Assignment - VI

Problem statement :-

Design a mobile app for media player to store data using internal or external storage.

Pre requisite :-

1) Basic concept of internal or external storage.
2) Basic concept of internal & external memory.

S/W & H/W :-

Android Studio, 4GB RAM, 64 bit OS,

Objective :-

Implement app to store data using internal or external storage.

Outcome :-

After completion of this assignment student are able to implement app to store data using internal or external storage.

Theory :-

Actually, the first thing you do is create an activity. These are where all the action happens, because they are the screen that allow user to interact with your app. In short activities are one of the basic building blocks of Android application. The process for creating, starting & stopping an activity & handle navigation betn activities. The various stages in lifecycle of an activity & how to handle each stage gracefully. The way to manage configurations changes & persist data within your activity.

- Android Preference Example :-

Android shared preference is used to store & retrive primitive information. In Android, string, long, integer, number, etc. are considered as primitive data types.

It is used to store data in key & value pair so that we can retrive the value on basis of key.

Android provides many binds of storage for applications to store their data. These storage places are shared preference, internal & external storage, sqlite storage, and storage via network connection. It is widely used to get information from user such as in settings.

- Android Internal storage Example :-

We are able to save or read data from device internal memory. File Input Stream & File Output Stream classes are used to read & write data into file.

In order to use internal storage to write some data in file call the open File Output() method with the name of file & mode. The mode could be private, public, etc. The syntax is,

FileOutputStream fout = openFileOutput(
"file name here", MODE_WORLD-READABLE);

Apart from the methods of read & close; there are other method provided by File-InputStream class for better reading files. These methods are listed below.

1) available()

This method returns an estimated number of bytes that can be read or skipped without blocking for more input.

2) getchannel():-

This method returns a read only file channel that shares position within this stream.

3) getFD() :-

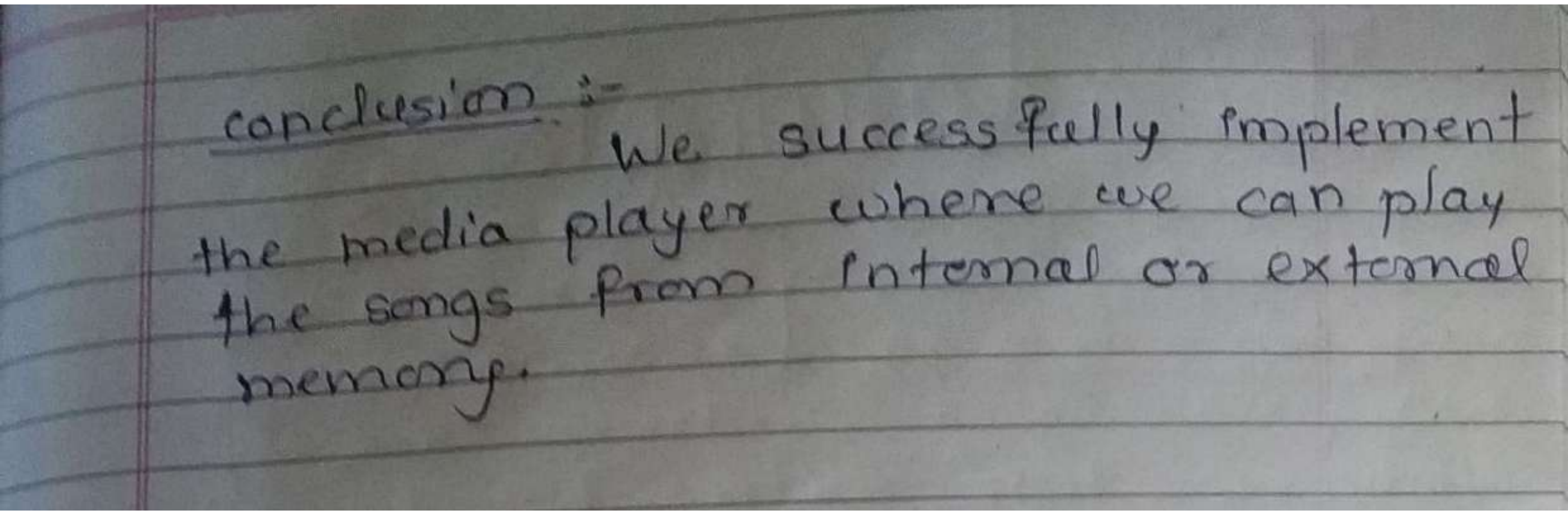
This method returns the underlying file descriptor.

4) read (byte [] buffer, int byte offset, int bytecount)

This method reads at most length bytes from this ~~sy~~ stream & stores them in the byte array b starting at offset.

Android External Storage :-

Like internal storage, we are able to save or read data from the device external memory such as sdcard. The file InputStream & FileOutputStream classes are used to read & write data into the file.



conclusion :-

We successfully implement the media player where we can play the songs from internal or external memory.