

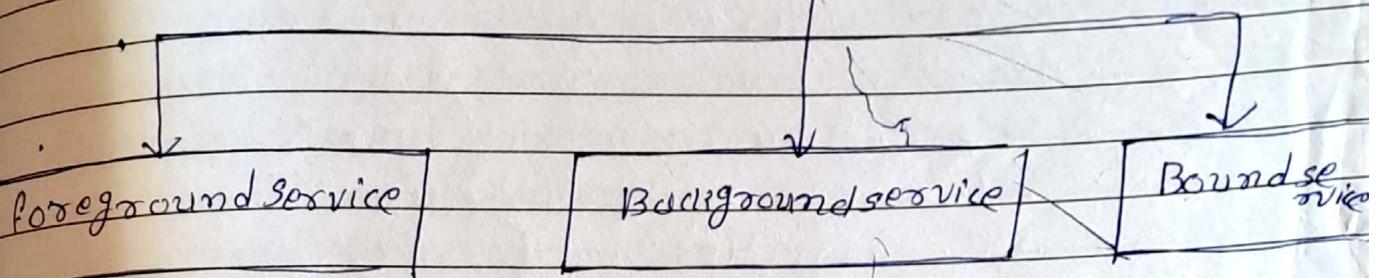
## Practicals

What is service?

Service are code that runs in the background.  
They can be started and stopped, services doesn't have UI.

Types of service?

### TYPES OF SERVICES



Foreground: Foreground services require user interface.  
Background: Background services do not require any user intervention.

Bound: This type of background service allows the components of the application like activity to bind themselves with it.

activity\_main.xml

```

<?xml version = "1.0" encoding = "UTF-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
  xmlns:android = "http://schemas.android.com/apk/res/android"
  xmlns:app = "http://schemas.android.com/apk/res-auto"
  xmlns:tools = "http://schemas.android.com/tools">
  
```

```

    <Activity ...>
        <View ...>
            <!-- Some layout code here -->
    </Activity>
  
```

```

<!-- Some code here -->

```

android: layout-width = "match-parent"

android: layout-height = "match-parent"

tools: context = ".mainActivity")

(com.google.android.material.card.CardView)

android:id = "@+id/img-cardview"

android: layout-width = "match-parent"

android: layout-height = "wrap-content"

app: layout-constraintTop-toTopOf = "parent"

app: layout-constraintStart-toStartOf = "parent"

android: layout-marginStart = "30dp"

android: layout-marginEnd = "30dp"

android: layout-marginTop = "50dp"

app: cardElevation = "20dp"

app: cardCornerRadius = "15dp")

(androidx.constraintlayout.widget.ConstraintLayout

android: layout-width = "match-parent"

android: layout-height = "match-parent")

(com.google.android.material.card.CardView)

android: layout-width = "300dp"

android: layout-height = "300dp"

app: layout-constraintTop-toTopOf = "parent"

app: layout-constraintStart-toStartOf = "parent"

app: layout-constraintEnd-toEndOf = "parent"

app: layout-constraintBottom-toBottomOf = "parent"

android: layout-marginTop = "10dp"

android: layout-marginBottom = "10dp")

app: cardCornerRadius = "25dp")

<imageview>

    android:layout\_width = "match\_parent"

    android:layout\_height = "match\_parent"

    android:src = "@drawable/song"

/1

</com.google.android.material.card.CardView>

<View>

    android:layout\_width = "30dp"

    android:layout\_height = "120dp"

    android:background = "@drawable/design\_view2"

    app:layout\_constraintEnd\_toEndOf = "parent"

    app:layout\_constraintTop\_toTopOf = "parent"

    app:layout\_constraintBottom\_toBottomOf = "parent"

    android:layout\_marginEnd = "-10dp"/>/1

<View>

    android:layout\_width = "30dp"

    android:layout\_height = "120dp"

    android:background = "@drawable/design\_view2"

    app:layout\_constraintStart\_toStartOf = "parent"

    app:layout\_constraintTop\_toTopOf = "parent"

    app:layout\_constraintBottom\_toBottomOf = "parent"

    android:layout\_marginStart = "-10dp"/>/1

</com.google.android.material.card.CardView>

</com.google.android.material.card.CardView>

    android:id = "@+id/control\_cardview"

    android:layout\_width = "match\_parent"

android: layout\_height = "wrap\_content"  
app: cardElevation = "20dp"  
app: layout\_constraintEnd\_toEndOf = "parent"  
app: layout\_constraintStart\_toStartOf = "parent"  
app: layout\_constraintBottom\_toBottomOf = "parent"  
android: layout\_marginBottom = "-5dp")

<androidx.constraintlayout.widget.ConstraintLayout  
    android: layout\_width = "match\_parent"  
    android: layout\_height = "match\_parent")

<androidx.constraintlayout.widget.ConstraintLayout  
    android: id = "@+id/song\_layout"  
    android: layout\_width = "match\_parent"  
    android: layout\_height = "wrap\_content"  
    android: padding = "10dp"  
    app: layout\_constraintBottom\_toTopOf = "@+id/view"  
    app: layout\_constraintStart\_toStartOf = "parent"  
    app: layout\_constraintTop\_toTopOf = "parent")

## <TextView

    android: id = "@+id/song\_text"  
    android: layout\_width = "wrap\_content"  
    android: layout\_height = "wrap\_content"  
    android: text = "song name"  
    android: textSize = "24dp"  
    app: layout\_constraintBottom\_toBottomOf = "parent"  
    app: layout\_constraintEnd\_toEndOf = "parent"  
    app: layout\_constraintStart\_toStartOf = "parent"  
    app: layout\_constraintTop\_toTopOf = "parent")

<androidx.constraintlayout.widget.ConstraintLayout>

<View

```
    android:id="@+id/view_2"
    android:layout-width="10dp"
    android:layout-height="120dp"
    android:background="@drawable/design_view_3"
    app:layout-constraintBottom-toBottomOf="Parent"
    app:layout-constraintStart-toStartOf="Parent"
    app:layout-constraintEnd-toEndOf="Parent"/>
```

<androidx.constraintlayout.widget.ConstraintLayout>

```
    android:layout-width="wrap_content"
    android:layout-height="wrap-content"
    android:layout-marginTop="30dp"
    android:layout-marginBottom="10dp"
    app:layout-constraintBottom-toBottomOf="parent"
    app:layout-constraintEnd-toEndOf="parent"
    app:layout-constraintStart-toStartOf="parent"
    app:layout-constraintTop-toBottomOf="@+id/song_layout"/>
```

<com.google.android.material.FloatingActionButton>

```
    android:id="@+id/btn-play"
    android:layout-width="wrap-content"
    android:layout-height="wrap-content"
    android:backgroundTint="@color/white"
    android:src="@drawable/ic-baseline-play-circle-24"
    app:elevation="10dp"/>
```

app:layout\_constraintTop\_toTopOf = "parent"

app:layout\_constraintBottom\_toBottomOf = "parent"

app:layout\_constraintStart\_toEndOf = "@+id/btn -

app:layout\_constraintEnd\_toEndOf = "previous"

app:layout\_constraintEnd\_toStartOf = "@+id/btn -  
stop + 1"

android:layout\_marginStart = "5dp"

1>

com.google.android.material.FloatingActionButton •

FloatingActionButton

android:id = "@+id/btn\_stop"

android:layout\_width = "wrap\_content"

android:layout\_height = "wrap\_content"

android:backgroundTint = "@color/white"

android:src = "@drawable/ic\_baseline\_stop\_24"

app:elevation = "0dp"

app:layout\_constraintStart\_toEndOf = "@+id/btn\_play"

app:layout\_constraintEnd\_toStartOf = "@+id/btn\_next"

app:layout\_constraintTop\_toTopOf = "parent"

app:layout\_constraintBottom\_toBottomOf = "parent"

android:layout\_marginStart = "5dp" />

com.google.android.material.FloatingActionButton.

FloatingActionButton

android:id = "@+id/btn\_next + 1"

android:layout\_width = "wrap\_content"

android:layout\_height = "wrap\_content"

android:backgroundTint = "@color/white"

android:src = "@drawable/ic\_baseline\_keyboard\_arrow\_right\_24"

app: elevation = "10dp"

app: layout\_constraintBottom\_toBottomOf =  
"parent"

app: layout\_constraintEnd\_toStartOf="@+id/btn  
play"

app: layout\_constraintStart\_toEndOf = "parent"

app: layout\_constraintTop\_toTopOf = "parent" />

(findViewById(R.layout.widget).constraintLayout  
layout)

(com.google.android.material.card.MaterialCardView)

(findViewById(R.layout.widget).constraintLayout  
layout)

MainActivity.kt

package com.example.madpracticals\_21012022046

import android.content.Intent

import android.os.Bundle

import android.widget.TextView

import androidx.appcompat.app.AppCompatActivity

import com.google.android.material.FloatingActionButton

button, FloatingActionButton

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

7

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val btnPlay: FloatingActionButton = findViewById(R.id.btn\_play)  
(R.id.btn\_play)

val btnStop: FloatingActionButton = findViewById(R.id.btn\_stop)  
(R.id.btn\_stop)

val btnNext: FloatingActionButton = findViewById(R.id.btn\_next)  
(R.id.btn\_next)

val btnPrevious: FloatingActionButton = findViewById(R.id.btn\_previous)  
(R.id.btn\_previous)

val txtSong: TextView = findViewById(R.id.song\_text)

txtSong.text = "No song playing"

var i = 0

btnPlay.setOnClickListener {

txtSong.text = "Song"

play()

if (i == 0)

2

btnPlay.setImageResource(R.drawable.ic\_media\_pause)

i = 1

3

else 2

btnPlay.setImageResource(R.drawable.ic\_media\_play)

i = 0

3

btnnext.setonClickListener {

btnprevious.setonClickListener {

}

}

fun play ()

{

Intent intent = new Intent(getApplicationContext(),  
MediaService.class);

.putExtra(MediaService.DATA\_KEY, mediaService,  
DATA\_VALUE).startService(this);

}

}

fun stop ()

{

Intent intent = new Intent(getApplicationContext(),  
MediaService.class); intent.putExtra(MediaService.

STOP\_SERVICE).startService(this);

}

}

medibservice.kt

package com.example.medipracticals.z1012022022;

```
import android.app.Service  
import android.content.Intent  
import android.media.MediaPlayer  
import android.os.IBinder
```

```
class medibservice : Service() {  
    companion object {  
        public final val DATA_KEY = "service"  
        public val DATA_VALUE = "play/pause"  
    }  
}
```

```
private lateinit var player: MediaPlayer  
override fun onBind(intent: Intent): IBinder {  
    TODO("Return the communication channel to  
    the service.")  
}
```

```
override fun onStartCommand(intent: Intent?,  
    flags: Int, startId: Int): Int {  
    if (intent != null) {  
        if (intent.getStringExtra(DATA_KEY) == null) {  
            return START_STICKY  
        }  
    }  
    return START_STICKY  
}
```

```
if (intent.getStringExtra(DATA_KEY) == null) {  
    player = MediaPlayer.create(this, R.raw.song)  
}
```

```
if (player.isPlaying) {  
    player.pause()  
}
```

else {

player.start();

}

return START\_STICKY;

}

override fun onDestroy() {

player.stop();

super.onDestroy();

}