

5. Using MDM (Enterprise deployment)

- Large companies के लिए - internal apps deploy करने के लिए use होती है schools, offices में internal app push करने के लिए।

6. Deployment via google play store

- APK के बजाय AAB file recommend होती है।

लेज़ीन पहले play store में भी APK file में ही deploy होती है।

Deployment methods

(APK file को install करने के तरीके)

1. Direct install on Device (manual installation)

step 1 : Android studio से APK generate करो

step 2 - phone से USB से connect करो

step 3 - APK को phone में copy करो

step 4 - install from "unknown sources" enable करो

steps - APK tap करके install करो।

2. via Android studio (Debugging proses)

- phone को USB से connect करो।

- Run button दबाओ (Android studio में)

- App directly install हो जाता है -
connected phone में

3. Via Email / WhatsApp / Link

- APK file को किसी को भेजो

- वो उसको download करके install करे।

4. Deploying on website

- APK को किसी website पर host करो।

- Users वहां से download करके install कर सकते हैं

5. Upload AAB file
 - Release > Production > create new release
 - Upload .aab file
 - Add release name and notes

6. App content details fill करो
 - Privacy policy
 - Age rating
 - App access permissions
 - Data collection info (Google form fill करना होता है)

7. Submit App for review

- Google app को check करती है
- Review के बाद app publish हो जाता है

→ Deploying APK files

APK (Android package) file को किसी device पर install करने, खोलना या distribute करना ही deployment कहलाता है।

what is APK

APK file Android app का final output होती है - जो .apk extension में होती है, जैसे MyApp.apk

Date _____
Page _____

steps to publish Android App (Android studio through)

1. App complete करें।
 - All features test करें।
 - Bugs fix करें।
 - UI clean

2. Generate signed APK / AAB (Bundle)

APK - Android package, AAB - Android App Bundle

steps in android studio

- Build > Generate signed Bundle / APK
- Choose Android app Bundle
- create new keystore
- fill credentials and generate .aab file

3. Create google play Developer account

4. Create app in play console

- App का name, language, category icon, screenshot, etc, fill करें।

- Add description in english + hindi if needed

Operations	meaning	method used
Insert	Add	insert ()
Select / fetch	read	query ()
Update	modify	update ()
Delete	Remove	delete ()

code snippets

Insert
`db.insert("table", null, values);`

Select / fetch
`db.query(...);`

Update
`db.update("table", values, ...);`

Delete
`db.delete("table", ...);`

→ Publishing android applications

Publishing ka matlab होता है
अपने android app को users के
लिए available बनाना, जैसे Google
play store पर upload करने

Advantage

- simple structure, easy to use
- Fast performance
- offline में काम करता है

Limitations

- large scale apps के लिए suitable नहीं
- security manually manage करनी पड़ती है

→ Connecting with SQLite Database and operations

Insert, Delete, update, fetch

Android में SQLite database से connect करने के लिए हम एक helper class बनाते हैं जो SQLiteOpenHelper class को extend करती है

step 1 - Helper class बनाना

step 2 - Basic CRUD operations

(A) Insert operation - Data add करना
/FETCH

(B) select operations - Data read करना

(C) update operations - Data modify करना

(D) delete operations - Data Delete करना

Features

1. बहुत ही हल्का और fast होता है
2. इसमें कोई server install करने की जरूरत नहीं होती है
3. यह SQL (structured query language) support करता है

SQLite Architecture

SQLite में 3 main components होते हैं

1. Database :- जिसमें tables होती हैं
2. Tables :- जिसमें rows & columns में Data होता है
3. Queries :- SQL के through Data update, read, insert, delete करते हैं

Important commands in SQLite

Command	Use
CREATE	Table बनाना
INSERT	Data add करना
SELECT	Data read करना
UPDATE	Data modify करना
DELETE	Data हटाना

Important Functions in content provider

method	work
query()	Data read करने के लिए
insert()	नया Data add करने के लिए
update()	Existing data change करने के लिए
delete()	Data delete करने के लिए
getType()	Data का type बताने के लिए

→ The SQLite database
SQLite एक lightweight relational Database है जो Data को device के अंदर ही store करता है। बिना किसी server के, ये छोटी apps के लिए fast और best database है।

Android में SQLite का use

जब हमें app में user का data store करना हो - जैसे login info, notes, shopping list, attendance etc.

ये Data device के internal storage में db file के form में save होता है।

→ Content provider
Content provider android में एक system होता है जो एक app का data दूसरी app के साथ share करता है एक secure और controlled way में.

Example:-

सोचो एक app contacts store करता है-
(जैसे phone app) और दूसरा app (whatsapp) उन contacts की access करना चाहता है।

→ ये काम content provider के through होता है.

Data कहा store होता है-

content provider database (sqlite), files, या web से data ले सकता है.

Why use content provider

Purpose	Description
Data sharing	App 1 का data App 2 के साथ share करना.
Secure Access	Unauthorized access से Data बचाता है.
Structured Data Access	Query, Insert, update, Delete कर सकते हैं.

Mobile Apps के context में
Android apps में external storage का use होता है -

- Photos, videos, Downloads जैसे data के लिए
- User खुद भी access कर सकते हैं
- ये shared होती है - multiple apps use कर सकते हैं

Features of external storage

Feature	Detail
Type	Non - Volatile (Permanent)
Access	User & multiple apps
security	Low (user can modify/delete)
Capacity	बड़ा (SD card, USB drive)
use	media files, downloads, backups etc

Internal storage

निजी उपयोग के लिए

1. सिर्फ app या system
2. Security - high
3. storage size - limited

External storage

पब्लिक उपयोग के लिए

- User + multiple apps
- security - low
- storage size (SD card, USB)

Internal storage के Features

Feature	Detail
1. Type	Non-Volatile (Permanent)
2. Access	sinf app या system
3. Security	High (app-specific)
4. Capacity	Limited (device के storage के हिसाब से)
5. Use	small and security

→ External storage

External storage का मतलब होता है- जी Device के बाहर या user-accessible जगह जहाँ हम data को store कर सकते हैं- जैसे SD card, USB drive, या mobile का external folder (Downloads, DCM)

External storage वो memory होती है- जिसे user easily access, modify या delete कर सकता है- जो storage non-volatile होती है, मतलब power off होने के बाद भी data delete नहीं होता।

→ Internal storage

Internal storage का मतलब होता है device की अन्दर वाली memory जहाँ data को permanently (persistently) store किया जा सकता है - जैसे mobile phone, tablet के अन्दर

Internal storage वो जगह होती है जहाँ apps, user data, और system files store की जाती हैं।

ये non-volatile memory होती है - मतलब power off होने के बाद भी data delete नहीं होता है।

Mobile apps के context में

आप android app बना रहे हो तो

- Internal storage - app के अन्दर का secure storage.
- सीर्फ उसी app को access मिलता है।
- बाकी apps या user manually access नहीं कर सकता

use करते करते हैं।

- जब data secure रखना हो
- जब data होता हो, जैसे login details, App setting, Notes, drafts etc

(b) NoSQL Database (Non-relational, flexible)

जब Data structure fix नहीं है।

Ex:- MongoDB, Firebase

Ex:- Social media comments या user profile data

3. cloud storage -

Data को internet पर store किया जाता है।
secure होता है कहीं से भी access कर सकते हैं।

Ex:- Google Drive

4. Local storage -

Mostly web या mobile apps के लिए होता है।
user के device में होता है Data store होता है।

Ex:- Browser का local storage.

5. External storage Devices

भाज भी backup या offline storage के लिए
use होता है।

Pen drive, Hard Disk, memory card etc

Option	Use case
1. File storage	Simple Data
2. SQL Database	structured, relational
3. NoSQL database	flexible, unstructured Data
4. Cloud storage	Online access & backup
5. Local/app storage	Temporary device storage
6. External Devices	offline manual storage

Unit - 5

→ storing the data persistently - Data storage options.

Persistent Data storage options का मतलब होता है - ऐसा data जो device में permanent या long time तक save रहे, चाहे power off हो जाए फिर भी वो delete ना हो, ऐसे data को हम permanent memory में store करते हैं-

Persistent Data store करने के options.

Or Preferences

1. File storage

Data को text files या binary files के form में store किया जाता है

Ex:- txt, csv

use होता है जब data simple और छोटा है

2. Database

जब Data बड़ा हो या structured हो, तब Database use किया जाता है

Data को tables में row-column format में store करते हैं.

a. Relational Databases (जैसे SQL)

use करते हैं - जब Data के बीच relation हो

Ex- my SQL, PostgreSQL, SQLite

Ex- student table or course table के बीच relationship.