|  |
| --- |
| **학번 : 201911204, 이름 : 이주형** |
| **2023년 13주차 실습 체크 리스트** |
| |  |  | | --- | --- | | **실습 1** | **완성** | | **실습 2** | **완성** | |
| **실습 1 : [ Service ]** |
| **activity\_main.xml** |
| <?xml version="1.0" encoding="utf-8"?> <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools"  android:orientation="vertical"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity">   <ListView  android:id="@+id/listview"  android:entries="@array/songlist"  android:layout\_width="match\_parent"  android:layout\_weight="1"  android:layout\_height="0dp"/>   <ProgressBar  android:id="@+id/progressBar"  android:layout\_width="match\_parent"  android:layout\_height="40dp"  style ="@style/Widget.AppCompat.ProgressBar.Horizontal"/>   <LinearLayout  android:orientation="horizontal"  android:gravity="center"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content">  <Button  android:id="@+id/btnplay"  android:text="▶"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"/>  <Button  android:id="@+id/btnstop"  android:text="■"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"/>  </LinearLayout> </LinearLayout> |
| **MainActivity.kt** |
| package com.frost.lab01service  import android.content.BroadcastReceiver import android.content.Context import android.content.Intent import android.content.IntentFilter import androidx.appcompat.app.AppCompatActivity import android.os.Bundle import android.os.SystemClock import com.frost.lab01service.databinding.ActivityMainBinding   class MainActivity : AppCompatActivity() {  lateinit var binding: ActivityMainBinding  lateinit var songlist:Array<String>  lateinit var song:String   var runThread=false  var thread:ProgressThread?=null   override fun onCreate(savedInstanceState: Bundle?) {  super.onCreate(savedInstanceState)  binding = ActivityMainBinding.inflate(*layoutInflater*)  setContentView(binding.*root*)  initLayout()  }   private fun initLayout() {  songlist= *resources*.getStringArray(R.array.*songlist*)  song = songlist[0]   binding.*apply* **{** listview.setOnItemClickListener **{** parent, view, position, id **->** song = songlist[position]  startPlay()  **}** btnplay.setOnClickListener **{** startPlay()  **}** btnstop.setOnClickListener **{** stopPlay()  **}  }** registerReceiver(receiver, IntentFilter("com.example.MP3ACTIVITY"))   val intent = Intent(this, MyService::class.*java*)  startService(intent)  }   override fun onDestroy() {  super.onDestroy()  unregisterReceiver(receiver)  }   private fun stopPlay() {  val serviceBRIntent = Intent("com.example.MP3SERVICE")  serviceBRIntent.putExtra("mode", "stop")  sendBroadcast(serviceBRIntent)   runThread = false  binding.progressBar.*progress*=0  }   private fun startPlay() {  val serviceBRIntent = Intent("com.example.MP3SERVICE")  serviceBRIntent.putExtra("mode", "play")  serviceBRIntent.putExtra("song",song)  sendBroadcast(serviceBRIntent)   runThread=true  if(thread==null || !thread!!.*isAlive*){  binding.progressBar.*progress*=0  thread = ProgressThread()  thread!!.start()  }  }   inner class ProgressThread:Thread(){  override fun run() {  while(runThread){  binding.progressBar.incrementProgressBy(1000)  //progressBar 1초씩 증가  SystemClock.sleep(1000)  if(binding.progressBar.*progress* == binding.progressBar.*max*) {//끝난 경우  runThread = false  binding.progressBar.*progress*=0  }  }  }  }   var receiver = object: BroadcastReceiver(){  override fun onReceive(context: Context?, intent: Intent?) {  val mode = intent!!.getStringExtra("mode")  if(mode!=null){  when(mode){  "play"->{  binding.progressBar.*progress*=0  binding.progressBar.*max* = intent.getIntExtra("duration", -1)  }  "stop"->{  runThread = false  binding.progressBar.*progress*=0  }  "playing"->{  runThread=true  binding.progressBar.*max* = intent.getIntExtra("duration",-1)  binding.progressBar.*progress* = intent.getIntExtra("currentPos",-1)  song = intent.getStringExtra("song")!!  if(thread==null || !thread!!.*isAlive*){  thread = ProgressThread()  thread!!.start()  }  }  }  }  }  } } |
| **MyService.kt** |
| package com.frost.lab01service import android.app.NotificationChannel import android.app.NotificationManager import android.app.PendingIntent import android.app.Service import android.content.BroadcastReceiver import android.content.Context import android.content.Intent import android.content.IntentFilter import android.media.MediaPlayer import android.os.IBinder import androidx.core.app.NotificationCompat  class MyService : Service() {  lateinit var song:String  var player: MediaPlayer?=null   var manager:NotificationManager?=null  var notificationId=100   fun makeNotification(){  val channel = NotificationChannel("channel1", "mp3Channel",NotificationManager.IMPORTANCE\_DEFAULT)  manager = getSystemService(Context.NOTIFICATION\_SERVICE) as NotificationManager  manager?.createNotificationChannel(channel)  val builder = NotificationCompat.Builder(this, "channel1")  .setSmallIcon(R.drawable.baseline\_audiotrack\_24)  .setContentTitle("MP3")  .setContentText("MP3 플레이 중")  //넘어갈 인텐트 생성 및 정보 설정  val intent = Intent(this, MainActivity::class.java)  intent.flags = Intent.FLAG\_ACTIVITY\_NEW\_TASK or Intent.FLAG\_ACTIVITY\_CLEAR\_TOP   //팬딩 인탠트 만들어서 builder에 인텐트 세팅  val pendingIntent = PendingIntent.getActivity(this, 1,  intent, PendingIntent.FLAG\_UPDATE\_CURRENT or PendingIntent.FLAG\_IMMUTABLE)  builder.setContentIntent(pendingIntent)    val notification = builder.build()  startForeground(notificationId, notification)   }   override fun onBind(intent: Intent): IBinder? {  return null  }   var receiver = object: BroadcastReceiver(){  override fun onReceive(context: Context?, intent: Intent?) {  playControl(intent)  }  }   override fun onCreate() {  super.onCreate()  registerReceiver(receiver, IntentFilter("com.example.MP3SERVICE"))  }   override fun onStartCommand(intent: Intent?, flags: Int, startId: Int): Int {  if(player!=null&&player!!.isPlaying){  val mainBRIntent = Intent("com.example.MP3ACTIVITY")  mainBRIntent.putExtra("mode", "playing")  mainBRIntent.putExtra("song",song)  mainBRIntent.putExtra("currentPos",player!!.currentPosition)  mainBRIntent.putExtra("duration",player!!.duration)  sendBroadcast(mainBRIntent)  }  return START\_STICKY  }    override fun onDestroy() {  super.onDestroy()  unregisterReceiver(receiver)  }   private fun playControl(intent: Intent?) {  val mode = intent!!.getStringExtra("mode")  if(mode!=null){  when(mode){  "play"->{  song = intent.getStringExtra("song")!!  startPlay()  makeNotification()  }  "stop"->{  stopPlay()  stopForeground(Service.STOP\_FOREGROUND\_REMOVE)  }  }  }  }   private fun startPlay(){  val songid = resources.getIdentifier(song, "raw", packageName)  if (player != null && player!!.isPlaying) {  player!!.stop()  player!!.reset()  player!!.release()  player = null  }   player = MediaPlayer.create(this, songid)  player!!.start()    val mainBRIntent = Intent("com.example.MP3ACTIVITY")  mainBRIntent.putExtra("mode", "play")  mainBRIntent.putExtra("duration",player!!.duration)  sendBroadcast(mainBRIntent)   player!!.setOnCompletionListener **{** val mainBRIntent = Intent("com.example.MP3ACTIVITY")  mainBRIntent.putExtra("mode", "stop")  sendBroadcast(mainBRIntent)  stopPlay()  **}** }   private fun stopPlay(){  if(player!=null && player!!.isPlaying){  player!!.stop()  player!!.reset()  player!!.release()  player=null  }  } } |
| **실행결과** |
|  |