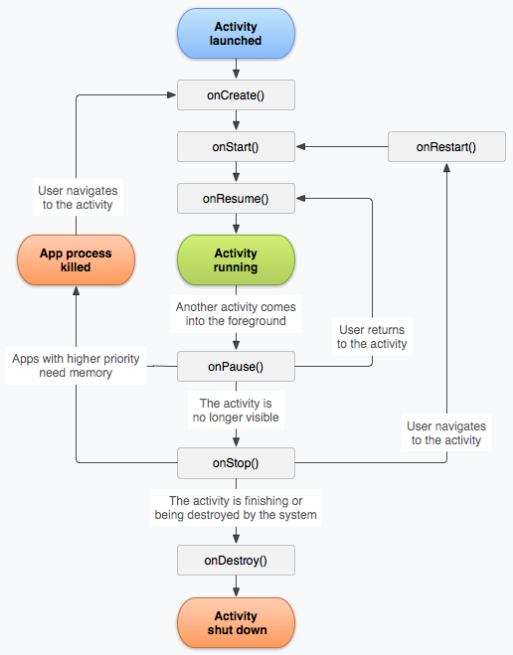
# Activity

* Activity trong android nghĩa là một màn hình duy nhất mà người dùng tương tác.Mỗi android có nhiều activity ,mỗi activity đại diện cho một màn hình khác nhau
* Mỗi activity được biểu diễn bởi một lớp java mà mở rộng từ các lớp “Activity” và được xác định trong tệp mainfest của ứng dụng.Khi người dùng chuyển giữa các hoạt động khác nhau,hệ thống tạo
* Mỗi hoạt động được mô tả bằng một lớp Java định nghĩa các phương thức sự kiện, như “onCreate”,” onStart”, và “onPause”, để xử lý các tác vụ liên quan đến hoạt động như khởi tạo, bắt đầu hoạt động và tạm dừng hoạt động.

1. **Vòng đời một activity**

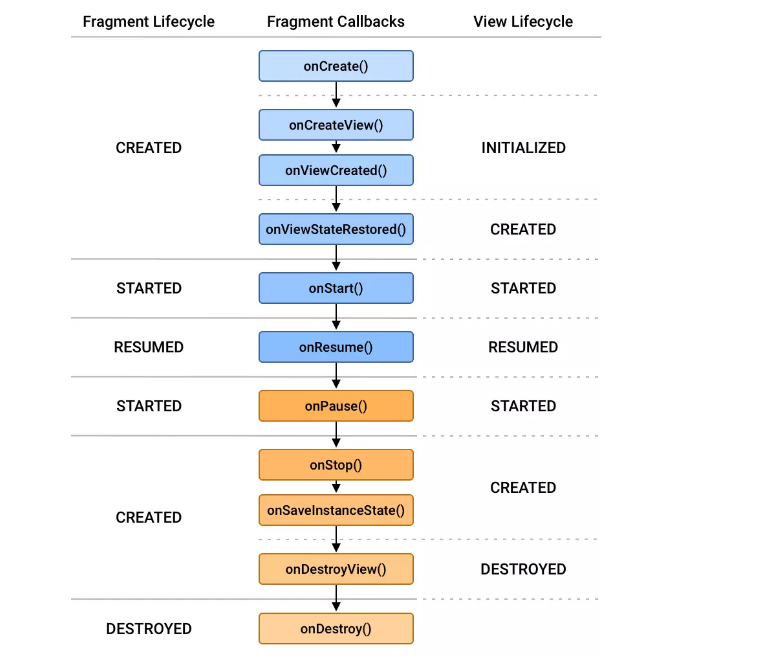
* Trong lập trình ứng dụng di động, vòng đời của một của một activity mô tả chuỗi các trạng thái mà một activity có thể đi qua từ khi nó được tạo cho đến khi nó bị hủy
* Để tạo một activity thì bạn phải tạo ra một lớp kế thừa Activity sau đó triển khai tối thiểu phương thức onCreate(Bundlesave InstanceStae), sau đó tùy ngữ cảnh mà Activity hoạt động vòng đời của nó:
* ****
* OnCreate(Bundle saveInstanceState): được gọi khi hoạt động mới được tạo , tại đây khởi tạo các biến , nạp giao diện layout …phương thức này cũng nhận dữ liệu lưu lại trạng thái hoạt động trước đó
* OnStart(): Được gọi ngay trước khi Activity hiện thị lên màn hình
* OnResume(): Được gọi ngay khi Activity có thể tương tác với người dùng,và Activity nằm trên cùng trong danh sách các Activity hệ thôngs. Sau phương thức này các chức năng của Activity sẽ hoạt động dựa trên tương tác của người dùng …cho đến khi phương thức OnPause được gọi
* OnPause(): Được gọi khi hệ thống sắp kích hoạt một Activity khác
* OnStop(): Được gọi khi nó bị ẩn đi.Sau phương thức này ,Activity có thể gọi onRestart() nếu nó được người dùng kích hoạt hoặc gọi lại OnDestroy đẻ kết thúc
* OnDestroy():gọi khi Activity bị hủy hoàn toàn

1. Các trạng thái chính của một vòng đời

* Running:Khi activity được kích hoạt và hệ thống để vào Blackstack, nó sẽ bước vào trạng thái active
* Pause: Là trạng thái tạm dừng trạng thái này xảy ra khi mà Activity của bạn vẫn đang chạy, người dùng vẫn nhìn thấy, nhưng Activity khi này lại bị che một phần bởi một thành phần nào đó. Cái sự che Activity này không phải hoàn toàn. Chính vì vậy mà Activity đó tuy được người dùng nhìn thấy nhưng không tương tác được.
* Stop: Trạng thái này khá giống trạng thái tạm dừng. Nhưng khi này Activity bị che khuất hoàn toàn bởi 1 thành phần nào đó hoặc bởi 1 ứng dụng khác và người dùng không thể thấy Activity của bạn được .
* Dead: Nếu activity được lấy ra khỏi BlackStack , chúng sẽ bị hủy và rơi vào trạng thái này .

# Fragment

* Trong lập trình ứng dụng di động, vòng đời của một fragment mô tả chuỗi các trạng thái mà một fragment có thể đi qua từ khi nó được tạo ra cho đến khi nó bị hủy.



* OnAttach(): Được gọi khi fragment được liên kết với activity của nó. Trong phương thức này, fragment có thể truy cập vào activity chưa nó bằng cách sử dụng đối tượng Context
* OnCreate(): Được gọi khi fragment được tạo ra . Trong phương thức này các thành phần cơ bản của một fragment như giao diện người dùng và các thành phần khác khởi tạo
* OnCreateView(): Được gọi khi fragment cần tạo giao diện người dùng của nó. Trong phương thức này,fragment phải tạo một đối tượng View để hiển thị lên màn hình
* onActivityCreated():được gọi khi activity chứa fragment đã hoàn thành onCreate() của nó .

Trong phương thức này ,fragment có thể truy cập vào activity chứa nó và các thành phần của nó.

* onStart(): Được gọi khi fragment bắt đầu hiển thị trên màn hình. Trong phuownh thức này , các tài nguyên khác như máy ảnh hocacwj kết nối internet được yêu cầu
* OnPause():Được gọi khi fragment tạm dừng, ví dụ như khi một cuộc gọi đến hoặc khi fragment khác được mở lên. Trong phương thức này các tài nguyên được giải phóng để giải phóng bộ nhớ và hạn chế sử dụng pin
* OnResume():được gọi khi fragment trở thành hoạt động chính trên màn hình. Trong phương thức này các đối tượng liên quan đến giao diện người dùng như các thread hoặc annimation được khởi chạy
* OnStop():được gọi khi các fragment không còn hiển thị trên màn hình. Trong phương thức này,các tài nguyên cần thiết sẽ được giải phòng
* OnDestroyView():được gọi khi fragment được hủy giao diện người dùng của nó. Trong phương thức này ,các thành phần cơ bản của fragment như giao diện người dùng và các thành phần khác được giải phóng
* onDestroy():được gọi khi fragment bị hủy. trong phương thức này các đối tượng không cần thiết được giải phóng để giải phóng bộ nhớ
* OnDetach():được gọi khi fragment không còn được liên kết với activity giá rẻ