```kotlin
/*As the name suggests, this function is used to fetch the songs present in your phones and
returns the arraylist of the same*/
fun getSongsFromPhone(): ArrayList<Songs> {
    var arrayList = ArrayList<Songs>()

    /*A content resolver is used to access the data present in your phone
    * In this case it is used for obtaining the songs present your phone*/
    var contentResolver = myActivity?.contentResolver

    /*Here we are accessing the Media class of Audio class which in turn a class of Media
Store, which contains information about all the media files present
    * on our mobile device*/
    var songUri = MediaStore.Audio.Media.EXTERNAL_CONTENT_URI

    /*Here we make the request of songs to the content resolver to get the music files from
our device*/
    var songCursor = contentResolver?.query(songUri, null, null, null, null)

    /*In the if condition we check whether the number of music files are null or not. The
moveToFirst() function returns the first row of the results*/
    if (songCursor != null && songCursor.moveToFirst()) {
        val songId = songCursor.getColumnIndex(MediaStore.Audio.Media._ID)
        val songTitle = songCursor.getColumnIndex(MediaStore.Audio.Media.TITLE)
        val songArtist = songCursor.getColumnIndex(MediaStore.Audio.Media.ARTIST)
        val songData = songCursor.getColumnIndex(MediaStore.Audio.Media.DATA)
        val dateIndex = songCursor.getColumnIndex(MediaStore.Audio.Media.DATE_ADDED)

        /*moveToNext() returns the next row of the results. It returns null if there is no
row after the current row*/
        while (songCursor.moveToNext()) {
            var currentId = songCursor.getLong(songId)
            var currentTitle = songCursor.getString(songTitle)
            var currentArtist = songCursor.getString(songArtist)
            var currentData = songCursor.getString(songData)
            var currentDate = songCursor.getLong(dateIndex)

            /*Adding the fetched songs to the arraylist*/
            arrayList.add(Songs(currentId, currentTitle, currentArtist, currentData,
currentDate))
        }
    }

    /*Returning the arraylist of songs*/
    return arrayList
}
```