**XÂY DỰNG ỨNG DỤNG PHÂN TÍCH HỆ THỐNG HVAC**

**Giới thiệu**

Hortonworks Connected Data Platform có thể được sử dụng để thu thập, làm sạch và hiển thị dữ liệu từ hệ thống máy thông gió, điều hòa không khí (HVAC) để duy trì nhiệt độ tối ưu trong tòa nhà văn phòng và giảm thiểu chi phí.

**Các Công nghệ Big Data được sử dụng để phát triển ứng dụng:**

* Dữ liệu cảm biến HVAC lịch sử
* [HDF Sandbox](https://www.cloudera.com/products.html)
  + [Apache Ambari](https://ambari.apache.org/)
  + [Apache NiFi](https://nifi.apache.org/)
* [HDP Sandbox](https://www.cloudera.com/products.html)
  + [Apache Ambari](https://ambari.apache.org/)
  + [Apache Hadoop - HDFS](https://hadoop.apache.org/docs/r2.7.6/)
  + [Apache Hive](https://hive.apache.org/)
  + [Apache Zeppelin](https://zeppelin.apache.org/)

**Mục tiêu và Đối tượng**

* Học cách viết shell script để tự động thiết lập môi trường phát triển
* Học cách xây dựng một luồng NiFi để thu thập dữ liệu cảm biến máy HVAC
* Học cách viết các script Hive để làm sạch dữ liệu cảm biến máy HVAC và chuẩn bị cho việc hiển thị
* Học cách hiển thị dữ liệu cảm biến máy HVAC trong Zeppelin

**Yêu cầu tiên quyết**

* Đã tải xuống và cài đặt phiên bản mới nhất của [Hortonworks HDP Sandbox]

<https://www.cloudera.com/downloads/hortonworks-sandbox/hdp.html?utm_source=mktg-tutorial> Sandbox

* Đọc qua [Hướng dẫn sử dụng Hortonworks Sandbox](https://www.cloudera.com/tutorials.html) để thiết lập ánh xạ tên máy chủ thành địa chỉ IP
* Nếu bạn không có ít nhất 16GB RAM cho HDP Sandbox và 4GB RAM cho máy tính của bạn, hãy tham khảo [Triển khai Hortonworks Sandbox trên Microsoft Azure](https://www.cloudera.com/tutorials.html)
* Kích hoạt Connected Data Architecture:
  + [Kích hoạt CDA cho VirtualBox](https://www.cloudera.com/tutorials.html#enable-connected-data-architecture-cda---advanced-topic)
  + [Kích hoạt CDA cho VMware](https://www.cloudera.com/tutorials.html#enable-connected-data-architecture-cda---advanced-topic)
  + [Kích hoạt CDA cho Docker](https://www.cloudera.com/tutorials.html#enable-connected-data-architecture-cda---advanced-topic)

**Tổng quan**

*Loạt bài hướng dẫn bao gồm các module hướng dẫn sau:*

1. [Khái niệm phát triển ứng dụng](https://www.cloudera.com/tutorials.html): Tập trung vào những kiến thức cơ bản về HVAC, các cảm biến phổ biến được sử dụng trong hệ thống HVAC và cách phân tích dữ liệu từ các cảm biến này để hiểu được tình trạng của hệ thống HVAC.

2. [Thiết lập môi trường phát triển](https://www.cloudera.com/tutorials.html): Bất kỳ cấu hình và/hoặc dịch vụ phần mềm nào có thể cần được cài đặt trước khi xây dựng đường ống dữ liệu và sổ tay trực quan hóa.

3. [Thu thập dữ liệu cảm biến HVAC](https://www.cloudera.com/tutorials.html): Tạo một phần của đường ống dữ liệu sử dụng Apache NiFi để nhập, xử lý và lưu trữ 1 tháng dữ liệu lịch sử cảm biến HVAC vào HDFS.

4. [Làm sạch dữ liệu HVAC thô](https://www.cloudera.com/tutorials.html): Tạo phần tiếp theo của đường ống dữ liệu sử dụng Apache Hive để tải dữ liệu vào các bảng Hive, để dữ liệu có thể được làm sạch và truy vấn để thu được thông tin quan trọng về tình trạng của hệ thống HVAC điều chỉnh nhiệt độ trong các tòa nhà trên các quốc gia khác nhau.

5. [Trực quan hóa dữ liệu cảm biến liên quan đến hệ thống máy HVAC](https://www.cloudera.com/tutorials.html): Thực hiện phân tích dữ liệu trên cảm biến HVAC để tìm Đặc điểm Nhiệt độ Toàn bộ Tòa nhà HVAC Theo Quốc gia - đếm số lượng khoảng nhiệt độ NÓNG, LẠNH, BÌNH THƯỜNG theo từng quốc gia, Nhiệt độ Cực đoan trong Các tòa nhà Có Sản phẩm HVAC - các tòa nhà đang trải qua nhiệt độ cực đoan, dù là NÓNG hay LẠNH, mặc dù họ đã có sản phẩm HVAC.

**CÁC KHÁI NIỆM VỀ PHÁT TRIỂN ỨNG DỤNG**

**Giới thiệu**

Mục tiêu của bài học này là giới thiệu cho bạn một số ý tưởng cơ bản về Hệ thống HVAC, chẳng hạn như khả năng trả lời câu hỏi về Hệ thống HVAC là gì, tại sao nó quan trọng và những mục tiêu mà hệ thống đang cố gắng đạt được. Mục tiêu tiếp theo là tìm hiểu về các cảm biến thông thường được sử dụng trong Hệ thống HVAC và cách bạn có thể phân tích và trực quan hóa thông tin về sức khỏe của Hệ thống HVAC. Phụ lục liệt kê nhiều thành phần thông thường tạo nên Hệ thống HVAC.

**Tiên quyết**

*Đọc tổng quan về Xây dựng một ứng dụng phân tích Hệ thống HVAC.*

**Mục lục**

* Các nguyên tắc phân tích Hệ thống HVAC
* Tóm tắt
* Đọc thêm
* Phụ lục A: Các thành phần tạo nên Hệ thống HVAC

**Các nguyên tắc phân tích Hệ thống HVAC**

***Hệ thống HVAC là gì?***

HVAC là một máy móc xử lý hệ thống sưởi ấm, thông gió và điều hòa không khí trong một tòa nhà.

***Tại sao Hệ thống HVAC quan trọng trong các tòa nhà thương mại?***

Hệ thống HVAC đóng vai trò quan trọng trong việc điều khiển nhiệt độ của tòa nhà, cung cấp sự lưu thông và lọc không khí tươi mát mà nếu không được bảo dưỡng đều đặn sẽ làm giảm năng suất làm việc, độ thoải mái và sức khỏe của người sử dụng.

***Mục tiêu của Hệ thống HVAC là gì?***

1. Điều chỉnh nhiệt độ của phòng để đảm bảo người sử dụng thoải mái.

2. Cung cấp lưu thông không khí tươi từ bên ngoài vào phòng và giảm thiểu sự tích tụ CO2, từ đó đảm bảo sức khỏe tốt hơn và hiệu suất làm việc tốt hơn tại nơi làm việc.

3. Cung cấp lọc không khí bằng cách sử dụng bộ lọc ở các giai đoạn khác nhau cho không khí đã được xử lý lại, vì không khí này có thể mang các chất gây ô nhiễm và bụi, gây ra các bệnh về hô hấp.

4. Tiết kiệm năng lượng và tiết kiệm chi phí do chi phí nhiên liệu tăng cao.

***Các cơ sở nào cần tính năng lọc không khí của Hệ thống HVAC là quan trọng nhất?***

* Phòng sạch
* Cơ sở y tế
* Vật liệu nguy hiểm

**Dữ liệu cảm biến HVAC**

***Dữ liệu cảm biến được thu thập từ đâu?***

Dữ liệu cảm biến được thu thập từ hoạt động của tòa nhà. Có các hệ thống HVAC trong 20phần trên không hoàn chỉnh và có vẻ bị cắt đoạn. Tuy nhiên, dựa trên phần đã được đưa ra, hãy cung cấp một số thông tin cơ bản về các khái niệm phát triển ứng dụng.

***Phát triển ứng dụng***

Phát triển ứng dụng là quá trình xây dựng các phần mềm hoặc ứng dụng máy tính có mục đích phục vụ một nhu cầu cụ thể của người dùng. Quá trình này bao gồm các bước từ việc nghiên cứu và thiết kế, triển khai và kiểm thử, cho đến triển khai và duy trì ứng dụng.

Dưới đây là một số khái niệm cơ bản liên quan đến phát triển ứng dụng:

***1. Nghiên cứu và thiết kế***

Trước khi xây dựng ứng dụng, cần thực hiện nghiên cứu và thiết kế. Bước nghiên cứu giúp hiểu rõ nhu cầu của người dùng và đánh giá khả năng kỹ thuật để thực hiện ứng dụng. Sau đó, bước thiết kế tạo ra các kế hoạch, sơ đồ, và giao diện người dùng để định hình cấu trúc và chức năng của ứng dụng.

***2. Triển khai***

Bước triển khai bao gồm việc viết mã, xây dựng các thành phần và tích hợp chúng thành một ứng dụng hoàn chỉnh. Trong quá trình này, lập trình viên sử dụng ngôn ngữ lập trình và các framework để tạo ra các chức năng và tính năng của ứng dụng.

***3. Kiểm thử***

Sau khi triển khai, quá trình kiểm thử được thực hiện để đảm bảo rằng ứng dụng hoạt động như mong đợi và không có lỗi. Kiểm thử có thể bao gồm các bài kiểm tra chức năng, kiểm tra hiệu suất, kiểm tra bảo mật và các bài kiểm tra khác để đảm bảo chất lượng của ứng dụng.

***4. Triển khai và duy trì***

Sau khi ứng dụng đã qua quá trình kiểm thử, nó có thể được triển khai để sử dụng rộng rãi cho người dùng. Sau khi triển khai, việc duy trì và hỗ trợ ứng dụng là quan trọng để giữ cho nó hoạt động ổn định và đáp ứng nhu cầu của người dùng. Các bản vá lỗi và cập nhật có thể được phát hành để cải thiện và mở rộng tính năng của ứng dụng.

***5. Giao diện người dùng***

Giao diện người dùng (GUI) là phần của ứng dụng mà người dùng tương tác trực tiếp. Nó cung cấp các phần tử và công cụ để người dùng nhập dữ liệu, xem thông tin và tương tác với ứng dụng. Các giao diện người dùng thường được thiết kế để tối ưu hóa trải nghiệm người dùng và đảm bảo tính thân thiện và dễ sử dụng.

**Tóm tắt**

Xin chúc mừng! Bây giờ bạn đã quen thuộc với mục đích của một Hệ thống HVAC. Bạn đã có ý tưởng về các cảm biến thông thường được sử dụng trong Hệ thống HVAC và cách dữ liệu từ những hệ thống này có thể được phân tích để xác định xem một hệ thống có cần được thay thế hay không. Với nền tảng này, bạn đã sẵn sàng để bắt đầu xây dựng ứng dụng.

**Đọc thêm**

* Video HVAC: 2-Fundamentals of HVAC - Basics of HVAC
* Bài viết HVAC: [5 HVAC Sensors You Need To Know About](https://www.primexvents.com/5-hvac-sensors-you-need-to-know-about/)
* Video Demo: Phân tích dữ liệu cảm biến

**Phụ lục A: Các thành phần tạo nên Hệ thống HVAC**

* **Máy lạnh/Máy điều hòa không khí**: sử dụng trao đổi nhiệt hoặc chất lỏng hoặc khí được lưu thông để làm lạnh không khí được thông qua nó
* **Bộ xử lý không khí**: là một quạt hoặc bơm thông gió đi qua bộ lọc chung sau đó đưa không khí vào hệ thống ống dẫn của tòa nhà
* **Bộ lọc không khí**: bao gồm các loại bộ lọc không khí khác nhau được sử dụng trong hệ thống tùy thuộc vào yêu cầu của người sử dụng và các hoạt động trong tòa nhà
* **HEPA**: bộ lọc tinh vi được sử dụng trong hệ thống ống dẫn hạ xuống
* **Ống dẫn không khí**: là ống dẫn tròn và hình chữ nhật, cung cấp một đường đi cho không khí điều chỉnh từ bộ xử lý không khí đến môi trường
* **Van điều khiển**: bao gồm một hoặc nhiều lá van được sử dụng để kiểm soát lưu lượng không khí thông qua ống dẫn
* **Van điều khiển thủ công**: đảm bảo các phần khác nhau của tòa nhà nhận được thông gió tỷ lệ theo diện tích và yêu cầu
* **Van điều khiển tự động**: thường được lắp đặt tại vách chống cháy và được đóng khi có hỏa hoạn
* **Đơn vị kết nối**: sử dụng van điều khiển tự động để kiểm soát lượng không khí được cung cấp đến một phòng hoặc khu vực
* **Động cơ điện khí hoặc số học số**: được điều chỉnh bởi bộ điều khiển nhiệt độ và điều khiển van tự động được sử dụng bởi đơn vị kết nối
* **Khu vực**: là một tập hợp các khu vực hoặc phòng trong một tòa nhà, có cùng nhu cầu về sưởi và làm lạnh. Thông thường, mỗi khu vực được gắn kết với một đơn vị kết nối và một bộ điềPhụ lục A: Các thành phần tạo nên Hệ thống HVAC
* **Chiller/Máy điều hòa không khí**: sử dụng bộ trao đổi nhiệt hoặc chất lỏng hoặc khí được tuần hoàn để làm lạnh không khí được thông qua nó.
* **Điều hòa không khí**: là quạt hoặc bơm thông gió đẩy không khí qua bộ lọc chung, sau đó đưa không khí đi qua hệ thống ống dẫn trong toàn bộ tòa nhà.
* **Bộ lọc không khí**: bao gồm các loại bộ lọc không khí khác nhau được sử dụng trong hệ thống tùy thuộc vào yêu cầu của người sử dụng và hoạt động trong tòa nhà.
* **HEPA**: bộ lọc tinh vi được sử dụng trong ống dẫn dưới lòng đất.
* **Hệ thống ống dẫn**: bao gồm ống dẫn tròn và hình chữ nhật, cung cấp đường đi cho không khí điều chỉnh từ điều hòa không khí đến môi trường.
* **Van điều khiển**: bao gồm một hoặc nhiều lá van được sử dụng để kiểm soát lưu lượng không khí thông qua hệ thống ống dẫn.
* **Van điều khiển thủ công**: đảm bảo rằng các phần khác nhau của tòa nhà nhận được thông gió tỷ lệ theo diện tích và nhu cầu.
* **Van điều khiển tự động**: thường được lắp đặt tại vách chống cháy và đóng trong trường hợp xảy ra hỏa hoạn.
* **Đơn vị kết nối**: sử dụng van điều khiển tự động để điều chỉnh lượng không khí được cung cấp đến một phòng hoặc khu vực.
* **Động cơ điện khí hoặc kỹ thuật số**: được điều chỉnh bởi bộ điều khiển nhiệt độ và điều khiển van tự động được sử dụng bởi đơn vị kết nối.
* **Khu vực**: là một tập hợp các khu vực hoặc phòng trong tòa nhà, mỗi khu vực có nhu cầu làm lạnh và sưởi đồng nhất. Thông thường, mỗi khu vực được gắn kết với một đơn vị kết nối và một bộ điều khiển nhiệt độ.
* **Cuộn sưởi**: cung cấp cách hiệu quả để đưa nhiệt vào những vùng ít cần thiết và được điều khiển bởi cùng hệ thống điều khiển nhiệt độ như đơn vị kết nối.
* **Ống giảm ồn**: giảm tiếng ồn trong một đoạn ngắn của hệ thống ống dẫn.
* **Lưới thông gió, đăng ký, ống thoát**: là cách để không khí từ hệ thống ống dẫn vào không gian làm việc.
* **Lượt hút không khí**: sau khi không khí đi vào không gian làm việc, nó sẽ lưu thông qua lượt hút và trở lại đến bộ xử lý không khí.

Để biết thêm thông tin về các thành phần tạo nên Hệ thống HVAC, xin tham khảo Video HVAC:

[2-Fundamentals of HVAC – Basics of VAC]

**THIẾT LẬP MÔI TRƯỜNG PHÁT TRIỂN**

**Giới thiệu**

Mục tiêu của chúng ta trong phần xây dựng ứng dụng phân tích hệ thống HVAC là thiết lập môi trường phát triển bằng mã shell, để chúng ta có thể tiếp tục tiếp cận dữ liệu, làm sạch nó và trực quan hóa các khía cạnh quan trọng để hiển thị thông tin chi tiết cho client. Chúng ta sẽ làm sạch bảng điều khiển NiFi trên HDF, để bắt đầu từ đầu mà không có luồng dữ liệu tồn tại trước đó.

**Yêu cầu tiên quyết**

* Đã kích hoạt CDA cho hệ thống phù hợp của bạn.

**Tổng quan**

* Xác minh đã hoàn thành các yêu cầu tiên quyết
* Tổng quan về Mã Shell được sử dụng trong cả hai phương pháp
* Phương pháp 1: Thiết lập Môi trường Phát triển bằng cách Thủ công
* Phương pháp 2: Thiết lập Môi trường Phát triển tự động
* Tóm tắt
* Đọc thêm

**Xác minh đã hoàn thành các yêu cầu tiên quyết**

***Ánh xạ IP của sandbox thành tên máy chủ mong muốn trong tệp hosts***

* Nếu bạn cần trợ giúp ánh xạ IP Sandbox thành tên máy chủ, xem Environment Setup -> Map Sandbox IP To Your Desired Hostname In The Hosts File trong [Learning the Ropes of HDP Sandbox](https://hortonworks.com/tutorial/learning-the-ropes-of-the-hortonworks-sandbox/)

***Thiết lập mật khẩu quản trị Ambari cho "HDF" và "HDP"***

Nếu bạn cần trợ giúp đặt mật khẩu quản trị Ambari,

* Đối với HDP, xem Admin Password Reset trong Learning the Ropes of HDP Sandbox
* Đối với HDF, xem Admin Password Reset trong Learning the Ropes of HDF Sandbox

***Khởi động tất cả các dịch vụ cần thiết cho "HDF" và "HDP"***

Nếu không chắc chắn, đăng nhập vào Bảng điều khiển Ambari admin

* Đối với HDF tại http://sandbox-hdf.hortonworks.com:8080 và xác minh rằng NiFi đã khởi động, nếu chưa, hãy khởi động nó.
* Đối với HDP tại http://sandbox-hdp.hortonworks.com:8080 và xác minh rằng HDFS, Hive và Zeppelin đã khởi động, nếu chưa, hãy khởi động chúng.

**Tổng quan về Mã Shell được sử dụng trong cả hai phương pháp**

Trong phương pháp 1, bạn sẽ chạy mã theo từng dòng để thiết lập môi trường phát triển. Tuy nhiên,trong phương pháp 2, bạn sẽ tải xuống và thực thi một tập lệnh shell để tự động thiết lập môi trường phát triển. Mã shell tee -a <file> << EOF ... EOF được sử dụng để thêm DNS Công cộng của Google vào /etc/resolve.conf để dịch một loạt các tên máy chủ thành địa chỉ IP. Vấn đề mà cấu hình này có thể giải quyết là sử dụng DNS Công cộng của Google để dịch s3.amazonaws.com thành địa chỉ IP mà các dịch vụ trong ứng dụng có thể sử dụng để truy cập dữ liệu. Một số dòng mã tiếp theo khởi tạo các biến sẽ được sử dụng trong các cuộc gọi API REST của Ambari. Hàm wait() đợi cho đến khi dịch vụ Ambari trên HDF hoặc HDP dừng hoặc khởi động. Hai lệnh curl được sử dụng để gửi dữ liệu JSON bằng các cuộc gọi API REST của Ambari để yêu cầu Ambari Server ghi đè trạng thái dịch vụ NiFi hiện có thành INSTALLED (nghĩa là STOPPED) hoặc STARTED. Đầu tiên, chúng ta thông báo với Ambari rằng chúng ta muốn DỪNG dịch vụ NiFi. Tiếp theo, chúng ta sao lưu và xóa luồng NiFi hiện có để làm sạch bảng điều khiển NiFi, để chúng ta có thể phát triển. Cuối cùng, chúng ta thông báo với Ambari để BẮT ĐẦU dịch vụ NiFi để các thay đổi có hiệu lực.

**Phương pháp 1: Thiết lập Môi trường Phát triển bằng cách Thủ công**

***Thiết lập HDF***

Chúng ta sẽ sử dụng các lệnh shell để thiết lập các dịch vụ cần thiết trên các nền tảng dữ liệu trong chuyển động và dữ liệu trong tĩnh từ các client shell web trong sandbox.

Mở client shell web HDF tại <http://sandbox-hdf.hortonworks.com:4200>

Trước khi thực thi mã shell, thay thế chuỗi "<Your-Ambari-Admin-Password>" trong dòng mã sau setup\_nifi "admin" "<Your-Ambari-Admin-Password>" ở dòng cuối cùng bằng mật khẩu bạn đã tạo cho người dùng quản trị ambari. Ví dụ, nếu mật khẩu Quản trị Ambari của chúng ta đã được đặt là yellowHadoop, thì dòng mã sẽ trông như sau: AMBARI\_USER\_PASSWORD="yellowHadoop"

Sao chép và dán mã shell sau vào client shell web HDF từng dòng một:

##

# Script sets up HDF services used in Building an HVAC System Analysis Application

##

DATE=`date '+%Y-%m-%d %H:%M:%S'`

LOG\_DIR\_BASE="/var/log/cda-sb/310/"

echo "$DATE INFO: Setting Up HDF Dev Environment for HVAC System Analysis App"

mkdir -p $LOG\_DIR\_BASE/hdf

setup\_public\_dns()

{

##

# Purpose of the following section of Code:

# NiFi GetHTTP will run into ERROR cause it can't resolve S3 Domain Name Server (DNS)

#

# Potential Solution: Append Google Public DNS to CentOS7 /etc/resolve.conf

# CentOS7 is the OS of the server NiFi runs on. Google Public DNS is able to

# resolve S3 DNS. Thus, GetHTTP can download HVAC Sensor Data from S3.

##

# Adding Public DNS to resolve msg: unable to resolv s3.amazonaws.com

# https://forums.aws.amazon.com/thread.jspa?threadID=125056

echo "$DATE INFO: Adding Google Public DNS to /etc/resolve.conf"

echo "# Google Public DNS" | tee -a /etc/resolve.conf

echo "nameserver 8.8.8.8" | tee -a /etc/resolve.conf

echo "$DATE INFO: Checking Google Public DNS added to /etc/resolve.conf"

cat /etc/resolve.conf

# Log everything, but also output to stdout

echo "$DATE INFO: Executing setup\_nifi bash function, logging to $LOG\_DIR\_BASE/hdf/setup-public-dns.log"

}

setup\_nifi()

{

##

# Purpose of the following section of Code:

# HDF Sandbox comes with a prebuilt NiFi flow, which causes user to be

# pulled away from building the HVAC System Analysis Application.

#

# Potential Solution: Backup prebuilt NiFi flow and call it a different name.

##

echo "$DATE INFO: Setting HDF\_AMBARI\_USER based on user input"

HDF\_AMBARI\_USER="$1" # $1: Expects user to pass "Ambari User" into the file

echo "$DATE INFO: Setting HDF\_AMBARI\_PASS based on user input"

HDF\_AMBARI\_PASS="$2" # $2: Expects user to pass "Ambari Admin Password" into the file

HDF\_CLUSTER\_NAME="Sandbox"

HDF\_HOST="sandbox-hdf.hortonworks.com"

HDF="hdf-sandbox"

AMBARI\_CREDENTIALS=$HDF\_AMBARI\_USER:$HDF\_AMBARI\_PASS

# Ambari REST Call Function: waits on service action completing

# Start Service in Ambari Stack and wait for it

# $1: HDF or HDP

# $2: Service

# $3: Status - STARTED or INSTALLED, but OFF

wait()

{

if [[ $1 == "hdp-sandbox" ]]

then

finished=0

while [ $finished -ne 1 ]

do

echo "$DATE INFO: Waiting for $1 $2 service action to finish"

ENDPOINT="http://$HDP\_HOST:8080/api/v1/clusters/$HDP\_CLUSTER\_NAME/services/$2"

AMBARI\_CREDENTIALS="$HDP\_AMBARI\_USER:$HDP\_AMBARI\_PASS"

str=$(curl -s -u $AMBARI\_CREDENTIALS $ENDPOINT)

if [[ $str == \*"$3"\* ]] || [[ $str == \*"Service not found"\* ]]

then

echo "$DATE INFO: $1 $2 service state is now $3"

finished=1

fi

echo "$DATE INFO: Still waiting on $1 $2 service action to finish"

sleep 3

done

elif [[ $1 == "hdf-sandbox" ]]

then

finished=0

while [ $finished -ne 1 ]

do

echo "$DATE INFO: Waiting for $1 $2 service action to finish"

ENDPOINT="http://$HDF\_HOST:8080/api/v1/clusters/$HDF\_CLUSTER\_NAME/services/$2"

AMBARI\_CREDENTIALS="$HDF\_AMBARI\_USER:$HDF\_AMBARI\_PASS"

str=$(curl -s -u $AMBARI\_CREDENTIALS $ENDPOINT)

if [[ $str == \*"$3"\* ]] || [[ $str == \*"Service not found"\* ]]

then

echo "$DATE INFO: $1 $2 service state is now $3"

finished=1

fi

echo "$DATE INFO: Still waiting on $1 $2 service action to finish"

sleep 3

done

else

echo "$DATE ERROR: Didn't Wait for Service, sandbox chosen not valid"

fi

}

# Stop NiFi first, then backup prebuilt NiFi flow, then start NiFi for

# changes to take effect

echo "$DATE INFO: Stopping HDF NiFi Service via Ambari REST Call"

curl -u $AMBARI\_CREDENTIALS -H "X-Requested-By: ambari" -X PUT -d '{"RequestInfo":

{"context": "Stop NiFi"}, "ServiceInfo": {"state": "INSTALLED"}}' \

http://$HDF\_HOST:8080/api/v1/clusters/$HDF\_CLUSTER\_NAME/services/NIFI

echo "$DATE INFO: Waiting on HDF NiFi Service to STOP RUNNING via Ambari REST Call"

wait $HDF NIFI "INSTALLED"

echo "$DATE INFO: Prebuilt HDF NiFi Flow removed from NiFi UI, but backed up"

mv /var/lib/nifi/conf/flow.xml.gz /var/lib/nifi/conf/flow.xml.gz.bak

echo "$DATE INFO: Starting HDF NiFi Service via Ambari REST Call"

curl -u $AMBARI\_CREDENTIALS -H "X-Requested-By: ambari" -X PUT -d '{"RequestInfo":

{"context": "Start NiFi"}, "ServiceInfo": {"state": "STARTED"}}' \

http://$HDF\_HOST:8080/api/v1/clusters/$HDF\_CLUSTER\_NAME/services/NIFI

echo "$DATE INFO: Waiting on HDF NiFi Service to START RUNNING via Ambari REST Call"

wait $HDF NIFI "STARTED"

# Log everything, but also output to stdout

echo "$DATE INFO: Executing setup\_nifi bash function, logging to $LOG\_DIR\_BASE/hdf/setup-nifi.log"

}

setup\_public\_dns | tee -a $LOG\_DIR\_BASE/hdf/setup-public-dns.log

setup\_nifi "admin" "<Your-Ambari-Admin-Password>" | tee -a $LOG\_DIR\_BASE/hdf/setup-nifi.log

**Thiết lập HDP**

Mở HDP shell web client tại <https://sanbox-hdp.hortonworks.com:4200>. Sao chép và dán mã sau vào web client từng dòng một.

##

# Sets up HDP services used in Building an HVAC System Analysis Application

##

DATE=`date '+%Y-%m-%d %H:%M:%S'`

LOG\_DIR\_BASE="/var/log/cda-sb/310/"

echo "Setting Up HDP Dev Environment for HVAC System Analysis App"

# Creates /sandbox directory in HDFS

# allow read-write-execute permissions for the owner, group, and any other users

mkdir -p $LOG\_DIR\_BASE/hdp

setup\_hdfs()

{

echo "$DATE INFO: Creating /sandbox/sensor/hvac\_building and /sandbox/sensor/hvac\_machine"

sudo -u hdfs hdfs dfs -mkdir -p /sandbox/sensor/hvac\_building/

sudo -u hdfs hdfs dfs -mkdir /sandbox/sensor/hvac\_machine

echo "$DATE INFO: Setting permissions for hvac\_buildnig and hvac\_machine to 777"

sudo -u hdfs hdfs dfs -chmod -R 777 /sandbox/sensor/hvac\_building/

sudo -u hdfs hdfs dfs -chmod -R 777 /sandbox/sensor/hvac\_machine

echo "$DATE INFO: Checking both directories were created and permissions were set"

sudo -u hdfs hdfs dfs -ls /sandbox/sensor

}

# Log everything, but also output to stdout

echo "$DATE INFO: Executing setup\_hdfs bash function, logging to /var/log/cda-sb/310/setup-hdp.log"

setup\_hdfs | tee -a $LOG\_DIR\_BASE/hdp/setup-hdp.log

**Phương pháp 2: Thiết lập Môi trường Phát triển tự động**

Chúng ta sẽ tải xuống và thực thi một tập lệnh shell để tự động thiết lập các nền tảng dữ liệu trong chuyển động và dữ liệu tĩnh từ client shell web trong sandbox.

Mở client shell web HDF tại http://sandbox-hdf.hortonworks.com:4200.

Trước khi thực thi tập lệnh shell, thay thế dòng mã shell AMBARI\_USER\_PASSWORD="<Your-Ambari-Admin-Password>" bằng mật khẩu bạn đã tạo cho người dùng quản trị Ambari. Ví dụ, nếu mật khẩu Quản trị Ambari của chúng ta đã được đặt là yellowHadoop, thì dòng mã sẽ trông như sau: AMBARI\_USER\_PASSWORD="yellowHadoop"

Sao chép và dán mã shell sau vào client shell web HDF với các thay đổi đã cập nhật của bạn:

AMBARI\_USER="admin"

AMBARI\_USER\_PASSWORD="<Your-Ambari-Admin-Password>"

wget https://raw.githubusercontent.com/hortonworks/data-tutorials/master/tutorials/cda/building-an-hvac-system-analysis-application/application/setup/shell/setup-hdf.sh

bash setup-hdf.sh $AMBARI\_USER $AMBARI\_USER\_PASSWORD

wget https://raw.githubusercontent.com/hortonworks/data-tutorials/master/tutorials/cda/building-an-hvac-system-analysis-application/application/setup/shell/setup-hdp.sh

bash setup-hdp.sh

**Tóm tắt**

Xin chúc mừng! Bây giờ bạn đã có môi trường phát triển sẵn sàng để bắt đầu xây dựng ứng dụng phân tích hệ thống HVAC. Các dịch vụ cần thiết cho việc phát triển ứng dụng đã được thiết lập, vì vậy chúng ta có thể bắt đầu xây dựng đường ống dữ liệu.

**Đọc thêm**

* NiFi Rest API
* Ambari Rest API
* Learning the bash Shell: Unix Shell Programming

**LẤY DỮ LIỆU CẢM BIẾN HVAC**

**Giới thiệu**

Bạn đã được tham gia vào dự án với vai trò Kỹ sư dữ liệu có các trách nhiệm sau: lấy dữ liệu cảm biến HVAC, chia dữ liệu thành hai luồng riêng biệt và lưu chúng vào các vị trí tương ứng trong hệ thống lưu trữ. Bạn đã được giao nhiệm vụ sử dụng [Apache NiFi](https://nifi.apache.org/docs.html) để xây dựng phần này của đường ống dữ liệu.

**Điều kiện tiên quyết**

* Bật CDA cho hệ thống phù hợp của bạn
* Thiết lập Môi trường phát triển

**Tổng quan**

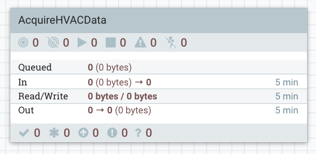
* Phương pháp 1: Xây dựng Luồng NiFi để lấy dữ liệu cảm biến HVAC
* Phương pháp 2: Nhập Nhóm quá trình AcquireHVACData của NiFi thông qua giao diện người dùng
* Phương pháp 3: Tự động triển khai Luồng NiFi thông qua REST Call
* Tóm tắt
* Đọc thêm

**Phương pháp 1: Xây dựng Luồng NiFi để lấy dữ liệu cảm biến HVAC**

Sau khi khởi động môi trường của bạn, mở HDF **NiFi UI** tại <http://sandbox-hdf.hortonworks.com:9090/nifi>

**Tạo Nhóm quá trình AcquireHVACData**

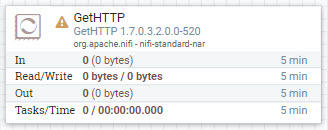
Kéo biểu tượng nhómprocess_group xử lý hoặc tên tuỳ chọn của bạn.



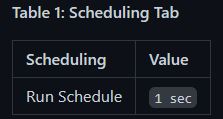
Nhấp đúp vào nhóm xử lý để đi vào bên trong nó. Ở phía dưới mặt vẽ, bạn sẽ thấy dòng dẫn **NiFi Flow >> AcquireHVACData**. Hãy bắt đầu kết nối các bộ xử lý để thu thập dữ liệu, tiền xử lý và lưu trữ.

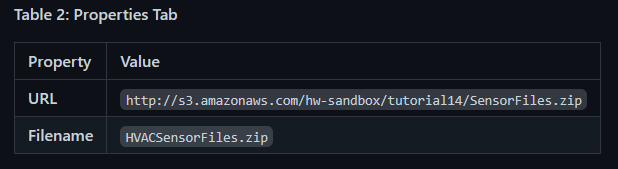
**Lấy nguồn dữ liệu cảm biến**

Thả biểu tượng bộ xử lý vào bảng vẽ NiFi. Thêm **GetHTTP**.



Giữ **control + click chuột** vào **GetHTTP** để cấu hình bộ xử lý





Nhấp Apply.

**Giải nén nội dung của FlowFile**

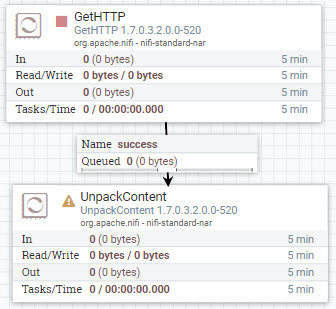
Thả biểu tượng bộ xử lý vào bảng vẽ NiFi. Thêm UnpackContent.

Tạo kết nối giữa các bộ xử lý GetHTTP và UnpackContent. Di chuột qua GetHTTP để xem biểu tượng mũi tên, nhấn vào bộ xử lý và kết nối nó với UnpackContent.

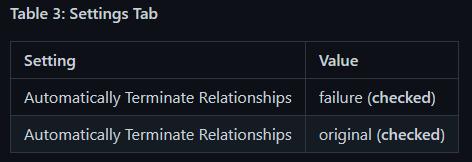
Cấu hình kết nối:

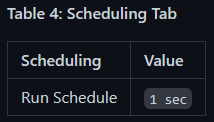


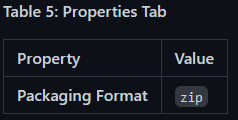
Bấm ADD.



Cấu hình bộ xử lý UnpackContent







Bấm APPLY.

**Định tuyến FlowFiles thành hai luồng dựa trên thuộc tính của chúng**

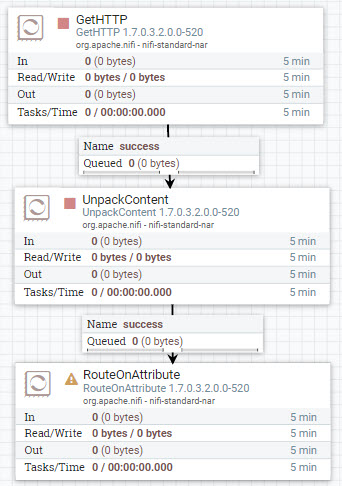
Thả biểu tượng bộ xử lý vào bảng vẽ NiFi. Thêm **RouteOnAttribute**.

Tạo kết nối giữa các bộ xử lý **UnpackContent** và **RouteOnAttribute**. Di chuột qua **UnpackContent** để xem biểu tượng mũi tên, nhấn vào bộ xử lý và kết nối nó với **RouteOnAttribute**.

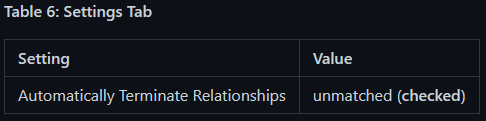
Cấu hình kết nối:

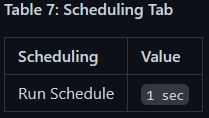


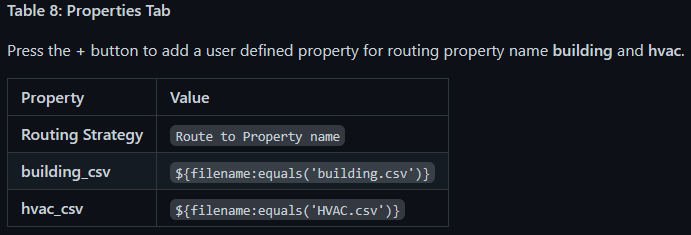
Bấm ADD.



Cấu hình bộ xử lý **RouteOnAttribute**:







Bấm APPLY.

**Lưu trữ dữ liệu vào HDFS**

Thả biểu tượng bộ xử lý vào bảng vẽ NiFi. Thêm 2 bộ xử lý **PutHDFS**.

Tạo kết nối giữa bộ xử lý **RouteOnAttribute** và cả hai bộ xử lý **PutHDFS**. Di chuột qua **RouteOnAttribute** để xem biểu tượng mũi tên, nhấn vào bộ xử lý và kết nối nó với **PutHDFS**. Lặp lại bước trước để kết nối với bộ xử lý **PutHDFS** khác.

Đối với một trong các bộ xử lý **PutHDFS**, cấu hình Kết nối Tạo:

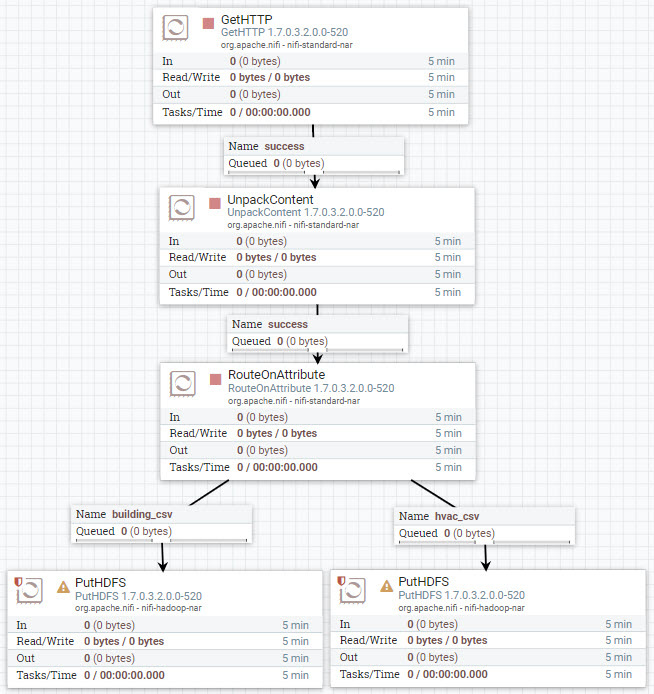
| **Kết nối** | **Giá trị** |
| --- | --- |
| Với Mối quan hệ | building\_csv (**đã chọn**) |

Nhấp vào **ADD**.

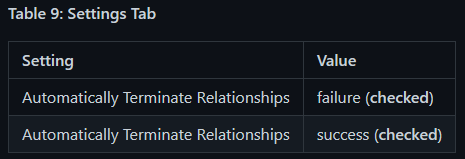
Đối với bộ xử lý **PutHDFS** khác, cấu hình Kết nối Tạo:

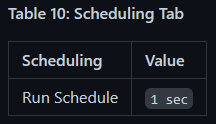
| **Kết nối** | **Giá trị** |
| --- | --- |
| Với Mối quan hệ | hvac\_csv (**đã chọn**) |

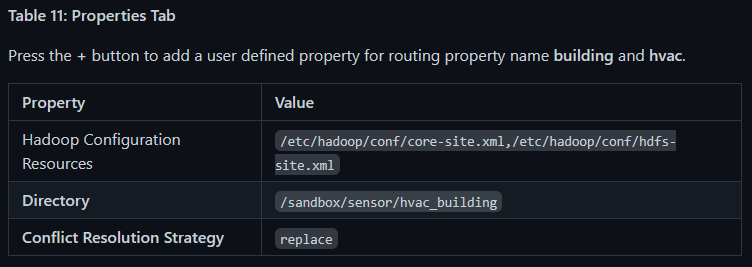
Nhấp vào **ADD**.



Cấu hình bộ xử lý **PutHDFS** cho kết nối mối quan hệ **building\_csv**:

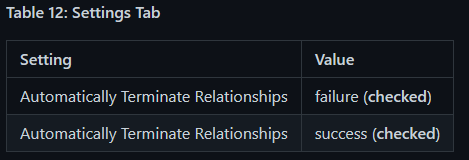


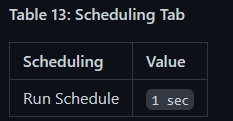


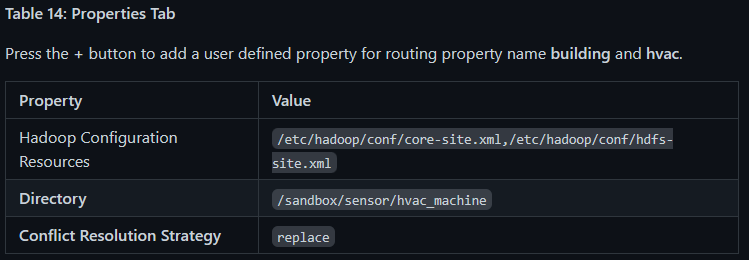


Bấm APPLY.

Cấu hình bộ xử lý **PutHDFS** cho kết nối mối quan hệ **hvac\_csv**



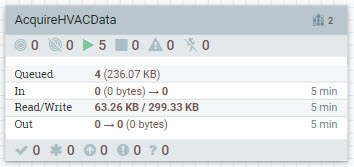




Nhấp APPLY.

**Bắt đầu quy trình nhóm xử lý để thu thập dữ liệu**

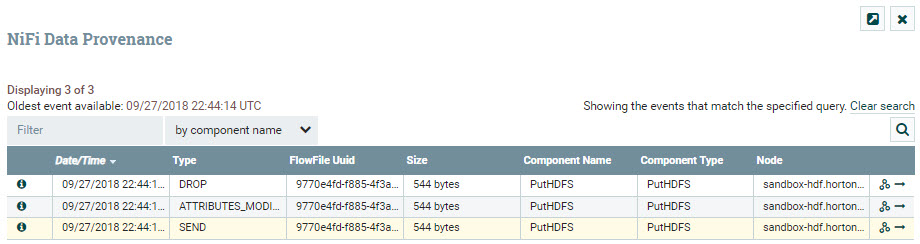
Trên thanh điều hướng, chọn cấp độ **Dòng NiFi**. Giữ **control + nhấp chuột** vào nhóm xử lý **AcquireHVACData**, sau đó chọn tùy chọn **start**.



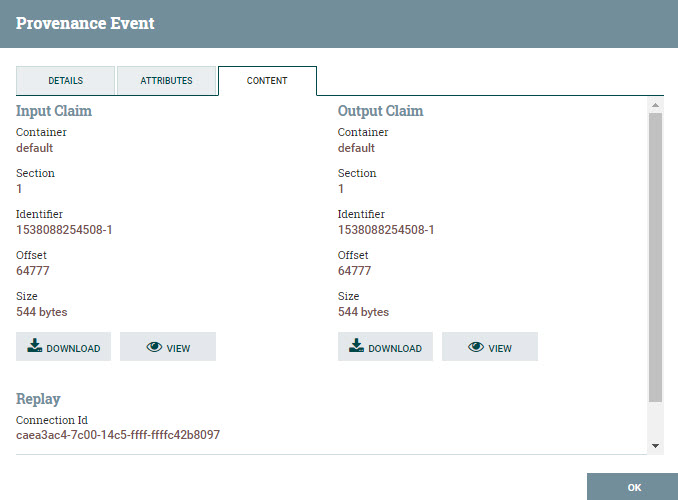
Sau khi NiFi ghi dữ liệu cảm biến của bạn vào HDFS, bạn có thể tắt nhóm xử lý bằng cách giữ **control + nhấp chuột** vào nhóm xử lý **AcquireHVACData**, sau đó chọn tùy chọn **stop**.

**Xác minh dữ liệu đã được lưu trữ bởi NiFi**

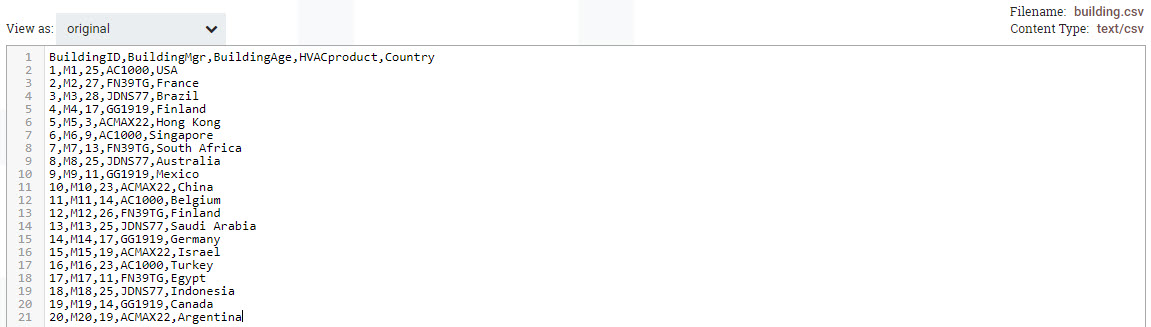
Vào nhóm xử lý **AcquireHVACData**, nhấn **control + nhấp chuột** vào bộ xử lý PutHDFS mà bạn chọn, sau đó nhấn **Xem nguồn gốc dữ liệu**.



Nhấp vào biểu tượng **i** trên hàng bên trái để xem chi tiết về một sự kiện nguồn gốc. Chọn sự kiện có loại **SEND**. Trong cửa sổ Sự kiện nguồn gốc, chọn tab **CONTENT**. Trên **Output Claim**, chọn **VIEW**.



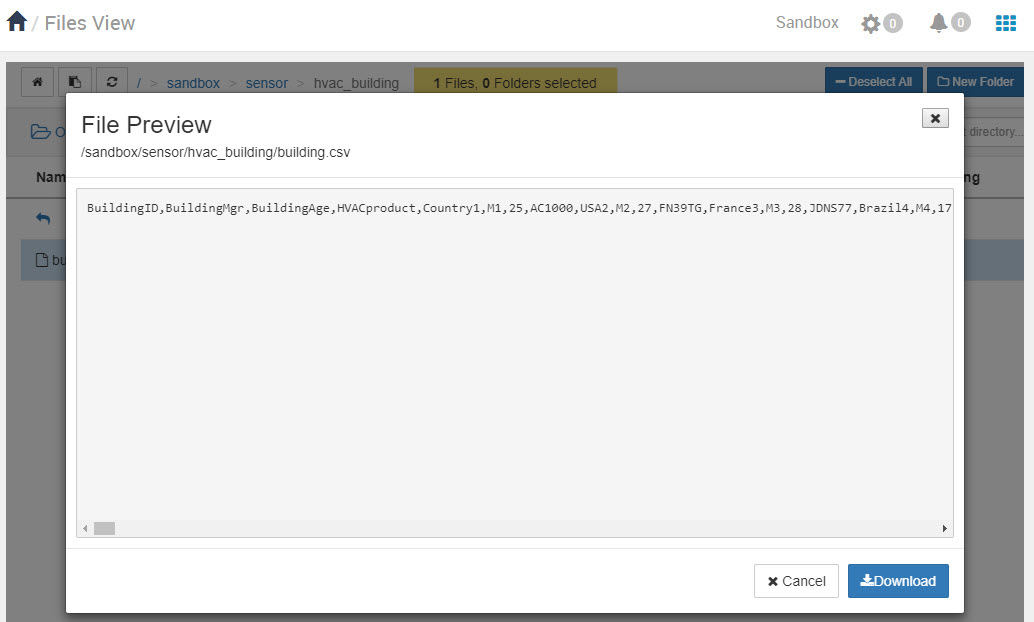
Bạn sẽ có thể xem dữ liệu mà NiFi đã gửi cho quá trình bên ngoài HDFS thông qua sự chứng thực dữ liệu của NiFi. Dữ liệu dưới đây hiển thị tập dữ liệu hvac.



Hãy xác minh rằng dữ liệu thực sự hiện có trong **HDFS** từ **Files View** của Ambari. Đăng nhập vào HDP Ambari tại địa chỉ <http://sandbox-hdp.hortonowrks.com:8080>.

Nhấp vào bộ chọn Ambari Views ở góc trên bên phải, sau đó nhấp vào **Files View**.

Nhập đường dẫn: /sandbox/sensor/hvac\_building, bạn sẽ có thể xem dữ liệu .csv cho **building.csv**. Tệp HVAC.csv nằm trong thư mục hvac\_temperature.



Dữ liệu **building.csv** có vẻ bị méo, nhưng khi chúng ta làm sạch nó bằng Apache Hive, nó sẽ được định dạng đúng trong một bảng.

**Cách tiếp cận 2: Nhập quy trình nhóm xử lý AcquireHVACData của NiFi thông qua giao diện người dùng**

Truy cập HDF Ambari tại địa chỉ [http://sandbox-hdp.hortonowrks.com:8080](http://sandbox-hdp.hortonowrks.com:8080/) với người dùng đăng nhập là admin và mật khẩu bạn đã đặt, sau đó truy cập vào giao diện người dùng NiFi bằng cách sử dụng Ambari Quick Link và nhập mẫu NiFi.

Tải xuống mẫu NiFi [acquire-hvac-data.xml](https://github.com/xvanausloos/hdp_data_tutorials/blob/master/tutorials/cda/building-an-hvac-system-analysis-application/application/development/nifi-template/acquire-hvac-data.xml)về máy tính cục bộ của bạn.

Sau khi bắt đầu môi trường sandbox, mở **NiFi UI** của HDF tại <http://sandbox-hdf.hortonworks.com:9090/nifi>.

Mở bảng điều hành nếu chưa mở, sau đó nhấn vào biểu tượng **Tải lên Mẫu**upload.

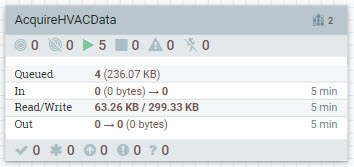
Nhấp vào biểu từng **Chọn mẫu**search_template.

Trình duyệt tệp trên máy tính cục bộ của bạn sẽ hiển thị, tìm tệp mẫu **acquire-hvac-data.xml** bạn vừa tải xuống, sau đó nhấp **Mở**, sau đó nhấp **Tải lên**.Bạn sẽ nhận được thông báo "Mẫu đã được nhập thành công." Nhấp OK để xác nhận.

Thả biểu tượng **Mẫu**templatevào bề mặt NiFi.

Thêm Mẫu có tên **acquire-hvac-data**.

Bắt đầu luồng NiFi. Giữ **control + nhấp chuột** vào nhóm xử lý **AcquireHVACData**, sau đó chọn tùy chọn **start**.



Sau khi NiFi ghi dữ liệu cảm biến của bạn vào HDFS, bạn có thể kiểm tra nhanh bằng cách nhìn vào các bộ xử lý PutHDFS bên trong nhóm xử lý, bạn có thể tắt nhóm xử lý bằng cách giữ **control + nhấp chuột** vào nhóm xử lý **AcquireHVACData**, sau đó chọn tùy chọn **stop**.

**Cách tiếp cận 3: Triển khai tự động luồng NiFi qua REST Call**

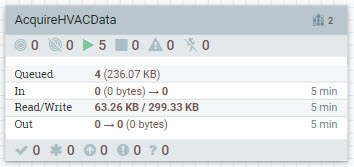
Mở HDF Sandbox Web Shell Client tại địa chỉ [http://sandbox-hdf.hortonworks.com:4200](http://sandbox-hdf.hortonworks.com:4200/). Sao chép và dán mã shell sau:

NIFI\_TEMPLATE="acquire-hvac-data"

wget https://raw.githubusercontent.com/hortonworks/data-tutorials/master/tutorials/cda/building-an-hvac-system-analysis-application/application/development/shell/nifi-auto-deploy.sh

bash nifi-auto-deploy.sh $NIFI\_TEMPLATE

Mở **NiFi UI** của HDF tại địa chỉ <http://sandbox-hdf.hortonworks.com:9090/nifi>. Bạn sẽ thấy mẫu NiFi đã được tải lên, nhập và bắt đầu.



Sau khi NiFi ghi dữ liệu cảm biến của bạn vào HDFS, bạn có thể kiểm tra nhanh bằng cách nhìn vào các bộ xử lý PutHDFS bên trong nhóm xử lý, bạn có thể tắt nhóm xử lý bằng cách giữ **control + nhấp chuột** vào nhóm xử lý **AcquireHVACData**, sau đó chọn tùy chọn **stop**.

**Tóm tắt**

Xin chúc mừng! Bạn đã thêm một luồng dữ liệu NiFi vào đường ống dữ liệu để tiếp nhận dữ liệu từ các cảm biến được nhúng trên hệ thống máy HVAC, chuẩn bị dữ liệu để có thể đọc được cho các dịch vụ bên ngoài thực hiện phân tích dữ liệu. Trong phần tiếp theo của đường ống dữ liệu, bạn sẽ sử dụng Apache Hive để thực hiện làm sạch và phân tích dữ liệu.

**Đọc thêm**

- [Hướng dẫn người dùng Apache NiFi](https://nifi.apache.org/docs/nifi-docs/html/user-guide.html)

**LÀM SẠCH DỮ LIỆU HVAC THÔ**

**Giới thiệu**

Mục tiêu tiếp theo của bạn là làm sạch dữ liệu cảm biến HVAC từ các máy HVAC và tòa nhà HVAC thành một định dạng hữu ích bằng cách sử dụng Hive Query Language tương tự SQL và Apache Zeppelin's JDBC Hive Interpreter. Bạn sẽ có cơ hội trải nghiệm thực tế với việc tạo bảng Hive chạy trên các tệp ORC để xử lý dữ liệu nhanh chóng và hiệu quả. Bạn sẽ hiểu được cách viết các kịch bản Hive để làm giàu dữ liệu của chúng ta để tiết lộ cho chúng ta khi nhiệt độ là lạnh, bình thường hay nóng. Ngoài ra, bạn sẽ học cách viết các truy vấn Hive trên dữ liệu để xác định các tòa nhà cụ thể nào liên quan đến các trạng thái nhiệt độ này.

**Yêu cầu tiên quyết**

* Bật CDA cho hệ thống phù hợp của bạn
* Cài đặt môi trường phát triển
* Nhận dữ liệu cảm biến HVAC

**Đề cương**

* Tải dữ liệu cảm biến HVAC vào bảng
* Tinh chỉnh dữ liệu cảm biến thô
* Tóm tắt
* Đọc thêm

**Tải dữ liệu cảm biến HVAC vào bảng**

Mở Ambari UI tại [http://sandbox-hdp.hortonworks.com:8080](http://sandbox-hdp.hortonworks.com:8080/) và đăng nhập với tài khoản admin và mật khẩu mà bạn đã đặt.

Trước khi chúng ta có thể làm sạch dữ liệu bằng Apache Hive, chúng ta cần tải dữ liệu cảm biến HVAC vào các Bảng Hive.

Có một vài cách khác nhau để nhập dữ liệu CSV vào Bảng Hive. Chúng ta có thể sử dụng Data Analytics Studio, trình thông dịch Hive JDBC của Zeppelin hoặc Hive shell. Trong ví dụ này, chúng ta sẽ sử dụng trình thông dịch Hive JDBC của Zeppelin để chuyển dữ liệu cảm biến HVAC vào các Bảng Hive.

**Mở giao diện Zeppelin từ Ambari**

Nhấp vào dịch vụ **Zeppelin Notebook** trong Ambari stack, trong hộp bên phải gọi là **Quick Links**, nhấp vào [**Zeppelin UI**](http://sandbox-hdp.hortonworks.com:9995/).

Nhấp vào **Create new note**.

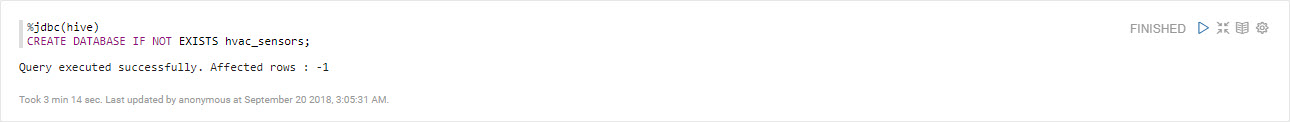
Nhập **Note Name** là Cleaning-Raw-HVAC-Data, chọn jdbc làm trình thông dịch mặc định, sau đó nhấp **Create**.

**Tạo cơ sở dữ liệu HVAC Sensor Data**

Sao chép và dán mã Hive sau vào Zeppelin:Tạo cơ sở dữ liệu **hvac\_sensors**. Sao chép và dán truy vấn Hive này, nhấn **shift + enter** để thực thi mã:

%jdbc(hive)

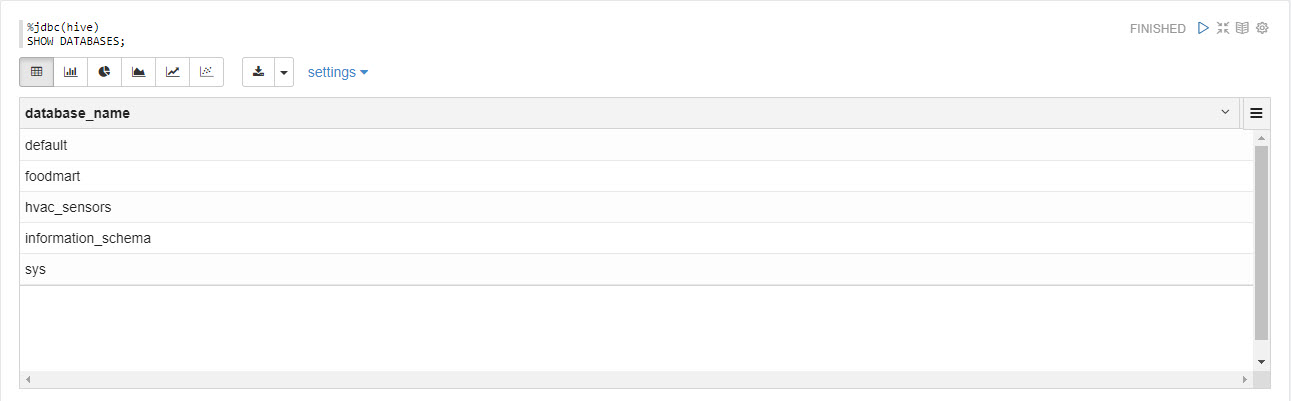
CREATE DATABASE IF NOT EXISTS hvac\_sensors;



Xác minh rằng cơ sở dữ liệu **hvac\_sensors** đã được tạo thành công:

%jdbc(hive)

SHOW DATABASES;



**Tải dữ liệu CSV của tòa nhà HVAC vào Bảng Hive**

Quá trình tải dữ liệu building.csv vào Hive yêu cầu chúng ta tạo một bảng ngoại vi để xác minh rằng Hive có quyền truy cập vào dữ liệu, nhưng không sở hữu dữ liệu CSV gốc để đảm bảo an toàn khỏi việc bị xóa trong trường hợp bảng ngoại vi bị xóa. Tiếp theo, chúng ta tạo một bảng nội bộ có tên là building, có định dạng ORC và chuyển dữ liệu từ bảng ngoại vi vào bảng nội bộ, để dữ liệu thuộc sở hữu của Hive, nhưng dữ liệu CSV gốc vẫn an toàn.

**Tạo Bảng Hive Ngoại vi**

Trước tiên, chúng ta sẽ tạo một bảng ngoại vi tham chiếu đến dữ liệu CSV của tòa nhà HVAC.

%jdbc(hive)

CREATE EXTERNAL TABLE IF NOT EXISTS hvac\_sensors.building\_csv (

`BuildingID` INT,

`BuildingMgr` STRING,

`BuildingAge` INT,

`HVACproduct` STRING,

`Country` STRING

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE

LOCATION '/sandbox/sensor/hvac\_building'

TBLPROPERTIES("skip.header.line.count"="1");

Chúng ta đã tạo một bảng ngoại vi **building\_csv** trong cơ sở dữ liệu **hvac\_sensors**, chúng ta sử dụng ký tự backticks cho mỗi thuộc tính để tránh gặp vấn đề với các từ khóa dành riêng trong Hive, và thông báo cho Hive lưu trữ dữ liệu trong bảng dưới dạng textfile, và cho biết dữ liệu ban đầu của building.csv được lưu trữ trong thư mục /sandbox/sensor/hvac\_building. Chúng ta đã tạo một bảng ngoại vi trước để đảm bảo dữ liệu gốc an toàn khỏi việc bị xóa nếu bảng bị xóa.

Tiếp theo, chúng ta sẽ xác minh việc nhập dữ liệu thành công bằng cách in ra một số mẫu 5 dòng đầu tiên:

%jdbc(hive)

SELECT \* FROM hvac\_sensors.building\_csv LIMIT 5;

**Tạo Bảng Hive Nội bộ để Sao chép Dữ liệu từ Bảng Ngoại vi**

Bây giờ chúng ta sẽ tạo một bảng Hive nội bộ. Chúng ta cũng có nhiều tùy chọn định dạng tệp để lưu trữ dữ liệu trong bảng, chúng ta sẽ sử dụng **định dạng ORC của Apache** vì nó cung cấp khả năng nén tốt và hiệu suất tuyệt vời.

%jdbc(hive)

CREATE TABLE IF NOT EXISTS hvac\_sensors.building (

`BuildingID` INT,

`BuildingMgr` STRING,

`BuildingAge` INT,

`HVACproduct` STRING,

`Country` STRING

)

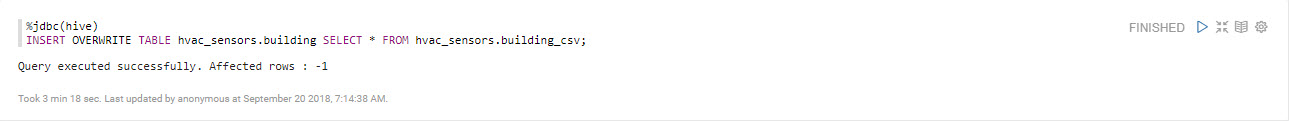
COMMENT 'Building holds HVAC product'

STORED AS ORC;

Sau khi đã tạo bảng nội bộ, chúng ta có thể sao chép dữ liệu từ bảng ngoại vi vào bảng hive nội bộ này.

%jdbc(hive)

INSERT OVERWRITE TABLE hvac\_sensors.building SELECT \* FROM hvac\_sensors.building\_csv;



Để xác minh rằng dữ liệu đã được sao chép thành công, giống như chúng ta đã làm trước đó, chúng ta có thể sử dụng truy vấn **select** để hiển thị 5 dòng đầu tiên của dữ liệu.

%jdbc(hive)

SELECT \* FROM hvac\_sensors.building LIMIT 5;

Tải dữ liệu từ tệp CSV của HVAC vào Bảng Hive

Quá trình tải dữ liệu từ tệp HVAC.csv vào Hive tương tự quá trình chúng ta đã thực hiện trước đó với tệp HVAC building.csv. Chúng ta sẽ thực hiện cùng quy trình cho HVAC.

**Tạo Bảng Hive External**

Chúng ta sẽ tạo một bảng Hive external trỏ đến dữ liệu máy HVAC trong tệp csv.

%jdbc(hive)

CREATE EXTERNAL TABLE IF NOT EXISTS hvac\_sensors.hvac\_machine\_csv (

`Date` STRING,

`Time` STRING,

`TargetTemp` INT,

`ActualTemp` INT,

`System` INT,

`SystemAge` INT,

`BuildingID` INT

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE

LOCATION '/sandbox/sensor/hvac\_machine'

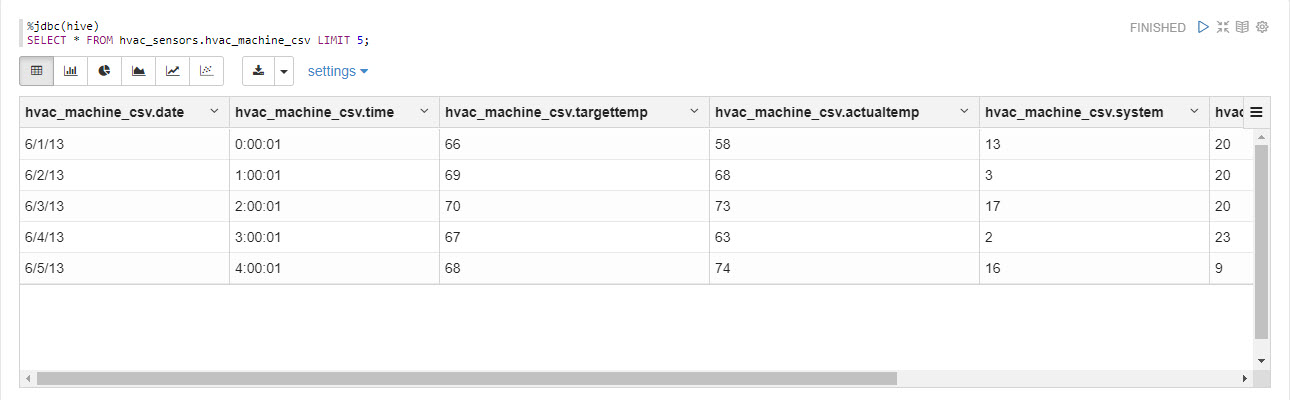
TBLPROPERTIES("skip.header.line.count"="1");



Hãy xác minh rằng việc nhập khẩu đã thành công bằng cách in ra một mẫu của 5 dòng đầu tiên:

%jdbc(hive)

SELECT \* FROM hvac\_sensors.hvac\_machine\_csv LIMIT 5;



**Tạo Bảng Hive Internal để Sao chép Dữ liệu từ Bảng External**

Bây giờ chúng ta sẽ tạo một bảng Hive internal với định dạng tệp lưu trữ là Apache ORC format.

%jdbc(hive)

CREATE TABLE IF NOT EXISTS hvac\_sensors.hvac\_machine (

`Date` STRING,

`Time` STRING,

`TargetTemp` INT,

`ActualTemp` INT,

`System` INT,

`SystemAge` INT,

`BuildingID` INT

)

COMMENT 'hvac\_machine holds data on attributes of the machine'

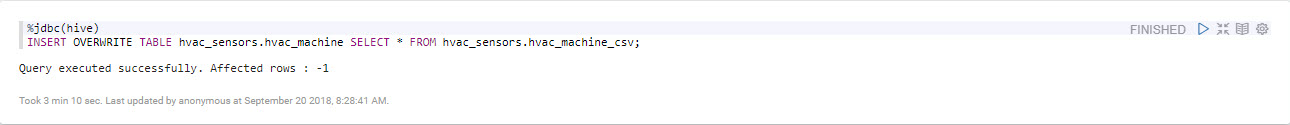
STORED AS ORC;



Sau khi tạo bảng internal, chúng ta có thể sao chép dữ liệu từ bảng external vào bảng Hive internal này.

%jdbc(hive)

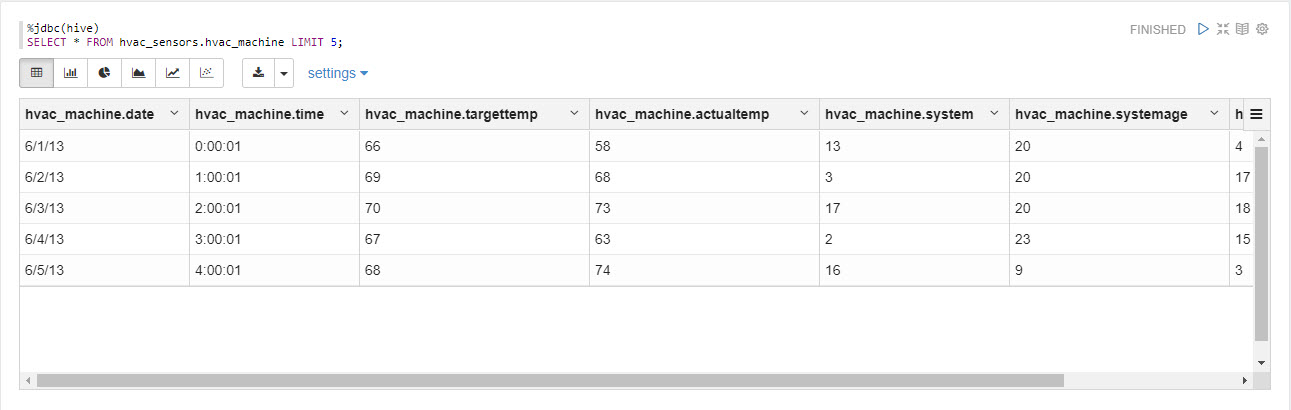
INSERT OVERWRITE TABLE hvac\_sensors.hvac\_machine SELECT \* FROM hvac\_sensors.hvac\_machine\_csv;



Bây giờ hãy xác minh rằng dữ liệu đã được sao chép từ bảng external vào bảng internal thành công bằng cách sử dụng.

%jdbc(hive)

SELECT \* FROM hvac\_sensors.hvac\_machine LIMIT 5;



**Tinh chỉnh dữ liệu cảm biến thô**

Chúng ta sẽ viết các script Hive để làm sạch dữ liệu cảm biến thô nhằm thu được những thông tin quan trọng sau:

* Giảm chi phí sưởi và làm mát
* Giữ nhiệt độ trong nhà ở khoảng 65-70 độ Fahrenheit (18-21 độ Celsius) để đảm bảo thoải mái
* Xác định các sản phẩm HVAC đáng tin cậy
* %jdbc(hive)
* CREATE TABLE hvac\_sensors.hvac\_temperatures AS
* SELECT \*, targettemp - actualtemp AS temp\_diff,
* IF((targettemp - actualtemp) > 5, 'COLD',
* IF((targettemp - actualtemp) < -5, 'HOT', 'NORMAL'))
* AS temprange,
* IF((targettemp - actualtemp) > 5, '1',
* IF((targettemp - actualtemp) < -5, '1', 0))
* AS extremetemp FROM hvac\_sensors.hvac\_machine;

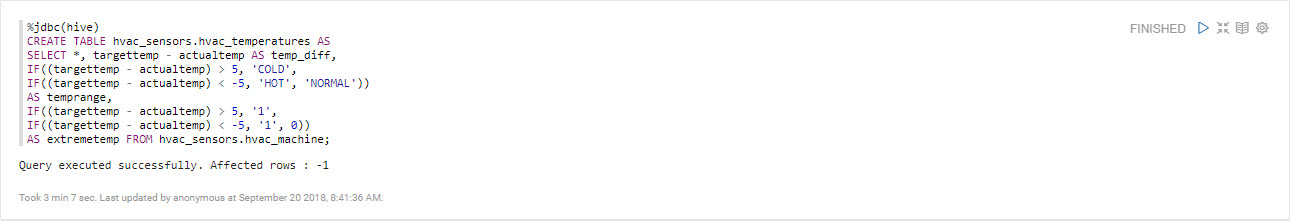
Lưu ý: Truy vấn trên có thể mất từ 10 đến 15 phút để hoàn thành

Truy vấn trên thực hiện công việc gì?

* Tạo một bảng mới **hvac\_temperatures** và sao chép dữ liệu từ bảng **hvac**
* Trên trang Kết quả truy vấn, sử dụng thanh trượt để cuộn sang phải. Bạn sẽ thấy hai thuộc tính mới xuất hiện trong bảng **hvac\_temperatures**

Hai thuộc tính mới đó là gì?

* **temprange** và **extremetemp**



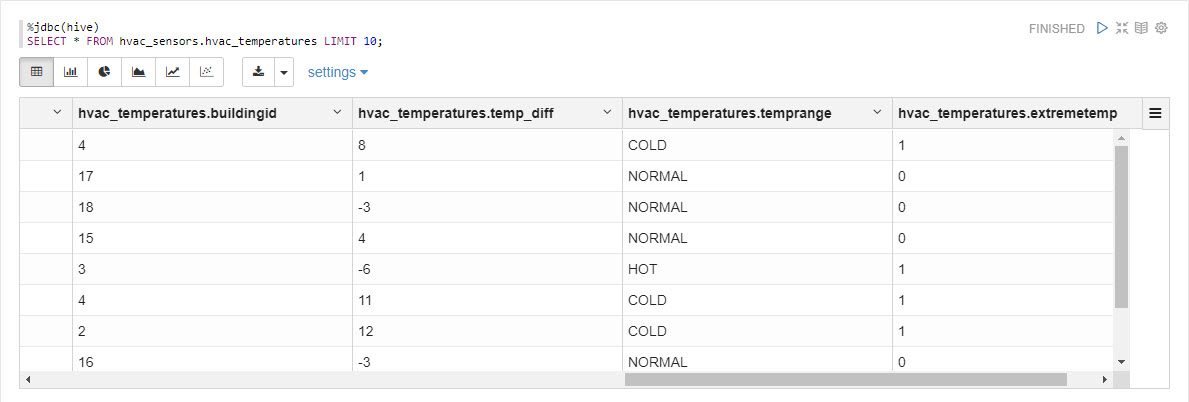
Hãy tải lên 10 dòng từ bảng **hvac\_temperatures**

%jdbc(hive)

SELECT \* FROM hvac\_sensors.hvac\_temperatures LIMIT 10;

Dữ liệu trong cột **temprange** cho biết gì về nhiệt độ thực tế?

* **NORMAL** - trong khoảng cách 5 độ so với nhiệt độ mục tiêu.
* **COLD** - lạnh hơn 5 độ so với nhiệt độ mục tiêu.
* **HOT** - nóng hơn 5 độ so với nhiệt độ mục tiêu.



Bây giờ chúng ta sẽ tạo một bảng kết hợp giữa hvac\_temperatures và buildings

%jdbc(hive)

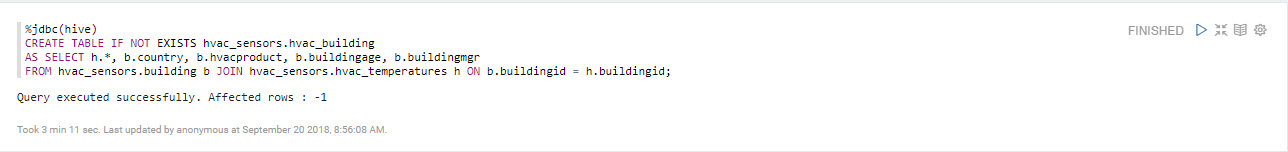
CREATE TABLE IF NOT EXISTS hvac\_sensors.hvac\_building

AS SELECT h.\*, b.country, b.hvacproduct, b.buildingage, b.buildingmgr

FROM hvac\_sensors.building b JOIN hvac\_sensors.hvac\_temperatures h ON b.buildingid = h.buildingid;

Dữ liệu của bảng **hvac\_sensors.hvac\_building** được lấy từ những bảng nào?

* **hvac\_temperature** và **buildings**

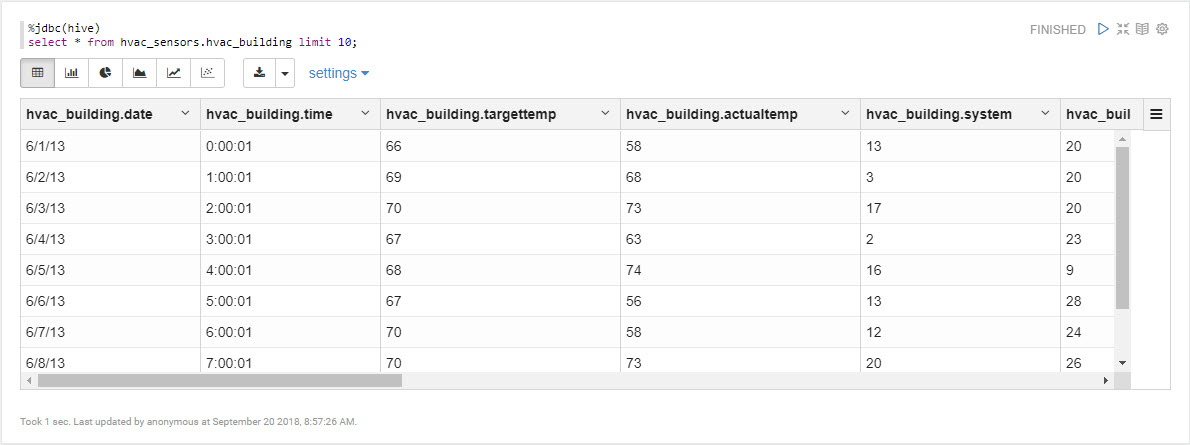


Sau khi truy vấn tạo thành công bảng **hvac\_sensors.hvac\_building**, bạn sẽ nhận được thông báo sau: Query executed successfully. Affected rows: -1.

Hãy tải lên 10 dòng dữ liệu từ bảng mới hvac\_sensors.hvac\_building

%jdbc(hive)

SELECT \* FROM hvac\_sensors.hvac\_building LIMIT 10;



**Tóm tắt**

Bạn đã thành công tải lên dữ liệu cảm biến HVAC từ tệp CSV vào Hive và tinh chỉnh dữ liệu thành một định dạng hữu ích. Bạn đã học cách tạo bảng Hivechạy trên các tệp ORC để xử lý dữ liệu nhanh chóng và hiệu quả. Bạn đã học cách viết các script Hive để làm phong phú dữ liệu của chúng ta để phân tích nhiệt độ thực tế có trạng thái lạnh, bình thường hoặc nóng. Bên cạnh đó, bạn đã sử dụng dữ liệu để tìm hiểu xem các tòa nhà cụ thể nào liên quan đến các mức nhiệt độ này: **NORMAL, HOT HOẶC COLD**. Bước tiếp theo của chúng ta là sử dụng các công cụ trực quan hóa dữ liệu khác của Zeppelin để kể câu chuyện về kết quả từ phân tích dữ liệu.

**Đọc thêm**

[Importing Data from CSV Files into Hive Tables](http://www.informit.com/articles/article.aspx?p=2756471&seqNum=4) (Tiếng Anh)

[How can I upload ORC files to Hive?](https://community.hortonworks.com/questions/47594/how-can-i-upload-ocr-files-to-hive.html) (Tiếng Anh)

[Optimizing Hive queries for ORC formatted tables](https://community.hortonworks.com/articles/68631/optimizing-hive-queries-for-orc-formatted-tables.html) (Tiếng Anh)

**TRỰC QUAN HÓA DỮ LIỆU CẢM BIẾN LIÊN QUAN ĐẾN HỆ THỐNG MÁY HVAC**

**Giới thiệu**

Mục tiếp theo của bạn là hành động như một Nhà phân tích dữ liệu và sử dụng các công cụ trực quan hóa tích hợp sẵn trong Apache Zeppelin để cung cấp những thông tin chi tiết về dữ liệu của bạn, chẳng hạn như minh họa các quốc gia có nhiệt độ cực đoan nhất và số lượng sự kiện **NORMAL** so với **HOT** và **COLD**. Bạn cũng sẽ minh họa những đơn vị HVAC nào dẫn đến việc đo nhiệt độ cực đoan nhất (extremetemp).

**Yêu cầu tiên quyết**

* Bật CDA cho hệ thống phù hợp của bạn
* Thiết lập Môi trường phát triển
* Thu thập dữ liệu cảm biến HVAC
* Làm sạch dữ liệu cảm biến HVAC

**Mô tả**

* Trực quan hóa đặc điểm nhiệt độ tòa nhà HVAC theo quốc gia
* Trực quan hóa nhiệt độ cực đoan trong các tòa nhà có Sản phẩm HVAC

**Trực quan hóa đặc điểm nhiệt độ tòa nhà HVAC theo quốc gia**

Nhấp vào dịch vụ **Zeppelin Notebook** trong giao diện Ambari, trong hộp bên phải gọi là **Quick Links**, nhấp vào [**Zeppelin UI**](http://sandbox-hdp.hortonworks.com:9995/).

Nhấp vào **Create new note**.

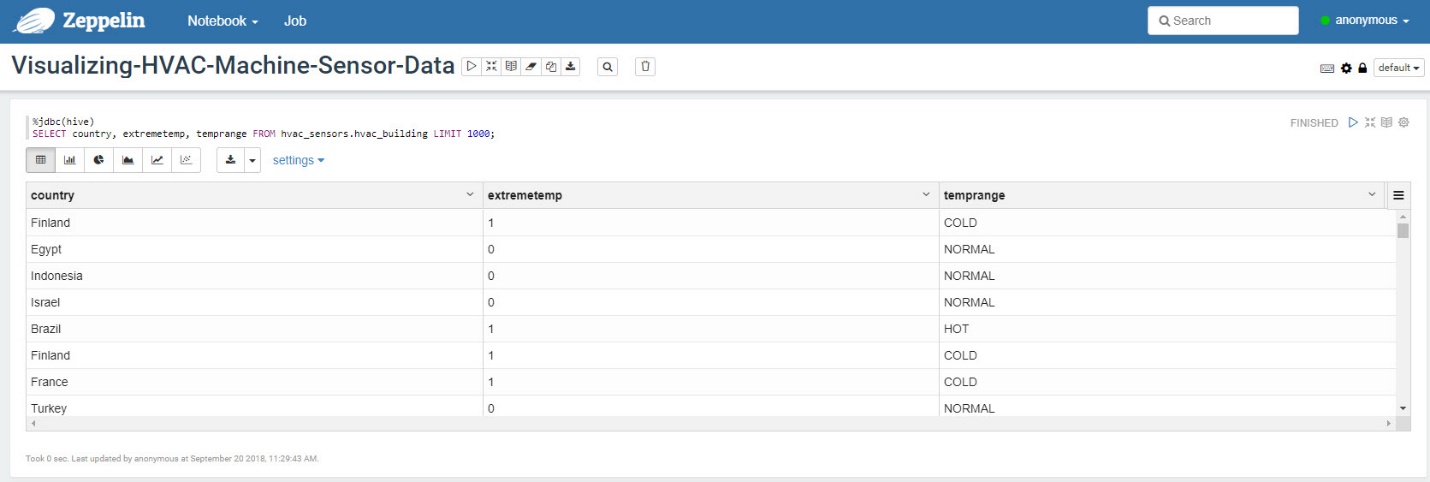
Nhập **Note Name** là Visualizing-HVAC-Machine-Sensor-Data, chọn jdbc làm trình thông dịch mặc định, sau đó nhấp vào **Create**.

Chúng ta sẽ sử dụng trình thông dịch JDBC Hive để chạy các truy vấn Hive và trực quan hóa kết quả.

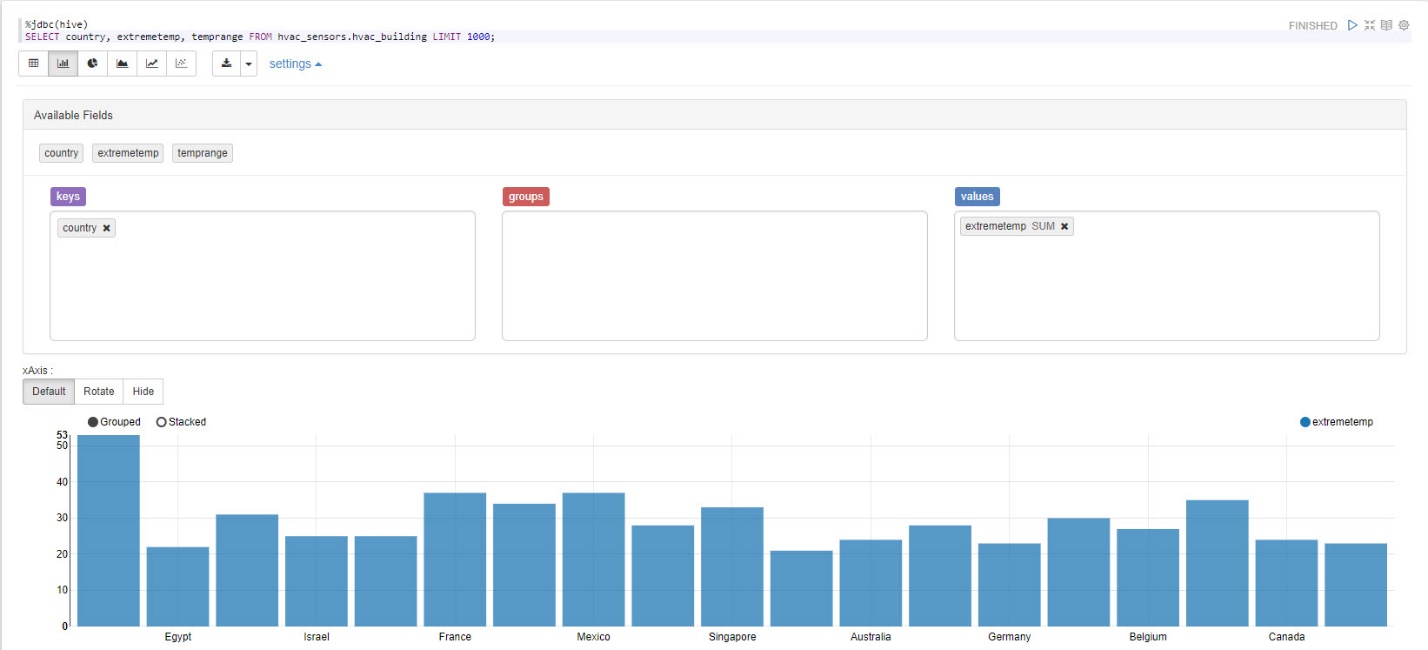
Trước tiên, chúng ta cần chọn các cột từ bảng hvac\_sensors.hvac\_building để minh họa vị trí **quốc gia** của tòa nhà hvac\_building, xem liệu **nhiệt độ** trong tòa nhà hvac\_building có cực đoan hay không và mức **temprange** là gì cho một hvac\_building. Sao chép và dán truy vấn Hive sau đây:

%jdbc(hive)

SELECT country, extremetemp, temprange FROM hvac\_sensors.hvac\_building LIMIT 1000;

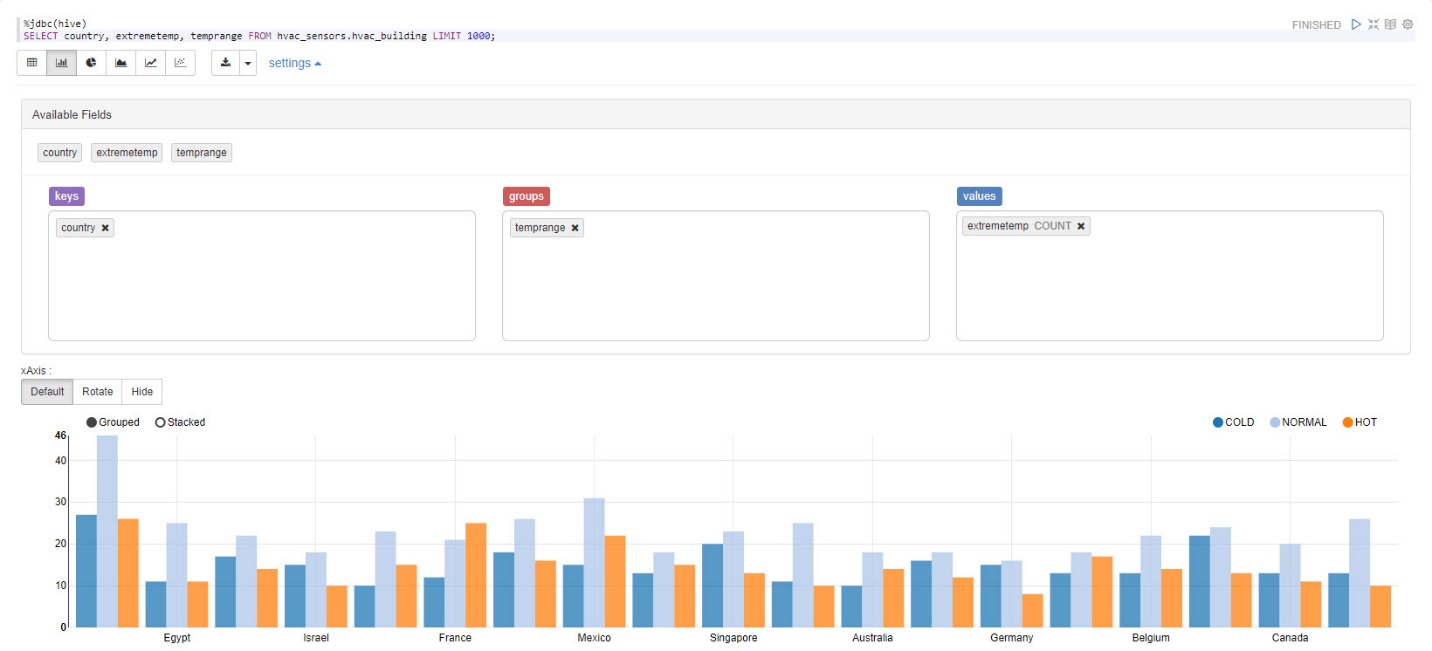


Chọn nút **biểu đồ thanh** nằm ngay dưới truy vấn để chuyển đổi trực quan hóa bảng thành biểu đồ thanh.



Hãy cấu hình biểu đồ này thêm, nhấp vào **settings** để mở các trường khả dụng khác.

* Sắp xếp các trường theo hình ảnh dưới đây.
* Kéo trường temprange vào hộp **groups**.
* Nhấp vào **SUM** trên extremetemp và thay đổi thành **COUNT**.
* Đảm bảo rằng country là duy nhất dưới **Keys**.



Từ biểu đồ trên, chúng ta có thể thấy các quốc gia có các tòa nhà HVAC có nhiệt độ cực đoan nhất được chỉ định bằng đếm sự kiện ***extremetemp*** liên quan đến mức ***temprange*** cho số lượng sự kiện **NORMAL** so với **HOT** và **COLD.**

**Trực quan hóa nhiệt độ cực đoan trong các tòa nhà có Sản phẩm HVAC**

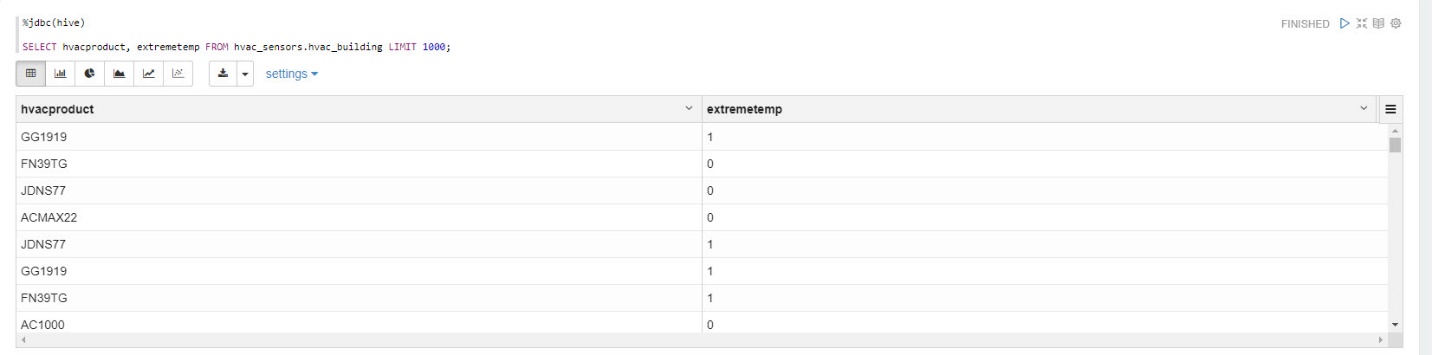
Có thể xác định được các tòa nhà cần nâng cấp HVAC và những tòa nhà không cần không? Hãy xác định câu trả lời đó trong các bước tiếp theo...

Hãy tạo thêm một ghi chú khác để trực quan hóa loại hệ thống HVAC nào dẫn đến số lượng ít nhất các giá trị extremetemp.

Sao chép và dán truy vấn Hive vào ghi chú Zeppelin tiếp theo:

%jdbc(hive)

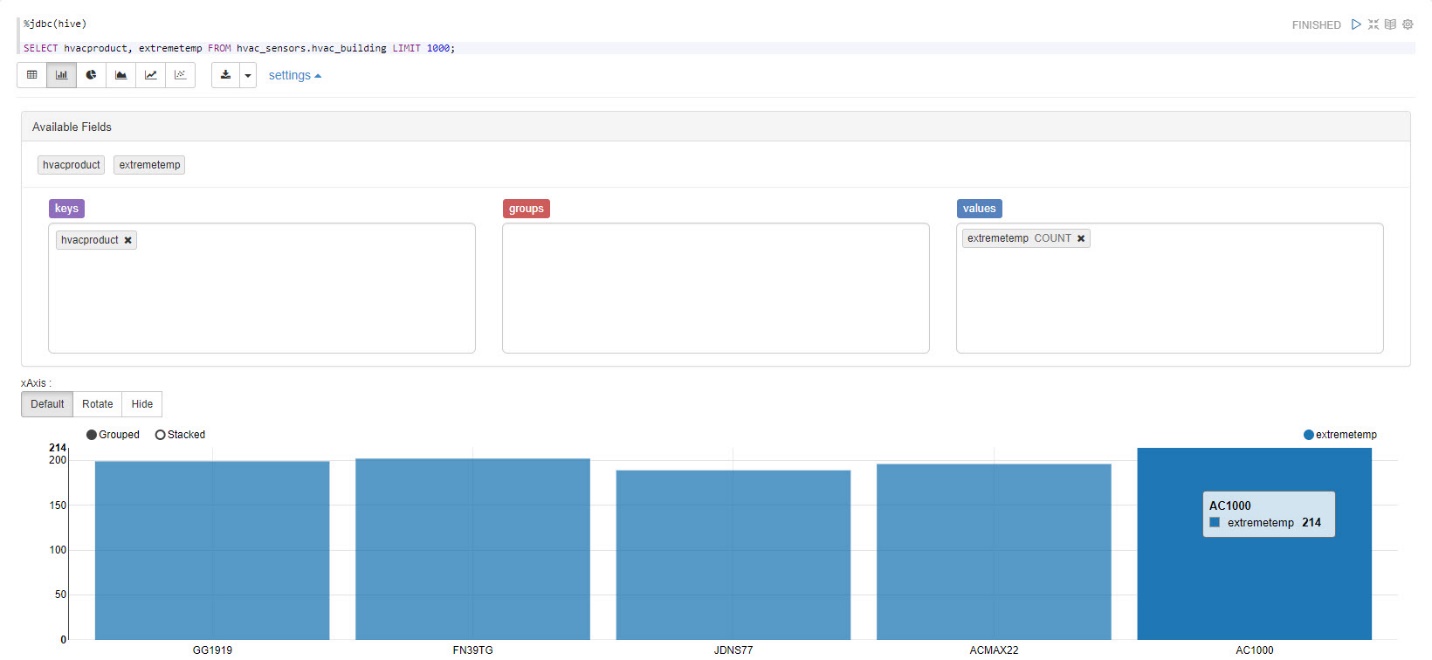
select hvacproduct, extremetemp from hvac\_sensors.hvac\_building;



Sắp xếp các trường theo hình ảnh dưới đây để chúng ta có thể tái tạo biểu đồ dưới đây.

Bây giờ chọn biểu đồ thanh, sau đó sắp xếp các trường cho hvacproduct và extremetemp.

* Đảm bảo rằng hvacproduct nằm trong hộp **Keys**.
* Đảm bảo rằng extremetemp nằm trong hộp **Values** và được đặt thành **COUNT**.



Bây giờ chúng ta có thể thấy những đơn vị HVAC nào dẫn đến số lượng giá trị extremetemp cao nhất. Do đó, chúng ta có thể đưa ra quyết định thông minh hơn khi mua hệ thống HVAC mới.

**Tóm tắt**

Chúng ta đã thành công trong việc tìm hiểu sâu hơn về dữ liệu của chúng ta thông qua việc trực quan hóa các thuộc tính quan trọng bằng cách sử dụng Apache Zeppelin. Chúng ta đã học cách trực quan hóa biểu đồ cho thấy các quốc gia có nhiệt độ cực đoan nhất và lượng sự kiện **NORMAL** so với **HOT** và **COLD**. Chúng ta đã học cách xem các đơn vị HVAC nào dẫn đến số lượng giá trị extremetemp cao nhất.

**Đọc thêm**

* [Cách tạo Trực quan hóa Bản đồ trong Apache Zeppelin](https://community.hortonworks.com/questions/78430/how-to-create-map-visualization-in-apache-zeppelin.html)
* [Sử dụng Angular trong Apache Zeppelin để tạo các trực quan hóa tùy chỉnh](https://community.hortonworks.com/articles/75834/using-angular-within-apache-zeppelin-to-create-cus.html)