**1. Khi khởi động chương trình, những sự kiện nào được gọi, cho biết trạng thái hiện tại của MainActivity:**

Khi ứng dụng được khởi chạy lần đầu tiên và MainActivity được tạo ra, các sự kiện vòng đời sau sẽ được gọi theo thứ tự:

* Sự kiện 1: onCreate(): Activity đang được tạo. Đây là nơi bạn khởi tạo ban đầu.
* Sự kiện 2: onStart()
* Sự kiện 3: onResume()

Thứ tự Toasts hiển thị:

1. CR424 - onCreate()
2. CR424 - onStart
3. CR424 - onResume

Trạng thái hiện tại của MainActivity: Đang hoạt động và có thể tương tác (Running/Resumed).

**2. Khi bấm vào nút Call Dialog (nút btncall), những sự kiện nào được gọi đối với MainActivity, cho biết trạng thái của MainActivity, subactivity**

Khi bấm vào btncall (nút này gọi Subactivity), các sự kiện vòng đời sẽ diễn ra như sau:

* MainActivity:
  + onPause(): MainActivity sắp mất focus vì Subactivity đang được khởi chạy và sẽ hiển thị trên nó. MainActivity vẫn hiển thị một phần nhưng không thể tương tác.
  + onStop(): Sau khi Subactivity hoàn toàn hiển thị và che khuất MainActivity, MainActivity sẽ bị dừng hoàn toàn. Nó không còn hiển thị trên màn hình nữa.
* Subactivity:
  + onCreate(): Subactivity được tạo.
  + onStart(): Subactivity sắp hiển thị.
  + onResume(): Subactivity đã sẵn sàng tương tác.

Thứ tự Toasts hiển thị (khi bấm nút):

1. CR424 - onPause() (của MainActivity)
2. CR424 - onCreate() (của Subactivity)
3. CR424 - onStart (của Subactivity)
4. CR424 - onResume (của Subactivity)
5. CR424 - onStop() (của MainActivity)

Trạng thái hiện tại:

* MainActivity: Dừng (Stopped).
* Subactivity: Đang hoạt động và có thể tương tác (Running/Resumed).

**3. Bấm vào nút OK, sự kiện nào được gọi đối với MainActivity cho biết trạng thái của MainActivity, subactivity**

* Subactivity:
  + onPause(): Subactivity sắp mất focus.
  + onStop(): Subactivity không còn hiển thị.
  + onDestroy(): Subactivity bị hủy hoàn toàn.
* MainActivity:
  + onRestart(): MainActivity đang bị dừng (stopped), và bây giờ nó cần hiển thị lại.
  + onStart(): MainActivity sắp hiển thị.
  + onResume(): MainActivity đã sẵn sàng tương tác.

Thứ tự Toasts hiển thị (khi bấm nút OK trong Subactivity để đóng nó):

1. CR424 - onPause() (của Subactivity)
2. CR424 - onStop() (của Subactivity)
3. CR424 - onDestroy() (của Subactivity)
4. CR424 - onRestart() (của MainActivity)
5. CR424 - onStart (của MainActivity)
6. CR424 - onResume (của MainActivity)

Trạng thái hiện tại:

* MainActivity: Đang hoạt động và có thể tương tác
* Subactivity: Đã bị hủy

**4. Tiếp tục Bấm nút Back trên thiết bị, những sự kiện nào được gọi với MainActivity, cho biết trạng thái hiện tại của MainActivity:**

* MainActivity:
  + onPause(): MainActivity mất focus.
  + onStop(): MainActivity không còn hiển thị.
  + onDestroy(): MainActivity bị hủy hoàn toàn và ứng dụng kết thúc.
* Thứ tự Toasts hiển thị:

1. CR424 - onPause() (của MainActivity)
2. CR424 - onStop() (của MainActivity)
3. CR424 - onDestroy() (của MainActivity)

Trạng thái hiện tại của MainActivity: Đã bị hủy

**5. Tiếp tục khởi động lại chương trình, sau đó bấm nút Home, những sự kiện nào được gọi, cho biết trạng thái hiện tại của MainActivity:**

Các sự kiện sau sẽ được gọi:

* onCreate()
* onStart()
* onResume()

Trạng thái hiện tại của MainActivity: Đang hoạt động và có thể tương tác

Sau đó bấm nút Home:

* MainActivity:
  + onPause(): MainActivity mất focus.
  + onStop(): MainActivity không còn hiển thị trên màn hình (đã chuyển xuống background).

Thứ tự Toasts hiển thị

1. CR424 - onPause() (của MainActivity)
2. CR424 - onStop() (của MainActivity)

Trạng thái hiện tại của MainActivity: Dừng (Stopped). MainActivity vẫn tồn tại trong bộ nhớ, nhưng không hiển thị cho người dùng. Nó có thể được khôi phục trạng thái bằng onRestart() nếu người dùng quay lại ứng dụng.

-Đoạn code khai báo và gắn ID cho các Textview, Button

* MainActivity:

android:id="@+id/textView2"

android:id="@+id/btncall"

android:id="@+id/textView"

* Subactivity:

android:id="@+id/textView"

android:id="@+id/textView2"

android:id="@+id/btnok"

-Giải thích lệnh

* **setContentView(R.layout.subactivity);** :Thiết lập giao diện người dùng cho Activity hiện tại bằng cách sử dụng file layout subactivity.xml (nằm trong thư mục res/layout). Khi Activity này được hiển thị, nó sẽ tải và hiển thị tất cả các View (TextView, Button, EditText) đ0ược định nghĩa trong subactivity.xml
* **btnok = (Button) findViewById(R.id.btn1); :** tìm kiếm một Button trong layout có ID là btn1 và gán tham chiếu đến Button đó vào biến btnok, cho phép bạn tương tác với Button đó trong code
* **Toast.makeText(this, "K16EVT - onStop", Toast.LENGTH\_SHORT).show();:** tạo và hiển thị một thông báo Toast ngắn gọn trên màn hình với nội dung "K16EVT - onStop"
* **Intent intent1 = new Intent(MainActivity.this, subactivity.class); startActivity(intent1);:** sử dụng để chuyển từ một Activity sang một Activity khác (MainActivity sang subactivity)
* **finish();:** kết thúc và đóng Activity hiện tại
* **protected void onStop() { // TODO Auto-generated method stub Toast.makeText(this, "K16EVT - onStop", Toast.LENGTH\_SHORT).show(); super.onStop(); }:** đây là 1 phương thức ghi đè từ lớp cha AppCompatActivity, nó sẽ gọi phương thức onStop của lớp cha khi giao diện không còn hiển thị với người dùng sau đó hiển thị ra 1 thông báo: **"K16EVT - onStop"**