



**SharedPreferences
Kotlin**

**Shared
Preferences
Android**



❑ 1 . Khái quát *SharedPreferences* :

1. *Giả sử bạn đang chơi game, có cuộc gọi đến => ưu tiên số 1 của moblie vẫn là call*

=> ứng dụng trò chơi của bạn sẽ chuyển sang phương thức tạm dừng (onPause) hoặc dừng onStop

Xem lại bài 15 activity lifecycle

=> Sử dụng sharedPreferences để lưu trữ dữ liệu => Khi bạn quay lại trò chơi một lần nữa, bạn sẽ có thể tiếp tục nơi bạn đã dừng lại Và cũng với số điểm cuối cùng của bạn.

2. *Bạn cũng có thể sử dụng lớp này để lưu lại dữ liệu, chẳng hạn như điểm cuối cùng, điểm cao nhất....*



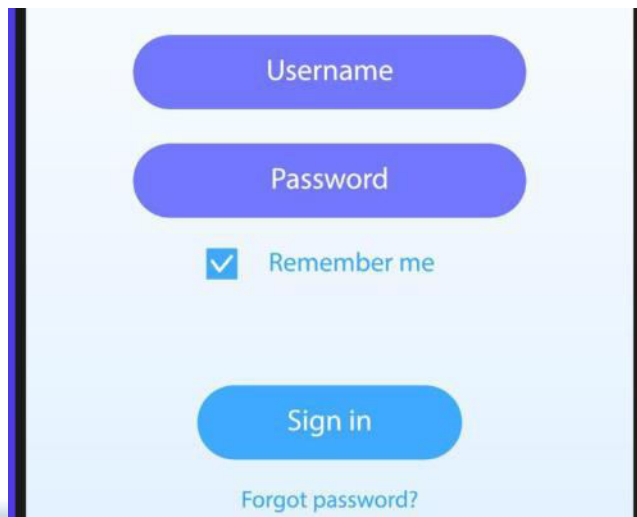
❑ 1 . Khái quát *SharedPreferences* :

3. *Bạn có thể sử dụng lớp này để lưu log đăng nhập, nhật ký người dùng.*

Ví dụ: hầu hết các ứng dụng đều có trang đăng nhập. Trên trang này, người dùng phải nhập tên người dùng và mật khẩu.

Nếu người dùng không muốn nhập thông tin này mỗi lần, lập trình viên có thể code thêm phần tích chọn Ghi nhớ và sau khi chọn hộp kiểm, ứng dụng sẽ lưu dữ liệu của người dùng.

Và khi người dùng quay lại ứng dụng, người dùng sẽ không phải mất thời gian nhập lại username và password.



Username

Password

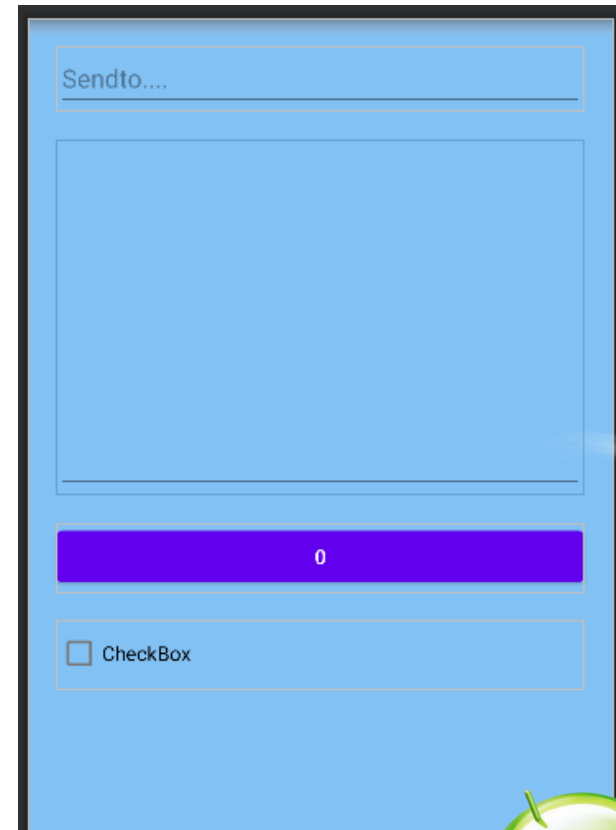
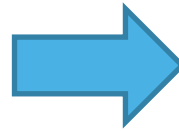
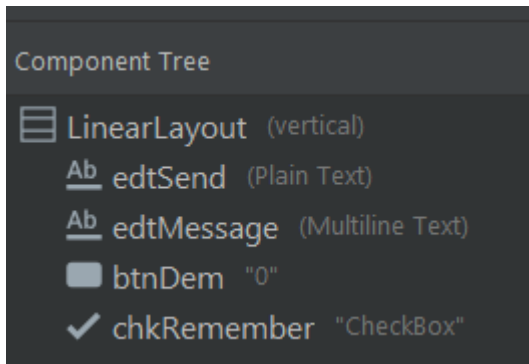
☒ Remember me

Sign in

[Forgot password?](#)

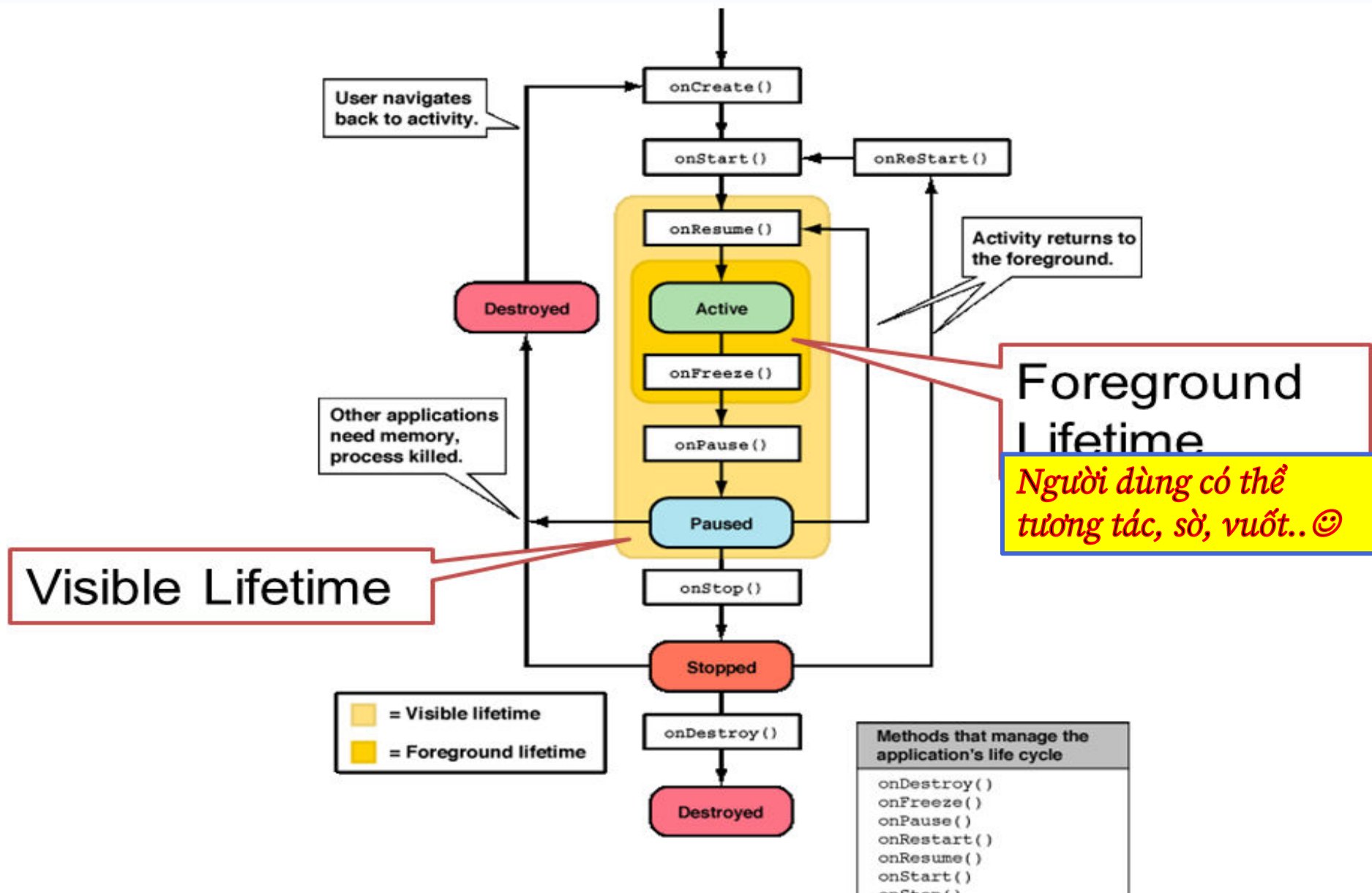


❑ 2. Thiết kế ứng dụng demo :



❑ 3 . Lưu trạng thái khi onPause:

Xem lại bài 15 activity lifecycle



❑ 3 . Lưu trạng thái khi onPause:

Xem lại bài 15 activity lifecycle

```
class MainActivity : AppCompatActivity() {
    //khai báo biến đếm:
    var dem=0
    var send:String? = null
    var message: String? = null
    var isChecked: Boolean? = null
    lateinit var sharedPreferences: SharedPreferences

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        btnDem.setOnClickListener { it: View!
            dem++
            btnDem.setText(""+dem)
            //tương đương với cách viết
            //btnDem.setText(dem.toString())
        }

    }

    override fun onPause() {
        super.onPause()
        saveData()
    }
}
```



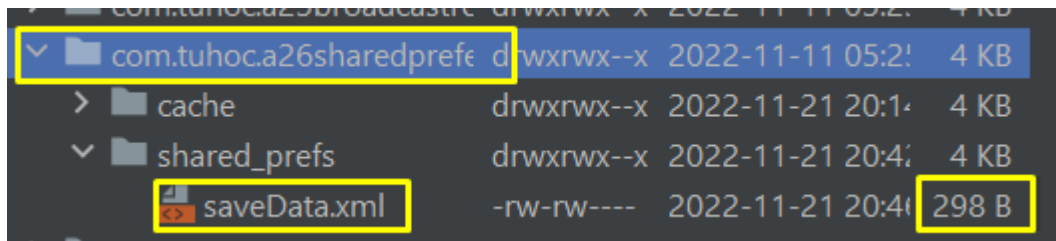
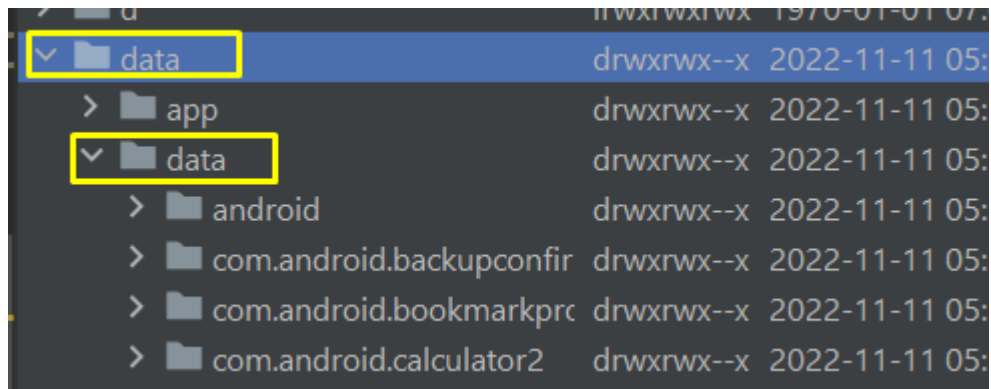
❑ 3 . Lưu trạng thái khi onPause (*code hàm saveData*) :

```
fun saveData(){
    sharedPreferences=this.getSharedPreferences( name: "saveData", Context.MODE_PRIVATE)
    //MODE_PRIVATE sẽ cấp quyền truy cập để lưu dữ liệu từ mọi nơi trong ứng dụng
    //còn bên ngoài ứng dụng sẽ không được cấp quyền
    send=edtSend.text.toString()
    message=edtMessage.text.toString()
    isChecked = chkRemember.isChecked

    //note: key phải là duy nhất
    val editor = sharedPreferences.edit()
    editor.putString("key_send", send)
    editor.putString("key_mes", message)
    editor.putInt("key_dem", dem)
    editor.putBoolean("key_remember", isChecked!!)
    editor.apply()
    Toast.makeText(applicationContext, text: "Data đã được lưu", Toast.LENGTH_SHORT).show()
}
```



❑ 4 . Kiểm tra vị trí lưu trữ của *sharedPreferences* :



❑ 5 . Gọi lại dữ liệu đã lưu trữ trong *sharedPreferences* :

MainActivity.kt

```
fun trieuHoiData(){
    sharedPreferences=this.getSharedPreferences( name: "saveData", Context.MODE_PRIVATE)
    send = sharedPreferences.getString( key: "key_send", defValue: null)
    // defValue: giá trị mặc định nếu o tìm đc key
    message=sharedPreferences.getString( key: "key_mes", defValue: null)
    dem=sharedPreferences.getInt( key: "key_dem", defValue: 0)
    isChecked = sharedPreferences.getBoolean( key: "key_remember", defValue: false)

    edtSend.setText(send)
    edtMessage.setText(message)
    btnDem.setText(""+dem)
    chkRemember.isChecked = isChecked!!
}

override fun onResume() {
    super.onResume()
    trieuHoiData()
}
```

