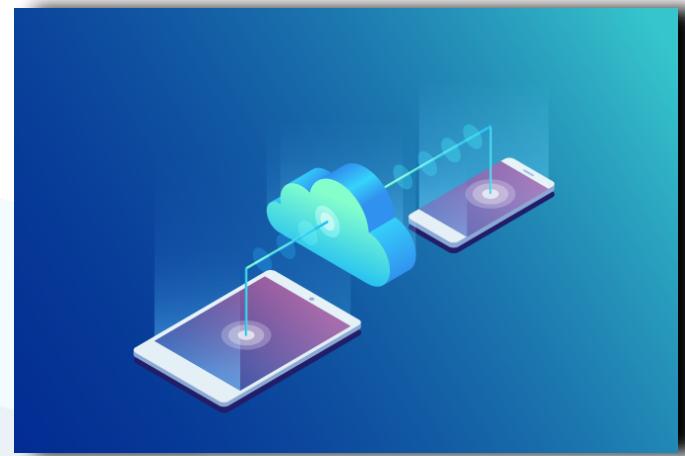
# Weaula 2

Introdução ao banco de dados na nuvem



Nesta webaula você vai ver como conectar os aplicativos Android à nuvem por meio da plataforma Firebase.

Atualmente, os aplicativos geram grandes volumes de dados de seus usuários. Os dados são armazenados na **nuvem** e estão disponíveis sempre que o usuário precisar. Os aplicativos Android são capazes de manter os dados do usuário sincronizados em todos os seus dispositivos móveis. Desta forma, é possível compartilhar as suas informações entre diferentes dispositivos móveis ao utilizar o mesmo aplicativo.



**Firebase** para Android é a plataforma da empresa Google que oferece produtos para desenvolver aplicativos de alta performance. De acordo com a Google (2018), a plataforma disponibiliza diversos produtos que podem ser combinados para oferecerem mais funcionalidades para os usuários, como:

#### Realtime Database

É um banco de dados hospedado na nuvem. Os dados são armazenados como JSON e sincronizados em tempo real com todos os clientes conectados.

#### Authentication

É um serviço de autenticação de usuários. Ele oferece suporte à autenticação por meio de senhas, números de telefone e provedores de identidade federados, como Google, Facebook, Twitter e muito mais.

#### Cloud Storage

Método responsável por informar ao *Adapter* a quantidade total de itens que devem ser exibidos.

## Realtime Database

O Realtime Database é um banco de dados [NoSQL](#) hospedado na nuvem. Os dados são armazenados como JSON e sincronizados em tempo real com todos os clientes conectados. No Realtime Database, não existem tabelas e registros. A sua estrutura é composta de *key-value* (chave-valor).

Por meio do banco de dados Realtime Database, o programador conseguirá desenvolver aplicativos Android capazes de armazenar dados na nuvem que estarão disponíveis para todos os usuários conectados ao aplicativo. (AZURE, 2018).

Para entender como o Realtime Database disponibiliza os dados armazenados para os usuários, clique nos botões dos seguintes aplicativos:

### Mensagens instantâneas

Esse aplicativo deve ser capaz de enviar e receber mensagens em tempo real. Cada mensagem enviada será armazenada no banco de dados e automaticamente será disponibilizada para todos os usuários conectados ao aplicativo.

### Rede social em dispositivo móvel

O usuário recebe atualizações de novas publicações a todo momento no *feed* de notícias. Se o usuário publicar alguma informação, ela será compartilhada com todos os usuários que também estão conectados a esse *feed* de notícias.

Para iniciar o desenvolvimento de um aplicativo com o Realtime Database, o Android Studio disponibiliza a janela Firebase Assistent, responsável por auxiliar o programador a conectar o projeto do Android Studio à plataforma Firebase.

**Janela Assistent Firebase**

 **Firebase**

Firebase gives you the tools and infrastructure from Google to help you develop, grow and earn money from your app. [Learn more](#)

---

▶  **Authentication**  
Sign in and manage users with ease, accepting emails, Google Sign-In, Facebook and other login providers. [More info](#)

▼  **Realtime Database**  
Store and sync data in realtime across all connected clients. [More info](#)

▶  [Save and retrieve data](#)

Fonte: captura de tela do Android Studio.

A Google (2018) garante que planejar como os dados serão salvos e recuperados tornam o processo de desenvolvimento o mais fácil possível. Como já vimos, o Realtime Database armazena os dados como JSON. Portanto, no Realtime Database não existem tabelas e registros. A sua **estrutura** é composta de key-value (chave-valor).

### Informações armazenadas – estrutura Realtime Database

raizDoProjeto:

+ produtos:

+ -LFUXkJLOgdPKYw7JINs

descricao: "Smartphone"

preco: 1800

+ -LFUXkb5xcNyAISw3k2d

descricao: "Tablet"

preco: 2200

Fonte: elaborada pelo autor.

Para adicionar dados no Realtime Database, você pode enviar ao banco de dados uma das seguintes opções:

1) Objeto Java personalizado, que atenda às seguintes características:

a) Possua um construtor que não receba parâmetros;

b) Possua Getters públicos para cada atributo.

Clique aqui para saber mais.

2) Dados compatíveis aos tipos JSON disponíveis:

a) String

b) Long

c) Double

d) Boolean

e) Map<String, Object>

f) List<Object>

Saiba Mais



Ao enviar um objeto Java para o Realtime Database, os atributos deste objeto serão convertidos para a estrutura JSON automaticamente.

Também é possível recuperar um único objeto ou uma coleção de objetos do Realtime Database. Quando recuperarmos uma coleção de objetos do Realtime Database, devemos garantir que sejam identificados quais objetos são iguais para que o aplicativo não apresente comportamentos inesperados pelo usuário.

Portanto, sobreponha os métodos *equals()* e *hashCode()* da classe Java.

Para ilustrar, criaremos um banco de dados na nuvem usando o Realtime Database para podermos sincronizar as informações do usuário em qualquer dispositivo móvel. Explore a galeria para ver a estrutura da classe Java para o objeto Produto.

```
1 package com.exemplo.unidadeiv;
2
3 import java.util.Objects;
4
5 class Produto {
6
7     private String id;
8     private String descricao;
9     private float preco;
10
```

Fonte: elaborada pelo autor.





Nesta webaula você viu a como incluir o Realtime Database em um projeto do Android Studio.





13



# Bons estudos!

14

