

**Student’s Name ID**

* *Đinh Nguyễn Phương Vy* 4501104282
* *Lê Thị An Bình* 4501104018
* *Nguyễn Thị Niệm* 4501104
* *Nguyễn Võ Hiếu Trung* 4501104263

ỨNG DỤNG DỰ BÁO THỜI TIẾT



**The Weather forecast team**

----🌣🌤🌥🌦☁🌨🌧🌩🌪----

app WEATHER

# **Acknowledgement**

Q

ua thời gian học tập và nghiên cứu qua học phần phát triển di động, nhóm chúng em xin chân thành cảm ơn sâu sắc tới cô Phan Nguyệt Minh, giảng viên phụ trách bộ môn Phát triển ứng dụng di động. Cô đã dạy và truyền đạt những kiến thức bổ ích về hệ điều hành Android, hiễu rõ hơn và nắm chắc kiến thức để có thể viết được một ứng dụng chạy trên di động. Với những kinh nghiệm mà cô đã dạy cho nhóm chúng em, tuy nhiên sẽ rất có thể ứng dụng của em sẽ không tránh khỏi những thiếu sót, mong cô sẽ cho nhóm em nhận xét và hướng dẫn để nhóm chúng em có thể rút kinh nghiệm. Nhóm em xin chân thành cảm ơn!

# **Table of contents**

[Acknowledgement 2](#_Toc73125469)

[Table of contents 3](#_Toc73125470)

[Introduction about other weather apps 4](#_Toc73125471)

[**Part 1.** **KHÁI QUÁT VỀ HỆ ĐIỀU HÀNH ANDROID** 5](#_Toc73125472)

[**Part 2.** **GIỚI THIÊU KHÁI QUÁT (Introduction)** 5](#_Toc73125473)

[**Part 3.** **CÁC CHỨC NĂNG CỦA WEATHER (the Fuctions of Weather)** 6](#_Toc73125474)

[**Part 4.** **HƯỚNG DẪN CÁCH SỬ DỤNG WEATHER (Instruction for using Weather)** 7](#_Toc73125475)

[Cấu trúc của Weather 7](#_Toc73125476)

[Màn hình 1: Khởi tạo màn hình 1 9](#_Toc73125477)

[Chuyển màn hình 13](#_Toc73125478)

[Màn hình 2: Khởi tạo màn hình 2 13](#_Toc73125479)

# **Introduction about other weather apps**

Khi cuộc cách mạng về công nghệ 4.0 đã và đang diễn ra, cuộc sống xã hội của chúng ta cũng phải dần thích nghi và làm quen nhưng đổi mới tân tiến về các công nghệ hiện đại hơn, tiện lợi hơn.

Nếu như ngày xưa những người thuộc thế hệ trước dựa vào những ánh sáng mặt trời, mặt trăng hay dựa vào những hành vi của động vật, côn trùng để nhận biết thời tiết, khí hậu thì ngày nay chính chúng ta phải tự tiên đoán trước mọi thứ.

Giữa thế giới mà khí hậu ngày càng khắc nghiệt và thay đổi, những “phương pháp” dự báo thời tiết ngày xưa đã không còn hoàn toàn chính xác. Chúng ta ngày nay dựa vào các công nghệ tân tiến, điển hình nhất là các vệ tinh thăm dò hay thậm chí các trí tuệ nhân tạo đã giúp chúng ta rất nhiều trong việc cho biết thời tiết hiện tại, dự báo thời tiết trong tương lai gần hay thậm chí là nhiều tuần, nhiều tháng sau đó.

Ngày trước chúng ta thường dựa vào nhà khí tượng học địa phương để tìm hiểu thời tiết sẽ như thế nào, nhưng xem xét có bao nhiêu người trong xã hội sử dụng điện thoại thông minh ngày nay, ứng dụng di động đã trở thành một cách tuyệt vời để có được dự báo mới nhất bất cứ khi nào bạn cần. Cho dù bạn đang đi trên đường hay chỉ không có quyền truy cập vào TV, bạn vẫn có thể biết liệu mình có nên đóng gói một chiếc ô hay không.

Hiện nay, có hàng trăm ứng dụng dự báo thời tiết trong cửa hàng App Store và CH Play giúp người dùng theo dõi thông tin mưa gió, bão lũ trong ngày hoặc trong tuần. Tuy nhiên, có nhiều ứng dụng không cập nhật nhanh và chính xác khiến bạn không bỏ qua những tin tức quan trọng. Dark Sky, Windy, Weatherbug, Solar, Living Earth, Weathertron và cả hằng trăm các ứng dụng dự báo trên di động hàng loạt được ra mắt.

Tuy vậy các ứng dụng này đến thời điểm hiện tại đã và đang làm rất tốt công việc dự báo và cung cấp hàng loạt các thông tin hữu ích, chính xác không chỉ đến người dân trên khắp miền đất nước mà còn hỗ trợ cho công tác tránh bão và đối phó với bão của cơ quan chức năng.

1. KHÁI QUÁT VỀ HỆ ĐIỀU HÀNH ANDROID

**Android** được biến đến là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Ban đầu, **Android** được phát triển bởi Android, Inc. với sự hỗ trợ tài chính từ **Google** và sau này được chính **Google** mua lại vào năm 2005.

ndroid ra mắt vào năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cầm tay mở: một hiệp hội gồm các công ty phần cứng, phần mềm, và viễn thông với mục tiêu đẩy mạnh các tiêu chuẩn mở cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy **Android** được bán vào năm 2008.

**Android** có mã nguồn mở và **Google** phát hành mã nguồn theo Giấy phép **Apache**, cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối **Android** một cách tự do. Ngoài ra, **Android** còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình Java có sửa đổi. Tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên **Android**, và số lượt tải ứng dụng từ Google Play, cửa hàng ứng dụng chính của **Android**, ước tính khoảng 25 tỷ lượt.

Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới vượt qua **Symbian OS** vào quý 4 năm 2010, và được các công ty công nghệ lựa chọn khi họ cần một hệ điều hành không nặng nề, có khả năng tinh chỉnh, và giá rẻ chạy trên các thiết bị công nghệ cao thay vì tạo dựng từ đầu. Kết quả là mặc dù được thiết kế để chạy trên điện thoại và máy tính bảng, **Android** đã xuất hiện trên TV, máy chơi game và các thiết bị điện tử khác. Bản chất mở của Android cũng khích lệ một đội ngũ đông đảo lập trình viên và những người đam mê sử dụng mã nguồn mở để tạo ra những dự án do cộng đồng quản lý.

1. GIỚI THIÊU KHÁI QUÁT (Introduction)

**Weather** là một ứng dụng di động dùng để xác định và thời tiết dựa trên các thiết bị vệ tinh.

****Nhóm chúng em đã lên ý tưởng về một ứng dụng có thể dự báo hiện tường thời tiết tại khu vực người dùng trong một thời gian gần đồng thời có thể đưa ra thông số chích xác về khí hậu xung quanh hiện tại với dữ liệu cập nhật liên tục.  
Vì vậy **Weather** sẽ luôn luôn có dữ liệu theo thời gian thực cập nhật theo từng phút trên giao diện bản đồ vệ tinh một cách mượt mà. Nhóm dự định sẽ cho ứng dụng **Weather** có mặt trên cả Android và **iOS**, hiển thị thông tin theo thời gian thực trên giao diện chính và các hiệu ứng chuyển động trên hình nền. Ứng dụng này đưa thông tin thời tiết hiển thị lên biểu tượng màn hình Home.

**Một số app dự báo thời tiết đang thịnh hành**

Bạn có thể tùy biến giao diện và thông số theo ý mình.

**Một ví dụ về app thời tiết**

1. CÁC CHỨC NĂNG CỦA WEATHER (the Fuctions of Weather)

Là app thời tiết Weather tất nhiên sẽ được trang bị những chức năng vô cùng tiện lợi cho việc xem xét điều kiện thời tiết như:

* Xem xét nhiệt độ (Temperature consideration)
* Độ ẩm (Humidity)
* Lưu lượng mây (Cloud-flow)
* Lưu lượng gió (Wind-flow)
* Tia Uv (Ultraviolet)
* Mặt trời lặn/mọc (Sunset**/**sunrise)

Trong Weather thì *xem xét nhiệt độ* à thước đo lường động năng trung bình của các phân tử trong không khí, được biểu thị bằng đơn vị hoặc độ được chỉ định trên thang đo chuẩn **(oC)**.

*Độ ẩm* là lượng hơi nước có trong không khí, hơi nước chính là dạng khí của nước và vô hình với mắt người. Độ ẩm là thước đo cho thấy khả năng về lượng mưa **(%)** hoặc sương mù **(%).**

*Lưu lượng mây, gió* có thể xem là một đại lượng, một thông số dùng để chỉ mức độ lưu thông của mây, gió. *Lưu lượng mây* sẽ biểu thị bằng **(%),** tuy nhiên *lưu lượng gió* sẽ được đo bằng tốc độ lưu thông của mây **(m/s)**.

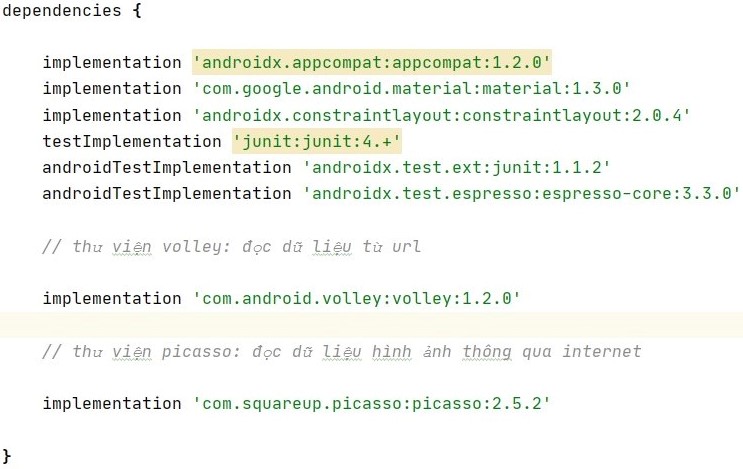
*Tia UV* hay còn gọi là tia tử ngoại, tia cực tím là sóng điện từ Phổ của tia cực tím có thể chia ra thành 2 vùng tia: vùng tử ngoại gần và vùng tử ngoại xạ (vùng tử ngoại chân không).

*Mặt trời lặn/mọc* cho biết khảng thời gian được dự đoán khi nào mặt trời lặn(mọc) trong ngày với độ chính xác cao.

**Mẫu các chức năng trong Weather**

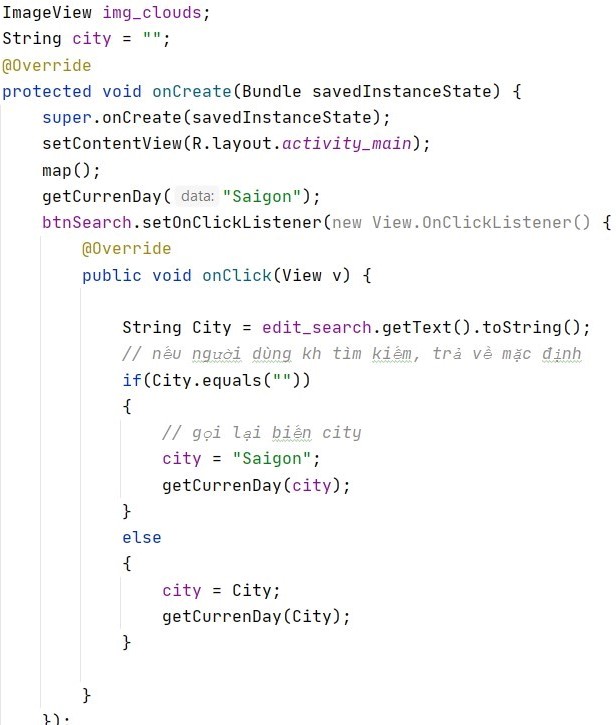
1. HƯỚNG DẪN CÁCH SỬ DỤNG WEATHER (Instruction for using Weather)

## Cấu trúc của Weather

Các thư viện được khai báo: 2 thư viện trong build.gard(app)

Hàm khởi tạo, khởi tạo import các thư viện có sẵn:

Phần đầu màn hình bao gồm đặt ngày tháng năm và tên thành phố:



Ngày tháng năm

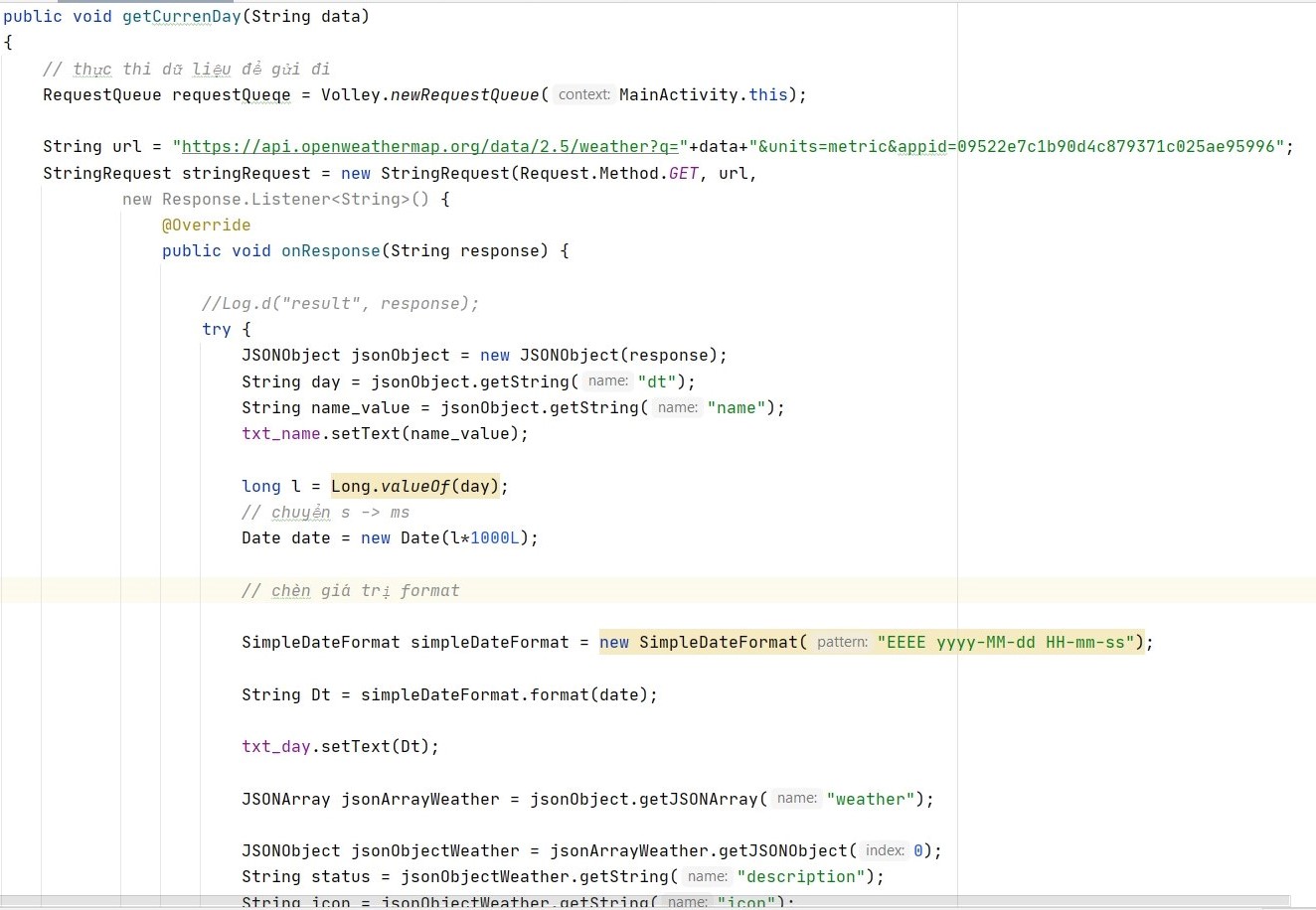
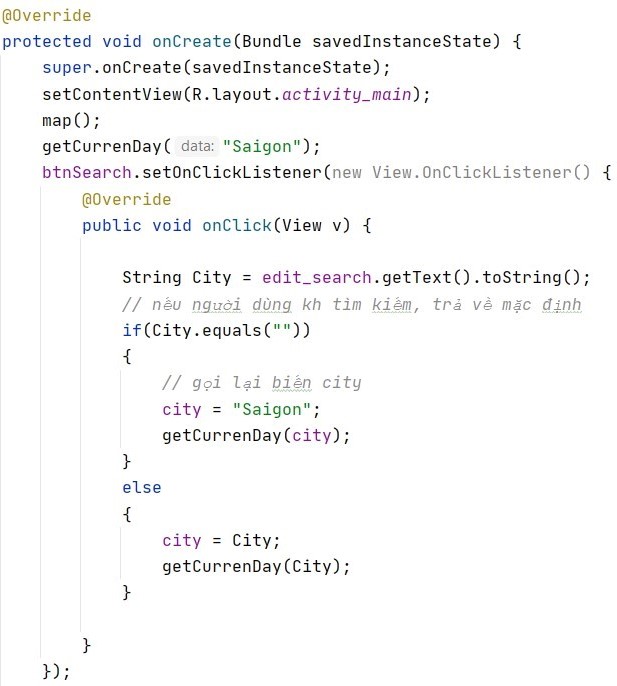
Tên thành phố

## Màn hình 1: Khởi tạo màn hình 1

Lập hàm gán giá trị cho từng biến

Khai báo “màn hình chính”

\_Khi lập trình ***Android***, để tương tác với các *View/Control* trong giao diện chúng ta thường thông qua thuộc tính Id của các *view/control* để truy suất thay đổi dữ liệu. Vì ***Android***chia màn hình (*Activity*) thành hai phần: Phần thiết giao diện, phần xử lý nghiệp vụ. Do đó để truy suất được tới các View trong phần giao diện ***Android***cung cấp hàm *findViewById*

\_Phần đầu của màn hình 1 bao gồm text và imgbutton:

**1**

Màn hình chính của Weather sẽ được mặc định là thời tiết ở Sài Gòn

\_Mặc định thời tiết sẽ thành phố Sài Gòn nhưng nếu người dùng mở ứng dụng sẽ hiển thị thời tiết sài Gòn còn kh sẽ hiện thị thời tiết khác bằng 1 hàm sau đây



**3**

**2**



**4**



**5**

## Chuyển màn hình

Khi người dùng tương tác bên màn hình 1 và hệ thống chuyển sang màn hình 2 thì đoạnn code này sẽ đảm nhận chuyện đó:

***Intent***: trong ***Android*** là object của ***class android.content.Intent***. ***Code*** của bạn có thể gửi ***Intent*** đến hệ thống ***Android*** với chỉ định thành phần mục tiêu gửi đến. Ví dụ, thông qua ***startActivity()*** bạn có thể xác định một ***Intent*** sử dụng để gọi chạy một *Activity* khác. Tại *Activity* mục tiêu, với ***startActivity()*** chúng ta có thể xác định được ***Intent*** của người gửi đến để khởi động Activity này.

***putExtra***: Sau khi khai báo ***Intent*** và trước khi bạn dùng nó để kích hoạt *activity* nào đó, bạn có thể sử dụng các phương thức được nạp chồng của nó để gửi dữ liệu Để truyền dữ liệu cho ***intent***, ta dùng phương thức ***putExtras()***. ***Extras*** là một cặp *key/value*. *key* luôn luôn là kiểu *string. value* bạn có thể sử dụng kiểu dữ liệu nguyên thủy hoặc đối tượng của *String*, *Bundle*, ....

PutExtra(String, String[])

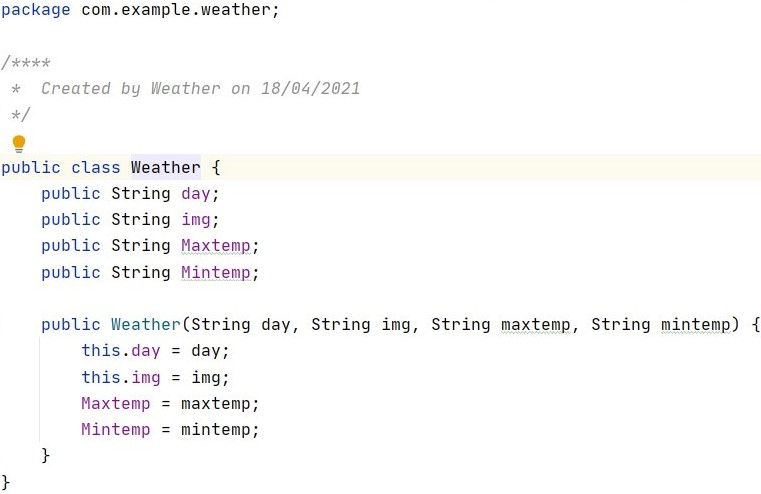
Add extended data to the intent.

## Màn hình 2: Khởi tạo màn hình 2

--**Khởi tạo thêm hai phần Java:**

\_ *Weather.java*

\_*CustomAdapter.java*



Từ đó ta tạo ra một lớp nhằm phục vụ cho người sử dụng: **class** ***CustomAdapter***

**2**

**1**

**Khởi tạo màn hình 2:**

Gán ID cho các biến: ***img\_back***, ***txt\_name***, ***ListView***

Adapter(trong ***customAdapter***) là gì?

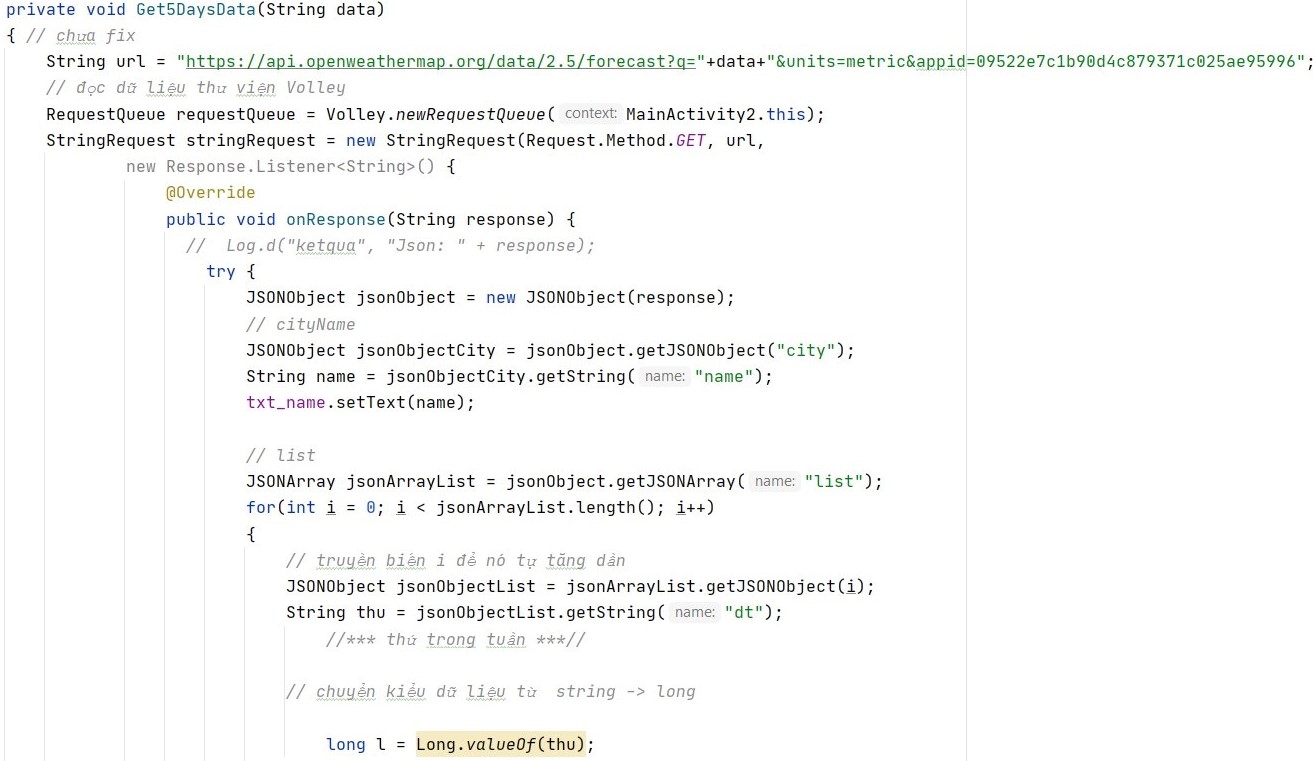
Một Adapter là một đối tượng của một lớp cài đặt giao diện Adapter. Nó đóng vai trò như là một liên kết giữa một tập hợp dữ liệu và một Adapter View, một đối tượng của một lớp thừa kế lớp trừu tượng AdapterView. Tập hợp dữ liệu có thể là bất cứ điều gì mà trình bày dữ liệu một cách có cấu trúc. Mảng, các đối tượng List và các đối tượng Cursor thường sử dụng bộ dữ liệu.

Một Adapter có trách nhiệm lấy dữ liệu từ bộ dữ liệu và tạo ra các đối tượng View dựa trên dữ liệu đó. Các đối tượng View được tạo ra sau đó được sử dụng để gắn lên bất kỳ Adapter View mà ràng buộc với Adapter

Bạn có thể tạo các lớp Adapter riêng của bạn từ đầu, nhưng hầu hết các nhà phát triển muốn sử dụng hoặc thừa kế các lớp Adapter được cung cấp bởi Android SDK, chẳng hạn như ArrayAdapter và SimpleCursorAdapter. Trong hướng dẫn này, chúng ta sẽ tập trung vào lớp ArrayAdapter.

Để liên kết ListView với Adapter mà chúng ta đã tạo ra ở bước trước đó, hãy gọi phương thức trong chỗ void map setAdapter().

List View

Nếu người dùng mở ứng dụng thì mặc định hiển thị cái Sài Gòn còn người dùng chọn thành phố thì hiện thành phố mà người dùng chọn bằng hàm như sao:

**1**



**2**

Nếu người dùng mún quay lại màn hình 1 thì sẽ là phần img\_back

***onBackPressed()***: Bình thường nếu ta không làm gì, ấn nút back sẽ khiến màn hình hiện tại bị thoát. Nếu ta muốn vô hiệu hóa nút back, ta dùng dòng sau.

@Override

**public** **void** onBackPressed(){

}   
Chú ý là bên trong không có dòng *super.onBackPressed()*;   
Cách này sẽ khiến nút back bị trơ, không có phản ứng gì, nếu người dùng đang ở trong màn hình application, bình thường ấn nút back sẽ quay về màn hình home, nếu bị vô hiệu hóa, sẽ phải ấn nút home mới quay về được.

Để trở về nhưng không mất dữ liệu thì trong main ta đặt như sau

**Đường link tham khảo:**

<https://developer.android.com/training/volley#groovy>  
<https://github.com/square/picasso>

<https://vi.wikipedia.org/wiki/Android_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)>

<https://atpsoftware.vn/top-15-ung-dung-du-bao-thoi-tiet-cho-thiet-bi-android-va-ios-tot-nhat-2019.html>