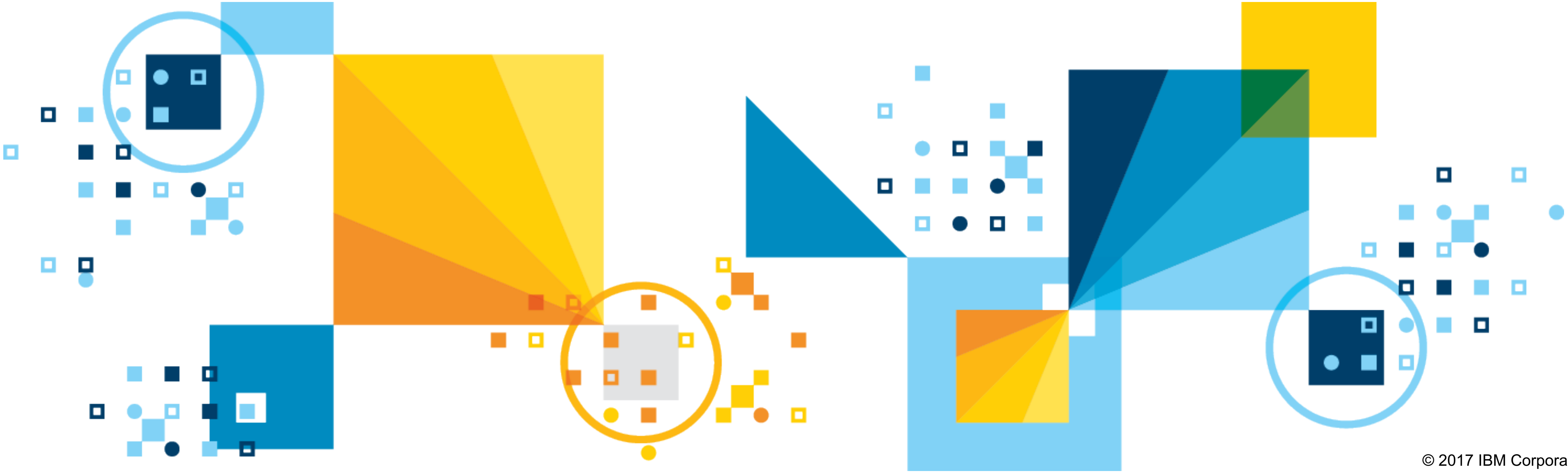
IBM Analytics

Roger E. Sanders – October 20, 2017

IoT development with Db2 and a SBC



Please note

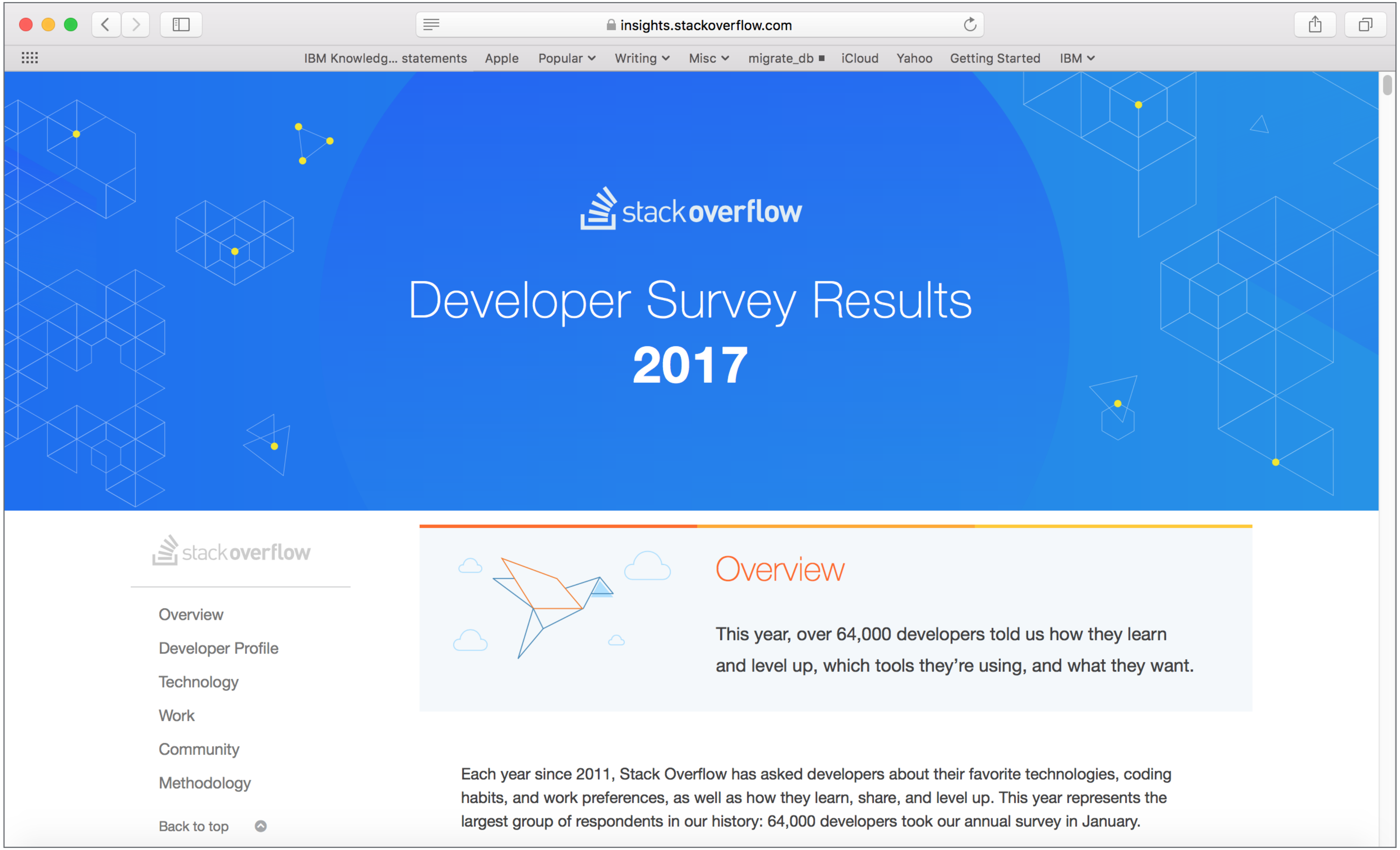
- IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM’s sole discretion.
- Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.
- The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.
- The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.
- Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

The Challenge:

How can we get young developers excited about working with Db2?

Developer profile

- **Developers are developers:** At their core, developers share a common set of attitudes and behaviors:
  - They're curious
  - They pay attention to detail
  - They possess clear thinking skills
  - They are passionate
  - They're quick learners
  - They are technology "geeks"
- **Developers have a quest for knowledge:** Developers embrace learning opportunities – they seek out ways to bolster their skill set and experiment with new technologies. ***According to a 2017 survey of over 30,00 developers, 90% are self-taught!***
- **Developers have power:** Once viewed as influencers only, developers have more buying decision power today than they did in the past.
- **There is a finite number of developers in the world:** There are approximately 21 million developers in the world; there is a lot of competition for their mindshare.



Developer Survey Results

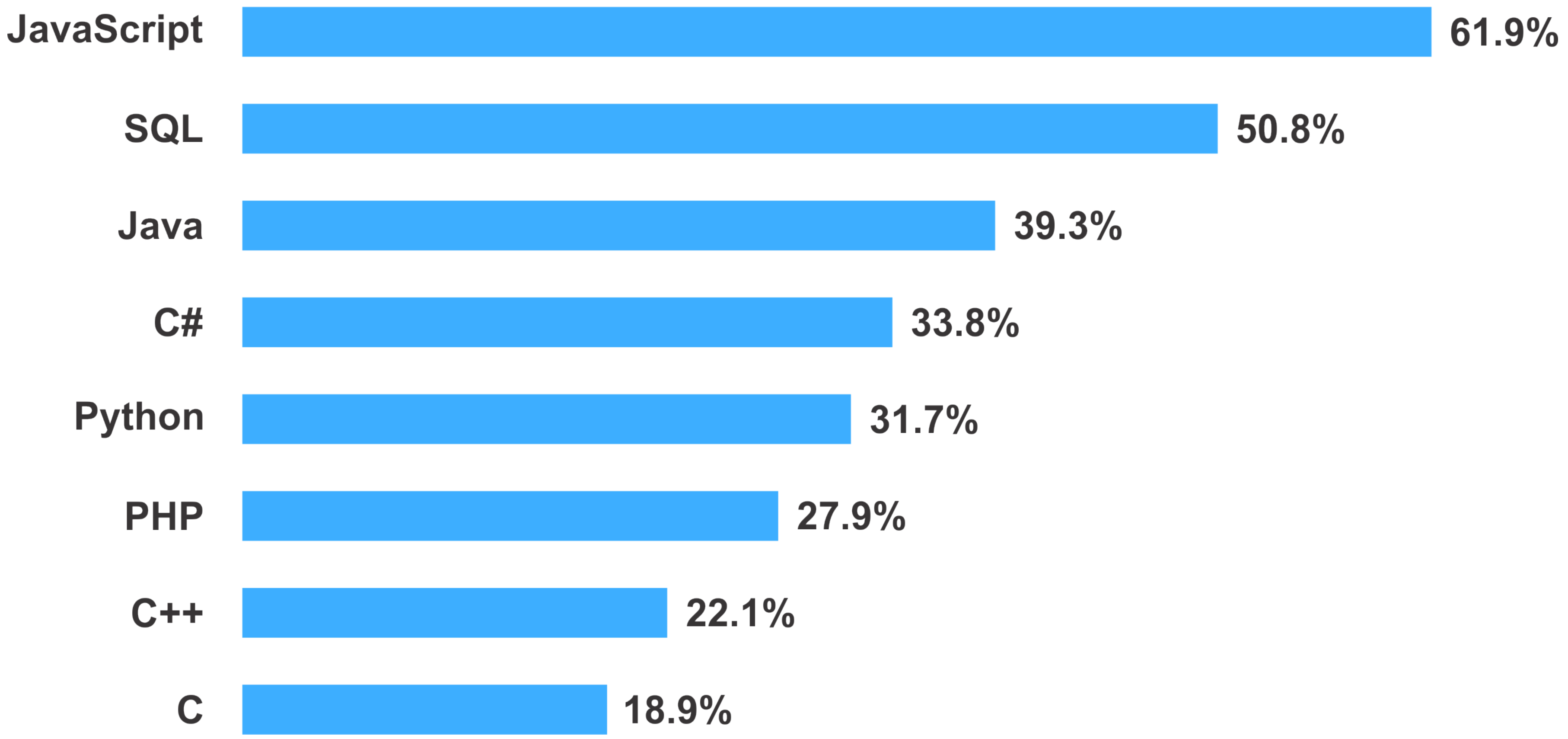
2017

Overview

This year, over 64,000 developers told us how they learn and level up, which tools they're using, and what they want.

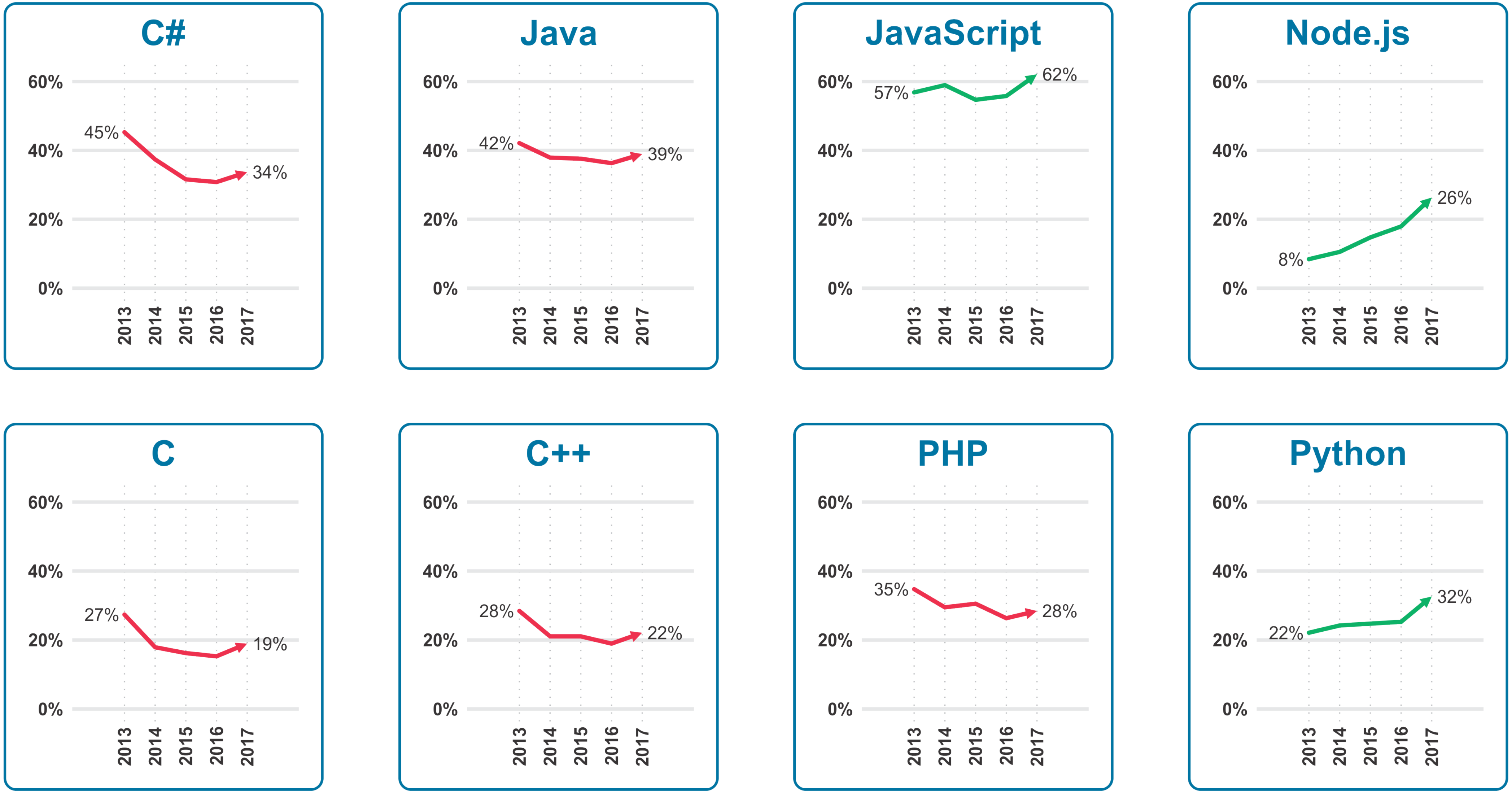
Each year since 2011, Stack Overflow has asked developers about their favorite technologies, coding habits, and work preferences, as well as how they learn, share, and level up. This year represents the largest group of respondents in our history: 64,000 developers took our annual survey in January.

Programming languages developers use

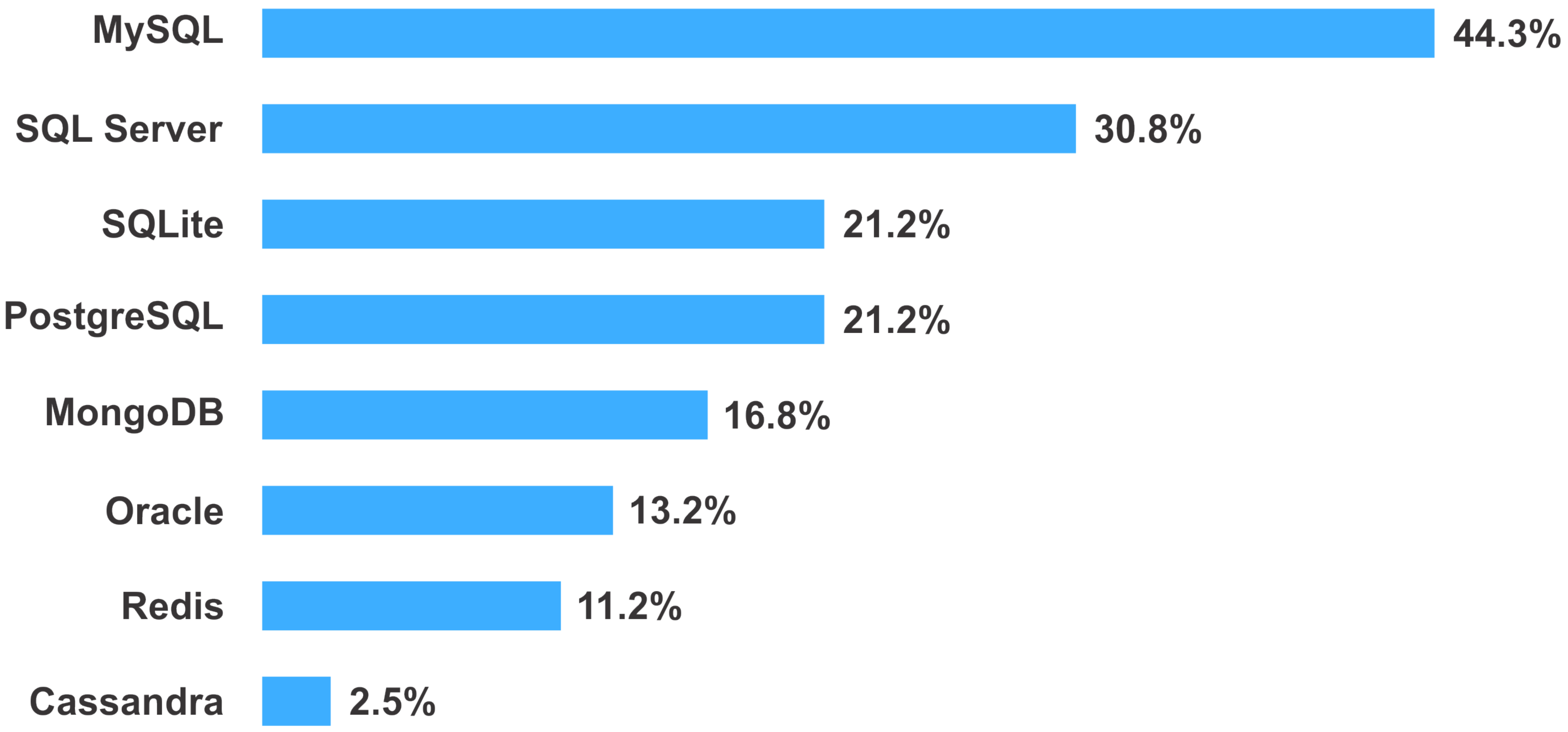


Source: Stack Overflow Developer Survey 2017
***36,935** responses; "Select all that apply." Shown as a percentage of the respondants who chose at least one language.*

Change in language popularity over time

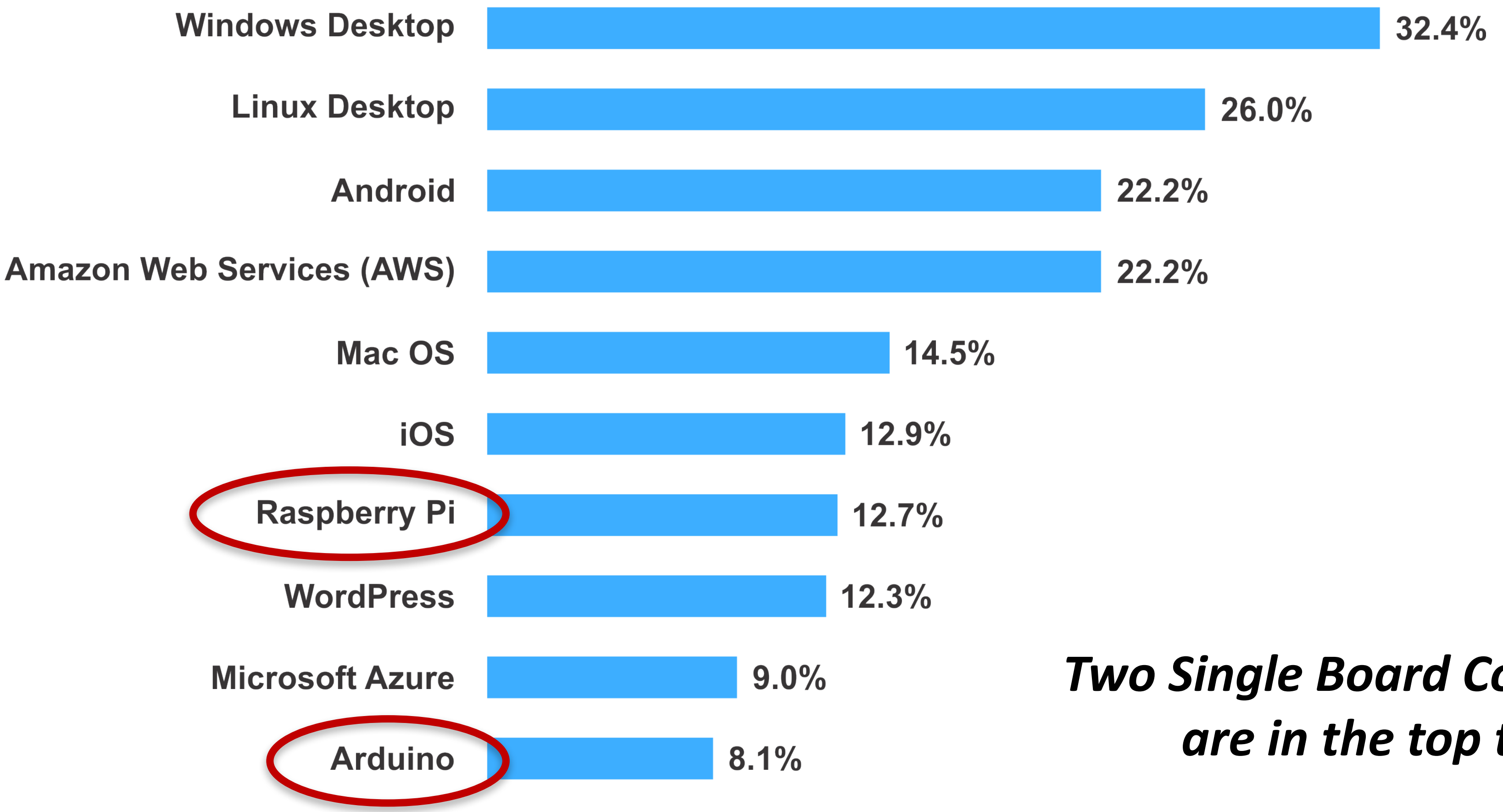


Databases developers use



Source: Stack Overflow Developer Survey 2017
***36,935** responses; "Select all that apply." Shown as a percentage of the respondants who chose at least one database.*

Platforms developers use



Two Single Board Computers are in the top ten!

Source: Stack Overflow Developer Survey 2017
***36,935** responses; "Select all that apply." Shown as a percentage of the respondants who chose at least one platform.*

What is a Single Board Computer (SBC)?

As the name implies, a **single-board computer** (**SBC**) is a computer that is built on a single circuit board. SBCs have many of the components found in desktop and laptop computers – microprocessors (CPUs), memory, storage, and input/output (I/O) ports – but, they typically do not offer expansion slots.

SBCs are used for a variety of things:

- for educational or demonstration purposes
- as embedded computer controllers
- as the heart of countless DIY projects – from home automation to robotics and just about anything in between

"Want to have a coffee pot tweet when the coffee is ready? Arduino."
Limor Fried, "*Geek 101: What Is Arduino?,*" PCWorld, 09/08/2011

Are we talking about Internet of Things (IoT)?

At its most basic definition, the **Internet of Things** (**IoT**) refers to a system of interrelated computing devices and "things" that have been given unique identifiers so they can communicate with each other to automate processes or produce large amounts of data that can be harnessed by control systems or analyzed for patterns.

The "thing" could be a person with a insulin pump implant, a farm animal with a biochip transponder, an automobile with built-in sensors that indicate when tire pressure is low – or any other natural or man-made object that can be assigned an IP address and given the ability to transfer data over a network.

So, yes – many DIY projects that use a SBC are, at their core, IoT projects!

(Don't believe me? Look at some of the projects found at www.hackster.io with the definition of IoT in mind.)

The Question:

Can a Db2 database be built on a SBC *and* can a popular language be used to collect data from sensors and store it in the database *and* can the database exploit features that are not available with other database offerings?

The Answer:

YES!!!

Why an SBC? Because they are quickly becoming part of school curriculums

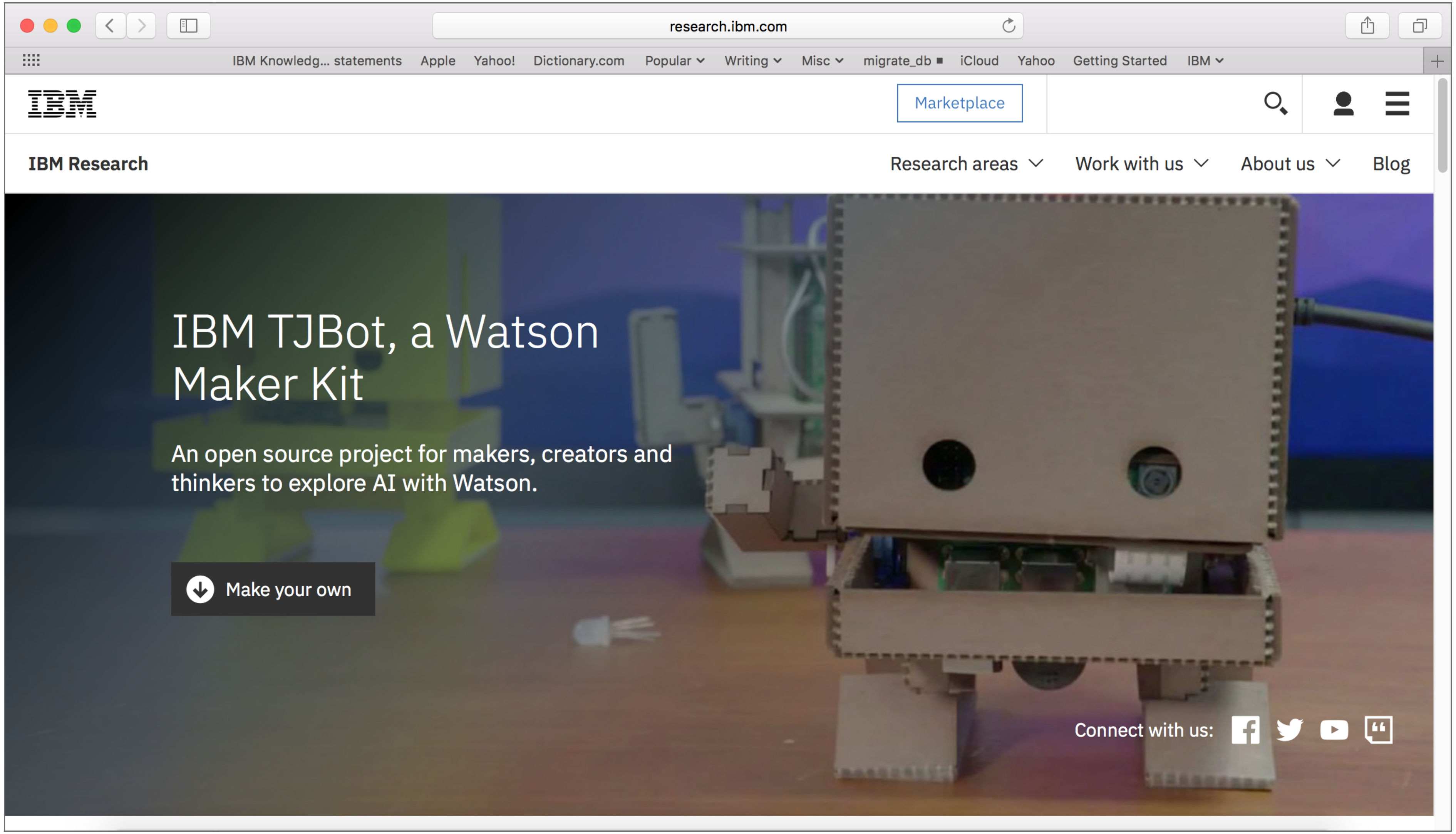
“There’s a new addition to the class, and it’s changing computer science education everywhere. Single board computers (SBC), originally intended for engineers, are now appearing in university, secondary, and even elementary school curriculums around the world. Students of all ages and experience levels are getting their hands on technologies that those same engineers rely on to run tests, develop prototypes, and bring their designs into production.”

Ken Kaminski,
beanz: The Magazine for Kids, Code, and Computer Science
August 2015 Issue: Back to School
https://www.kidscodecs.com/single-board-computers/

beanz

The Magazine for
Kids, Code, and
Computer Science

And, because IBM already has an SBC-based project kit for Watson



IBM TJBot, a Watson Maker Kit

An open source project for makers, creators and thinkers to explore AI with Watson.

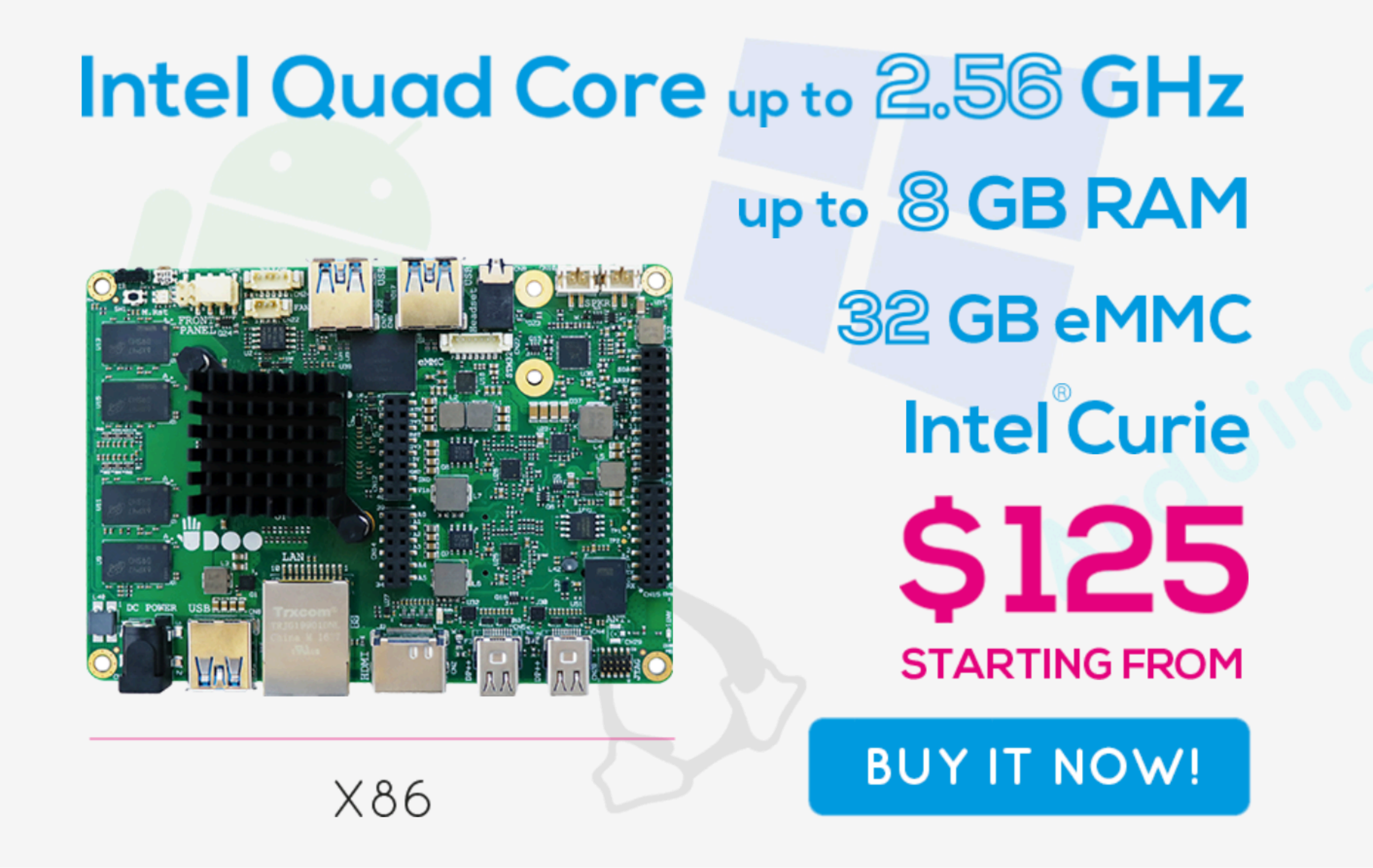
Another advantage of using SBCs – a wide variety of sensors are available

Which SBCs are users turning to?

Single Board Computer (SBC) Model	Number of 1st Choice Votes	MSRP
Raspberry Pi 3 Model B	624	$35.00
UDOO X86	86	$125.00
ODROID-XU4	78	$59.95
BeagleBone Black Rev C	75	$62.95
Raspberry Pi Zero W	73	$39.99
Raspberry Pi 2 Model B	67	$38.49
ODROID-C2	54	$55.75
DragonBoard 410C	53	$84.99
UP Squared	36	$100.68
Raspberry Pi Zero	34	$39.99

Source: Hacker Board Survey 2017 - LinuxGizmos.com

What is the UDOO X86?



Intel Quad Core up to 2.56 GHz

up to 8 GB RAM

32 GB eMMC

Intel® Curie

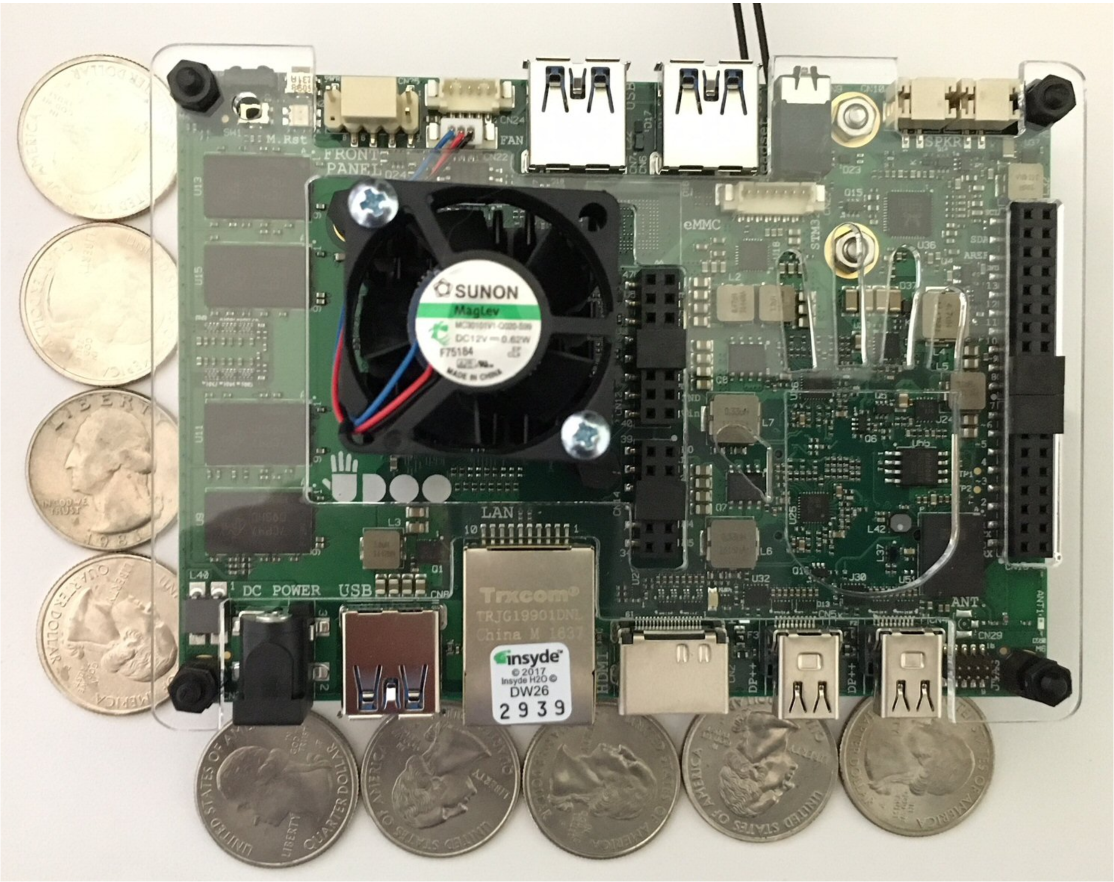
$125

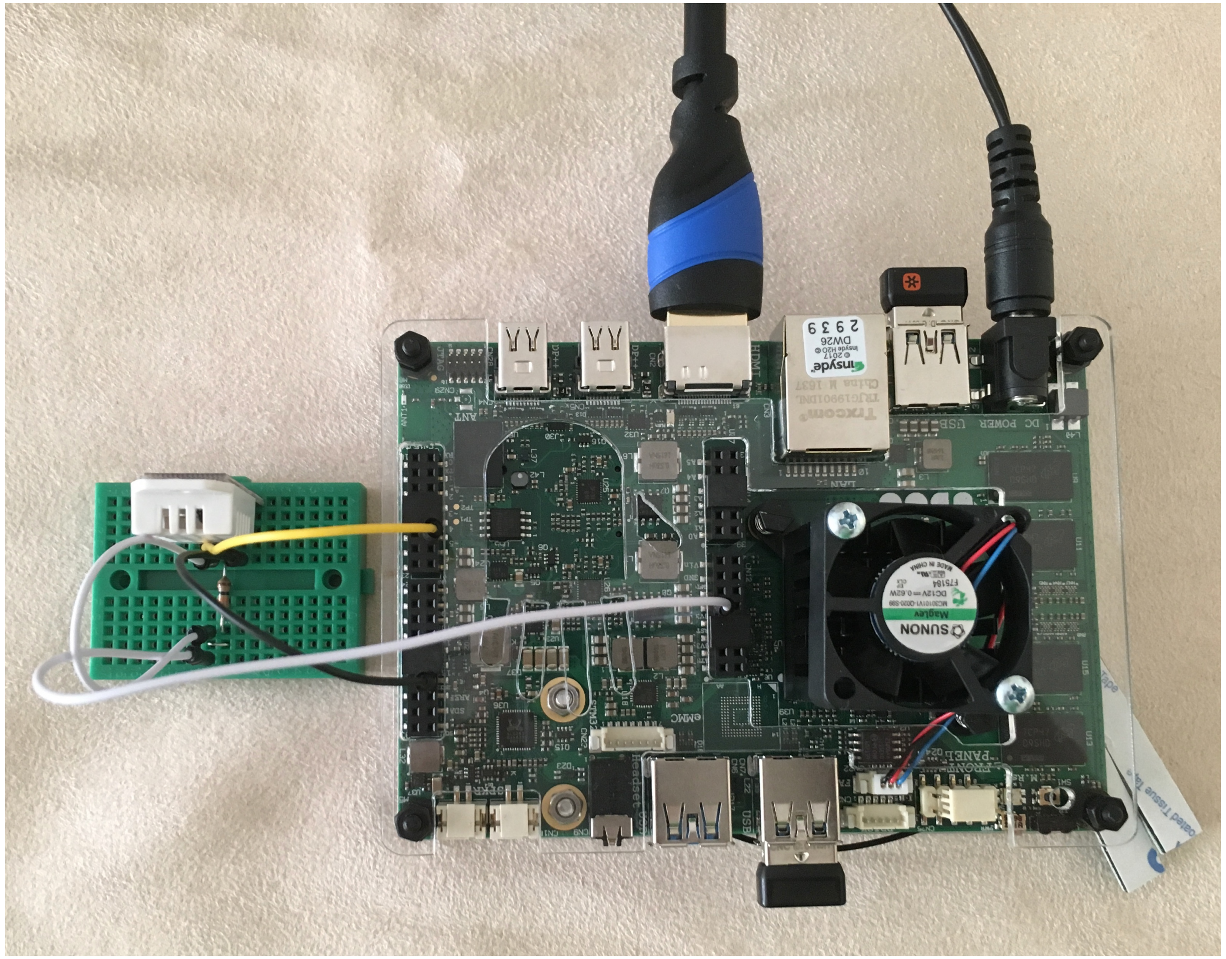
STARTING FROM

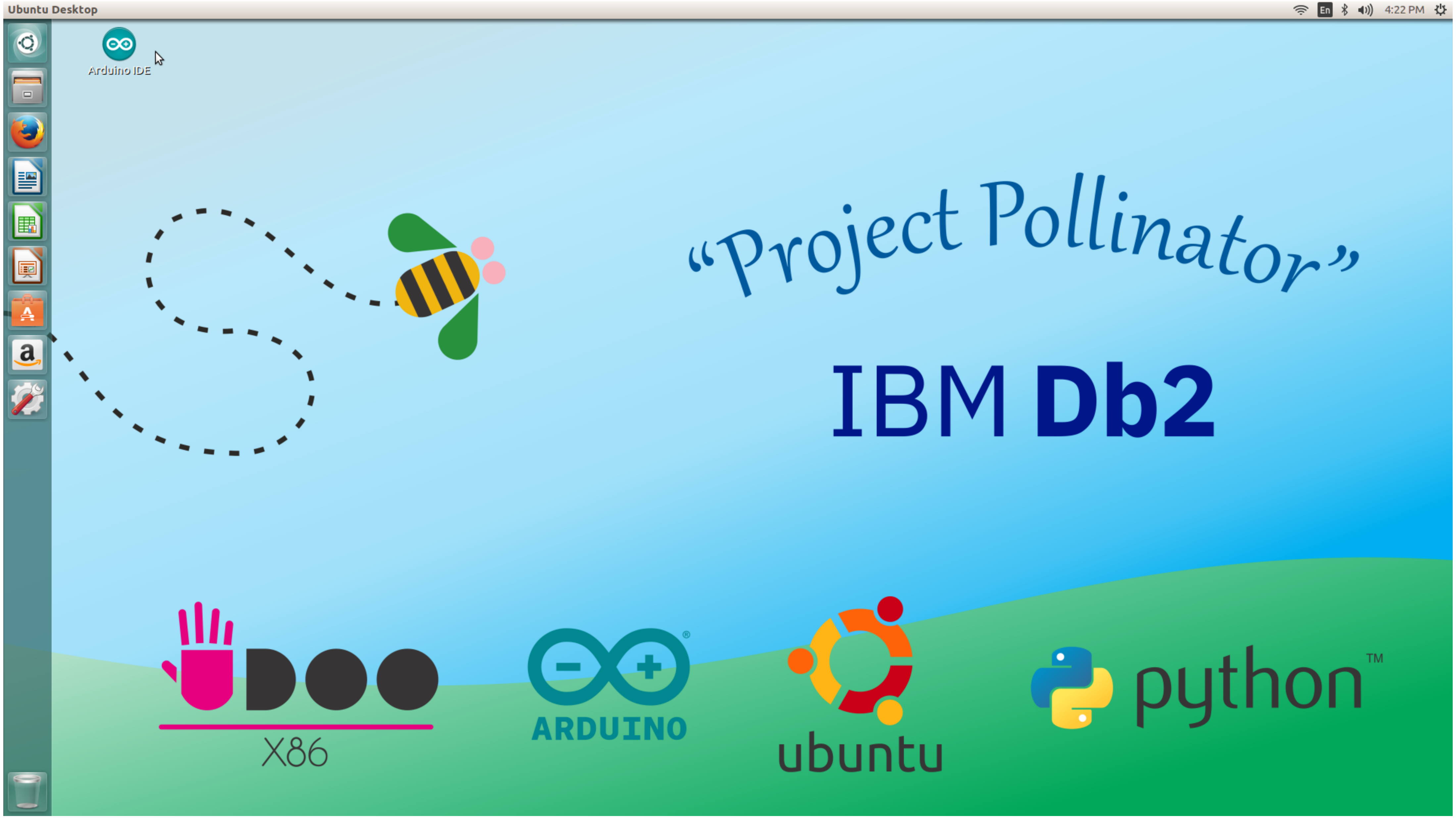
BUY IT NOW!

X86

A UDOO X86 Advanced SBC – by itself and with a temperature/humidity sensor attached (via the built-in Arduino)

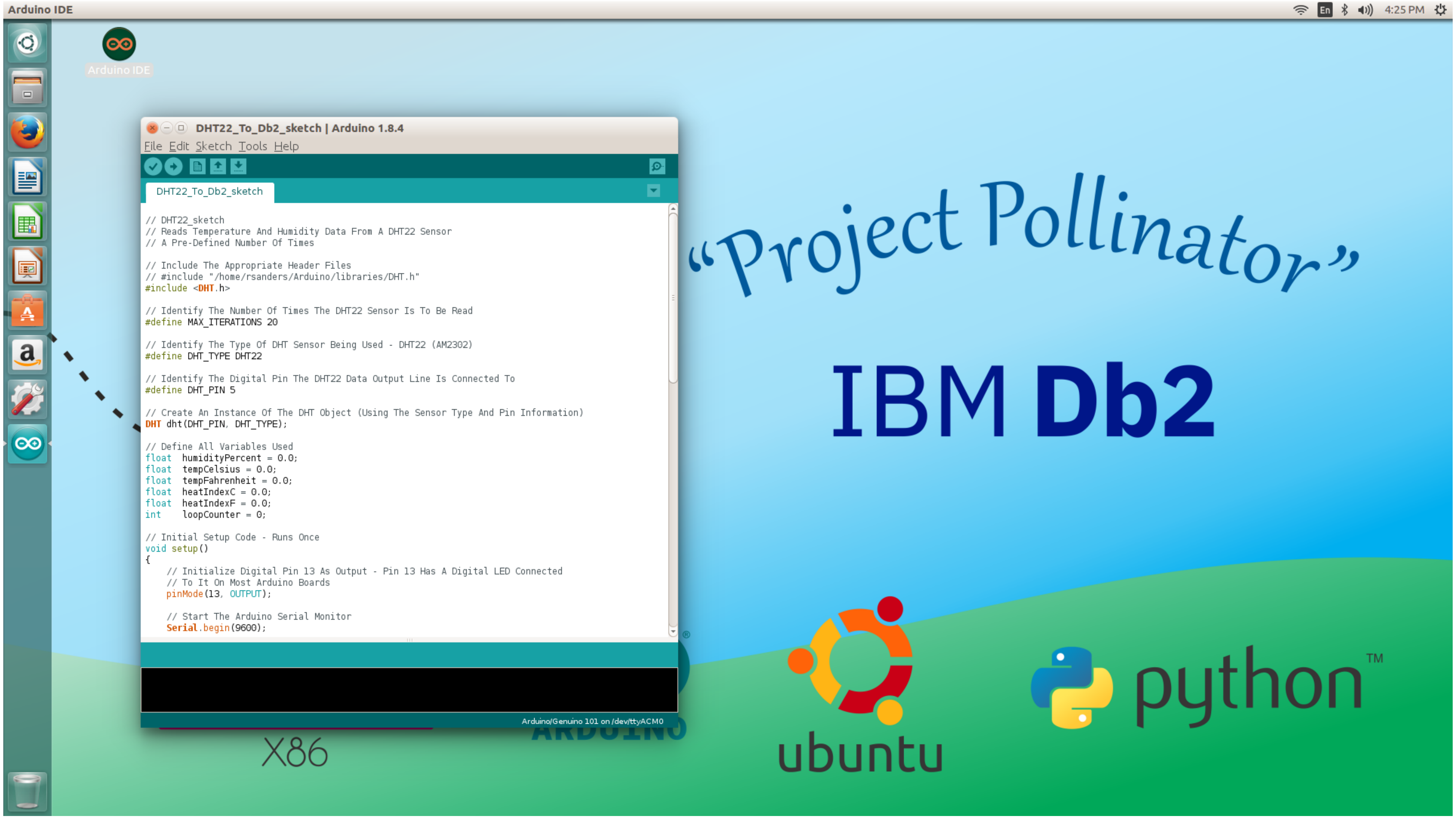




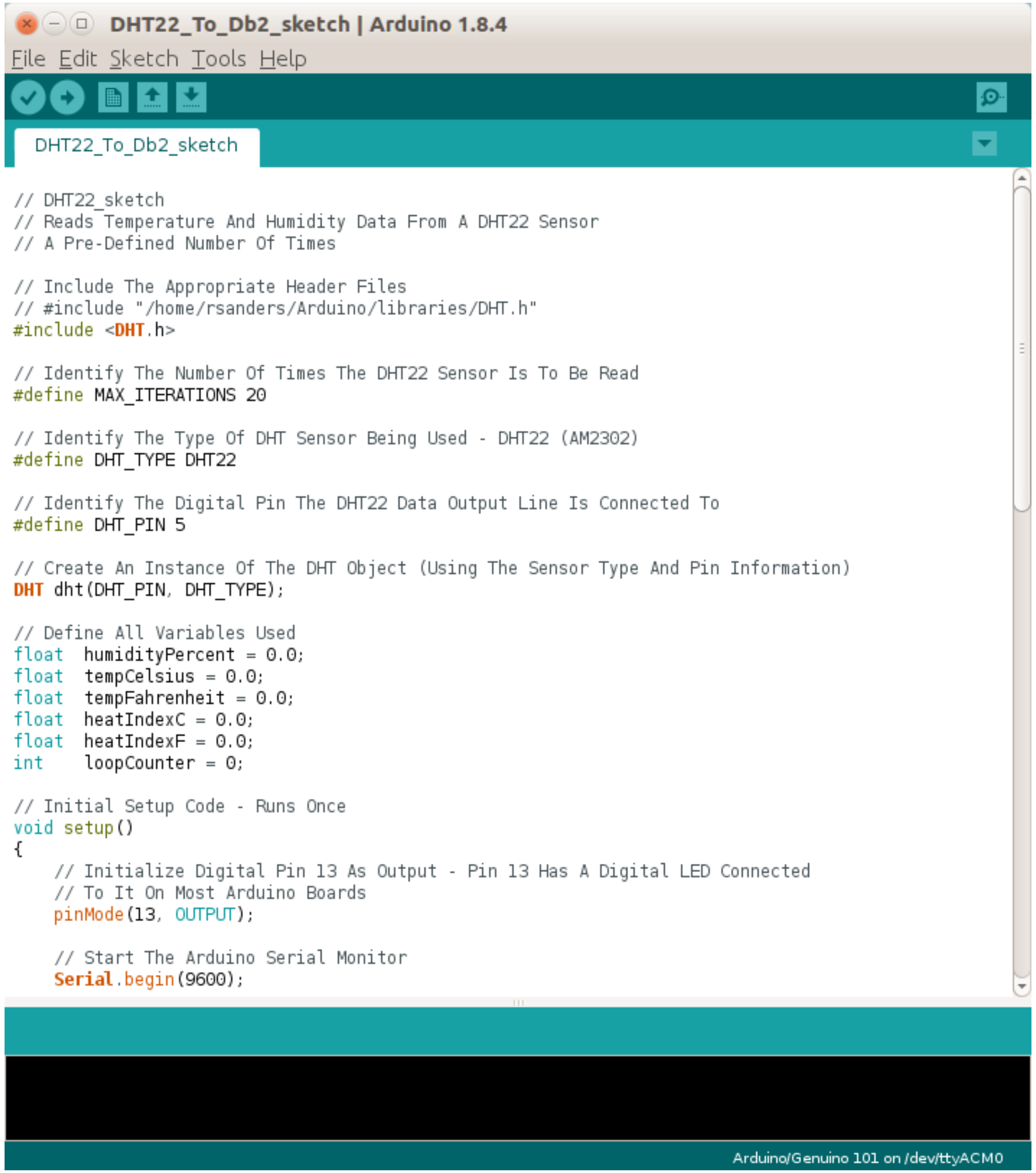

Ubuntu Desktop
En
4:22 PM
Arduino IDE
"Project Pollinator"
IBM Db2
UDOO
X86
ARDUINO
ubuntu
python™

4:24 PM
Arduino IDE
"Project Pollinator"
IBM Db2
ARDUINO
Genuino
AN OPEN PROJECT WRITTEN, DEBUGGED, AND SUPPORTED BY ARDUINO.CC AND THE ARDUINO COMMUNITY WORLDWIDE
LEARN MORE ABOUT THE CONTRIBUTORS OF ARDUINO.CC on arduino.cc/credits
X86
ARDUINO
ubuntu
python™

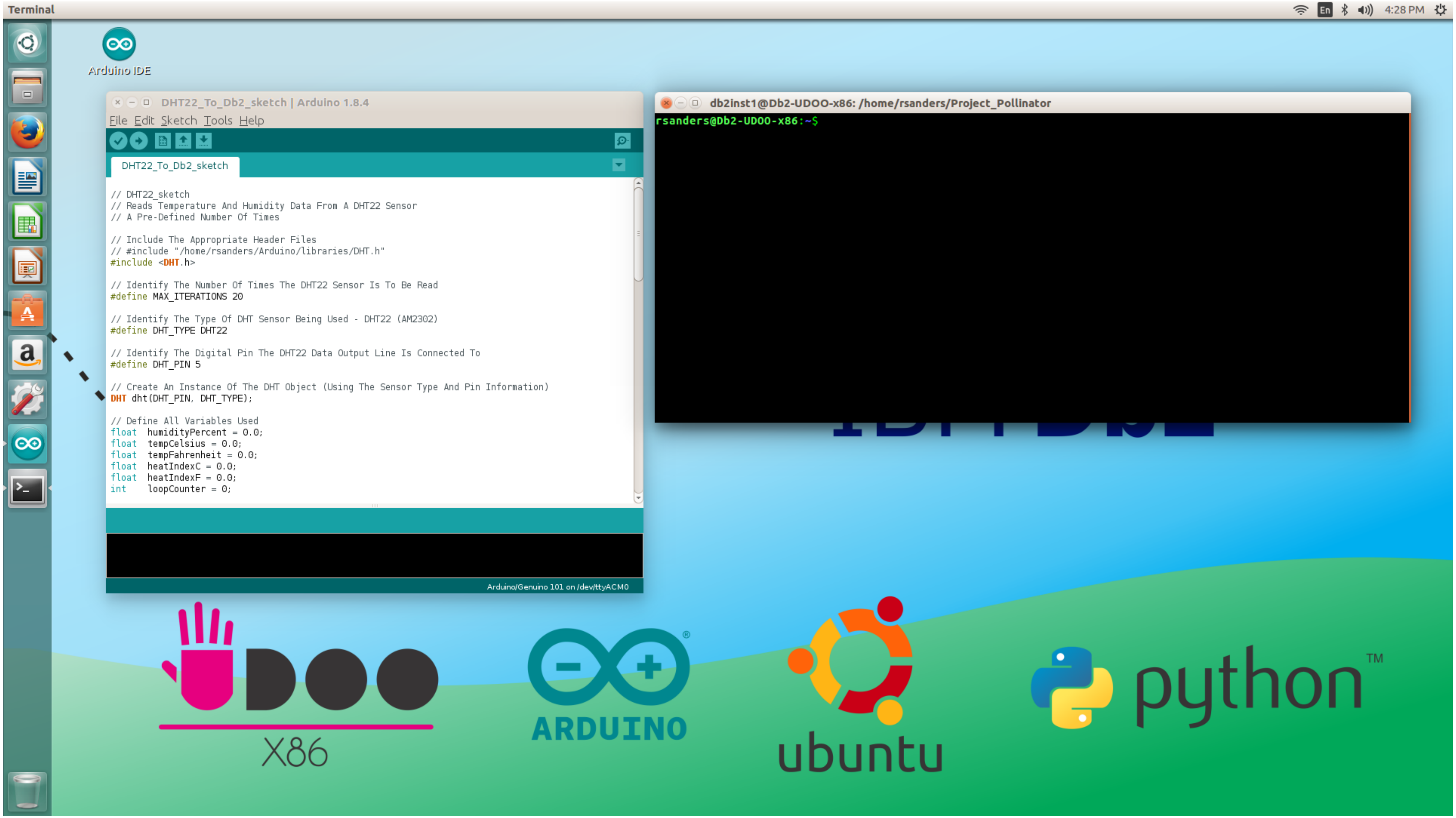
Initializing packages...



```
// DHT22_sketch
// Reads Temperature And Humidity Data From A DHT22 Sensor
// A Pre-Defined Number Of Times

// Include The Appropriate Header Files
// #include "/home/rsanders/Arduino/libraries/DHT.h"
#include <DHT.h>

// Identify The Number Of Times The DHT22 Sensor Is To Be Read
#define MAX_ITERATIONS 20

// Identify The Type Of DHT Sensor Being Used - DHT22 (AM2302)
#define DHT_TYPE DHT22

// Identify The Digital Pin The DHT22 Data Output Line Is Connected To
#define DHT_PIN 5

// Create An Instance Of The DHT Object (Using The Sensor Type And Pin Information)
DHT dht(DHT_PIN, DHT_TYPE);

// Define All Variables Used
float  humidityPercent = 0.0;
float  tempCelsius = 0.0;
float  tempFahrenheit = 0.0;
float  heatIndexC = 0.0;
float  heatIndexF = 0.0;
int    loopCounter = 0;

// Initial Setup Code - Runs Once
void setup()
{
    // Initialize Digital Pin 13 As Output - Pin 13 Has A Digital LED Connected
    // To It On Most Arduino Boards
    pinMode(13, OUTPUT);

    // Start The Arduino Serial Monitor
    Serial.begin(9600);
```


DHT22_To_Db2_sketch | Arduino 1.8.4
File Edit Sketch Tools Help
DHT22_To_Db2_sketch
Arduino/Genuino 101 on /dev/ttyACM0

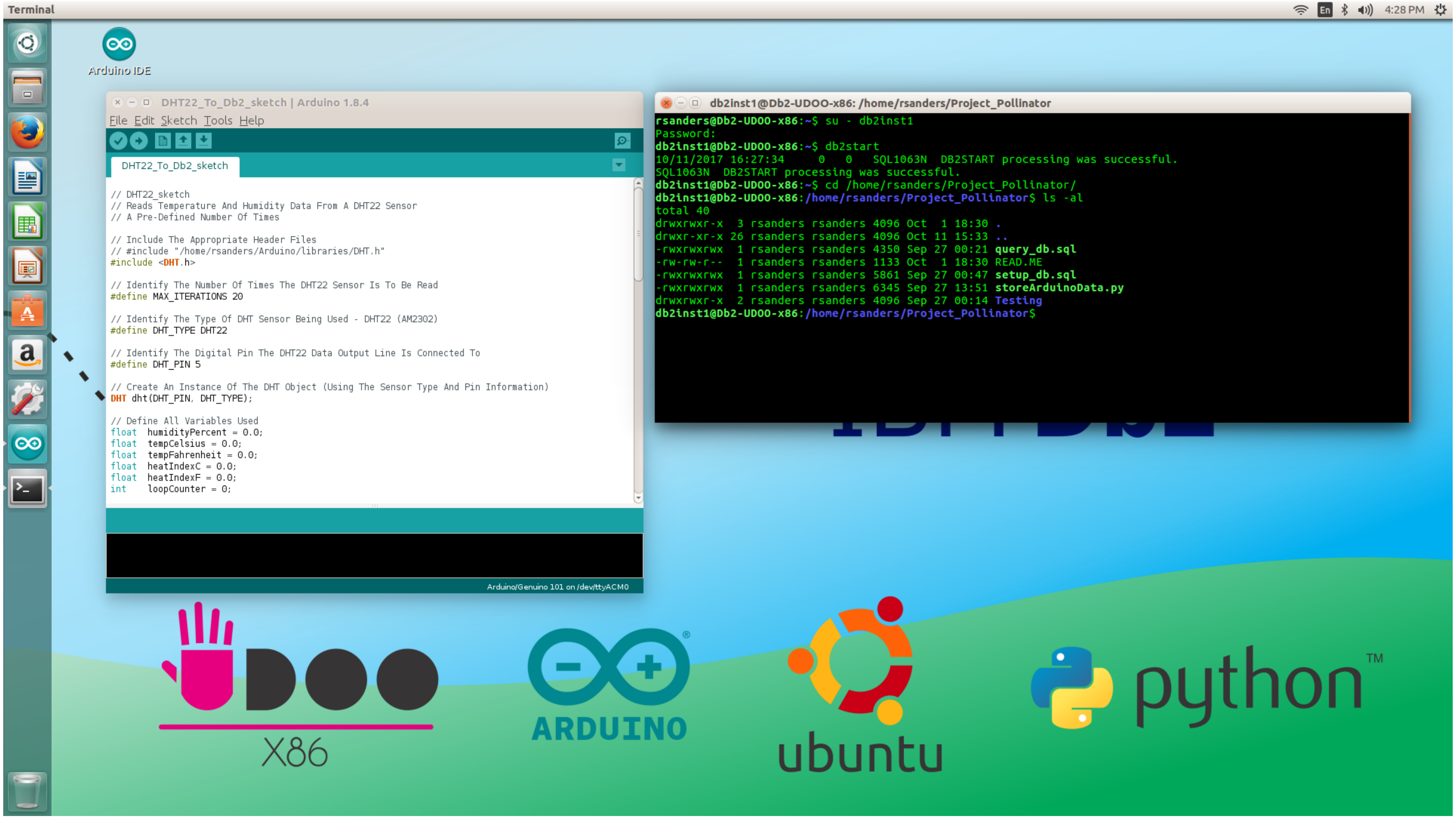
```
// DHT22_sketch
// Reads Temperature And Humidity Data From A DHT22 Sensor
// A Pre-Defined Number Of Times

// Include The Appropriate Header Files
// #include "/home/rsanders/Arduino/libraries/DHT.h"
#include <DHT.h>

// Identify The Number Of Times The DHT22 Sensor Is To Be Read
#define MAX_ITERATIONS 20

// Identify The Type Of DHT Sensor Being Used - DHT22 (AM2302)
#define DHT_TYPE DHT22

// Identify The Digital Pin The DHT22 Data Output Line Is Connected To
#define DHT_PIN 5

// Create An Instance Of The DHT Object (Using The Sensor Type And Pin Information)
DHT dht(DHT_PIN, DHT_TYPE);

// Define All Variables Used
float  humidityPercent = 0.0;
float  tempCelsius = 0.0;
float  tempFahrenheit = 0.0;
float  heatIndexC = 0.0;
float  heatIndexF = 0.0;
int    loopCounter = 0;

// Initial Setup Code - Runs Once
void setup()
{
    // Initialize Digital Pin 13 As Output - Pin 13 Has A Digital LED Connected
    // To It On Most Arduino Boards
    pinMode(13, OUTPUT);

    // Start The Arduino Serial Monitor
    Serial.begin(9600);
```



Terminal
4:28 PM
Arduino IDE
DHT22_To_Db2_sketch | Arduino 1.8.4
File Edit Sketch Tools Help
DHT22_To_Db2_sketch
Arduino/Genuino 101 on /dev/ttyACM0
db2inst1@Db2-UDOO-x86: /home/rsanders/Project_Pollinator
X86
ARDUINO
ubuntu
python

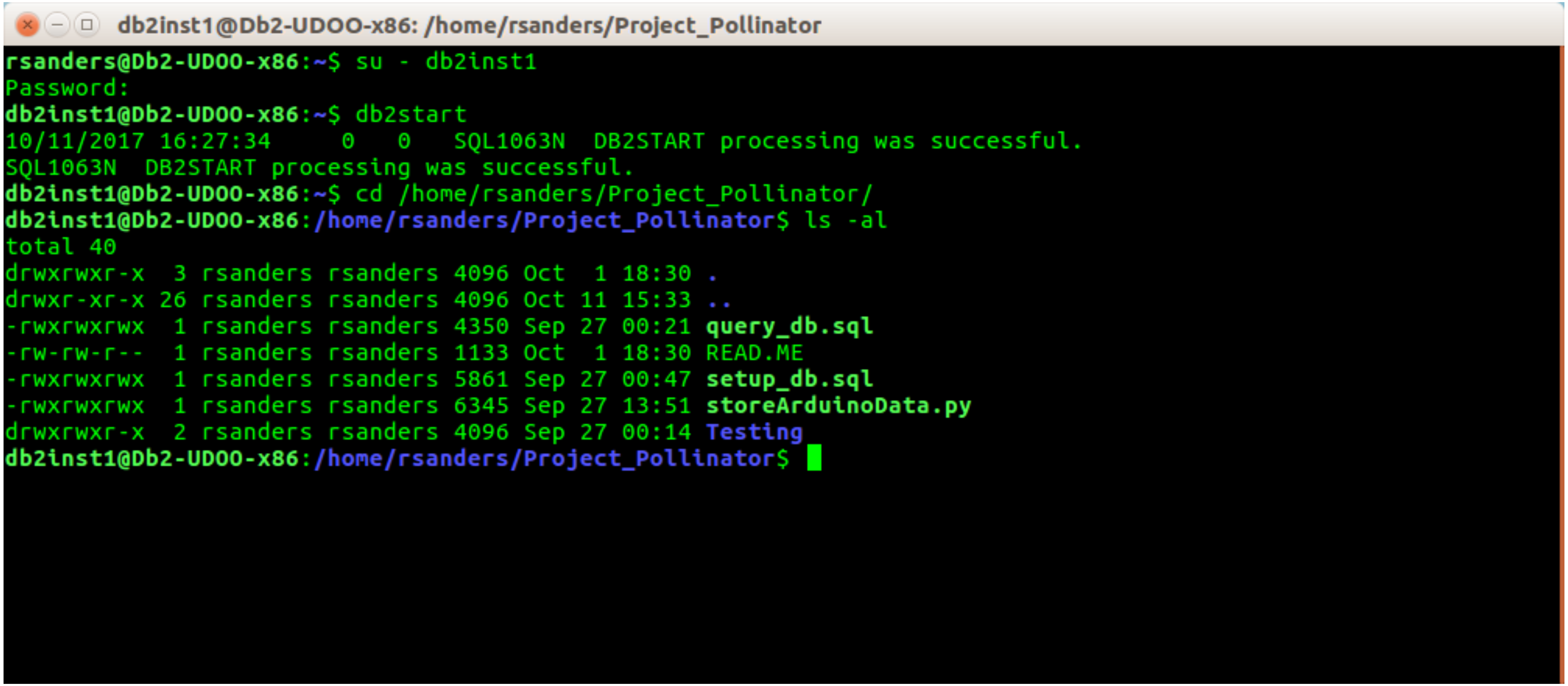
```
// DHT22_sketch
// Reads Temperature And Humidity Data From A DHT22 Sensor
// A Pre-Defined Number Of Times

// Include The Appropriate Header Files
// #include "/home/rsanders/Arduino/libraries/DHT.h"
#include <DHT.h>

// Identify The Number Of Times The DHT22 Sensor Is To Be Read
#define MAX_ITERATIONS 20

// Identify The Type Of DHT Sensor Being Used - DHT22 (AM2302)
#define DHT_TYPE DHT22

// Identify The Digital Pin The DHT22 Data Output Line Is Connected To
#define DHT_PIN 5

// Create An Instance Of The DHT Object (Using The Sensor Type And Pin Information)
DHT dht(DHT_PIN, DHT_TYPE);

// Define All Variables Used
float  humidityPercent = 0.0;
float  tempCelsius = 0.0;
float  tempFahrenheit = 0.0;
float  heatIndexC = 0.0;
float  heatIndexF = 0.0;
int    loopCounter = 0;
```

```
rsanders@Db2-UDOO-x86:~$
```


Terminal
En
4:28 PM
Arduino IDE
DHT22_To_Db2_sketch | Arduino 1.8.4
File Edit Sketch Tools Help
DHT22_To_Db2_sketch
Arduino/Genuino 101 on /dev/ttyACM0
db2inst1@Db2-UDOO-x86: /home/rsanders/Project_Pollinator
UDOO
X86
ARDUINO
ubuntu
python™

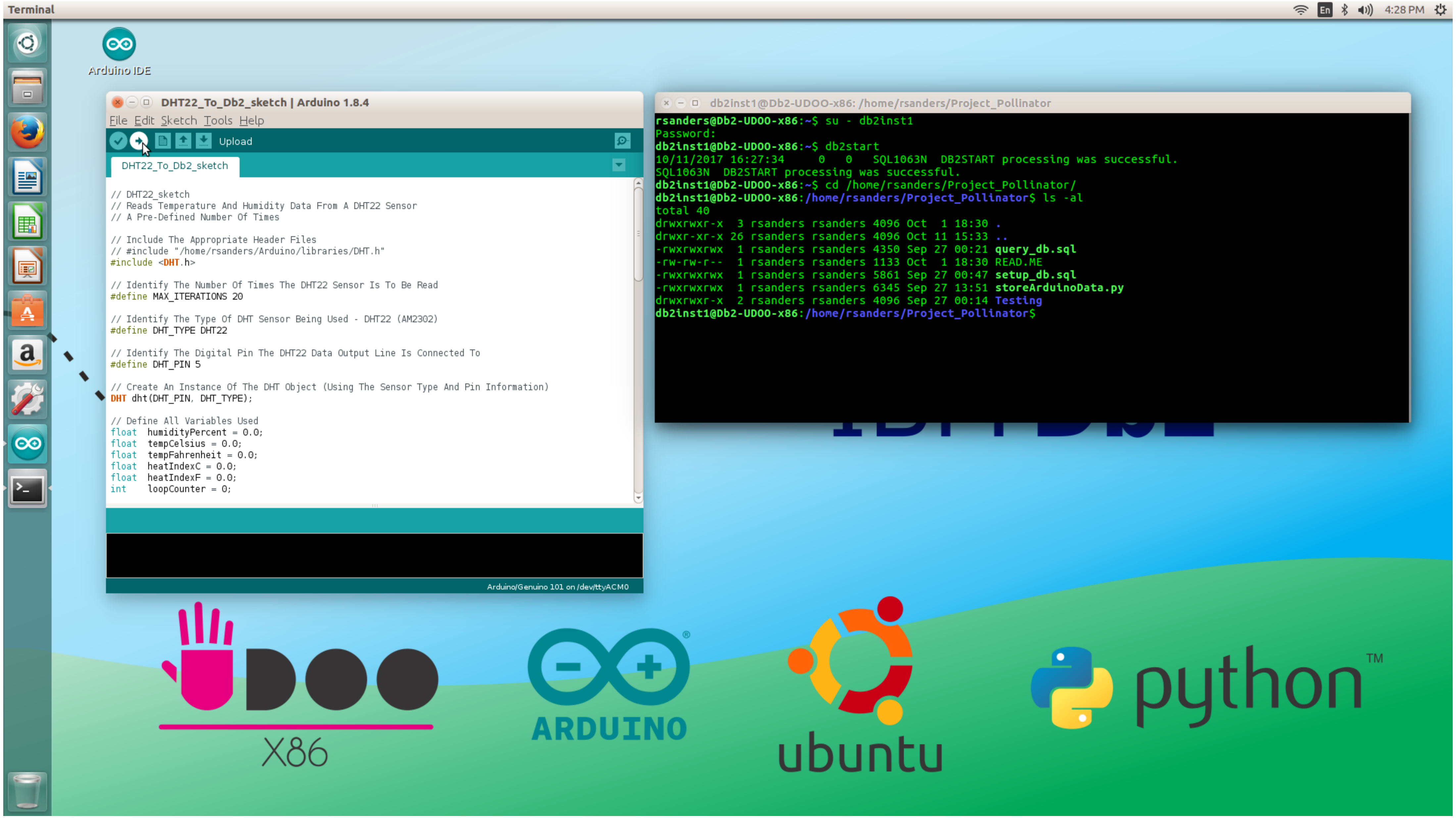
```
// DHT22_sketch
// Reads Temperature And Humidity Data From A DHT22 Sensor
// A Pre-Defined Number Of Times

// Include The Appropriate Header Files
// #include "/home/rsanders/Arduino/libraries/DHT.h"
#include <DHT.h>

// Identify The Number Of Times The DHT22 Sensor Is To Be Read
#define MAX_ITERATIONS 20

// Identify The Type Of DHT Sensor Being Used - DHT22 (AM2302)
#define DHT_TYPE DHT22

// Identify The Digital Pin The DHT22 Data Output Line Is Connected To
#define DHT_PIN 5

// Create An Instance Of The DHT Object (Using The Sensor Type And Pin Information)
DHT dht(DHT_PIN, DHT_TYPE);

// Define All Variables Used
float  humidityPercent = 0.0;
float  tempCelsius = 0.0;
float  tempFahrenheit = 0.0;
float  heatIndexC = 0.0;
float  heatIndexF = 0.0;
int    loopCounter = 0;
```

```
rsanders@Db2-UDOO-x86:~$ su - db2inst1
Password:
db2inst1@Db2-UDOO-x86:~$ db2start
10/11/2017 16:27:34     0   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.
db2inst1@Db2-UDOO-x86:~$ cd /home/rsanders/Project_Pollinator/
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$ ls -al
total 40
drwxrwxr-x  3 rsanders rsanders 4096 Oct  1 18:30 .
drwxr-xr-x 26 rsanders rsanders 4096 Oct 11 15:33 ..
-rwxrwxrwx  1 rsanders rsanders 4350 Sep 27 00:21 query_db.sql
-rw-rw-r--  1 rsanders rsanders 1133 Oct  1 18:30 READ.ME
-rwxrwxrwx  1 rsanders rsanders 5861 Sep 27 00:47 setup_db.sql
-rwxrwxrwx  1 rsanders rsanders 6345 Sep 27 13:51 storeArduinoData.py
drwxrwxr-x  2 rsanders rsanders 4096 Sep 27 00:14 Testing
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$
```



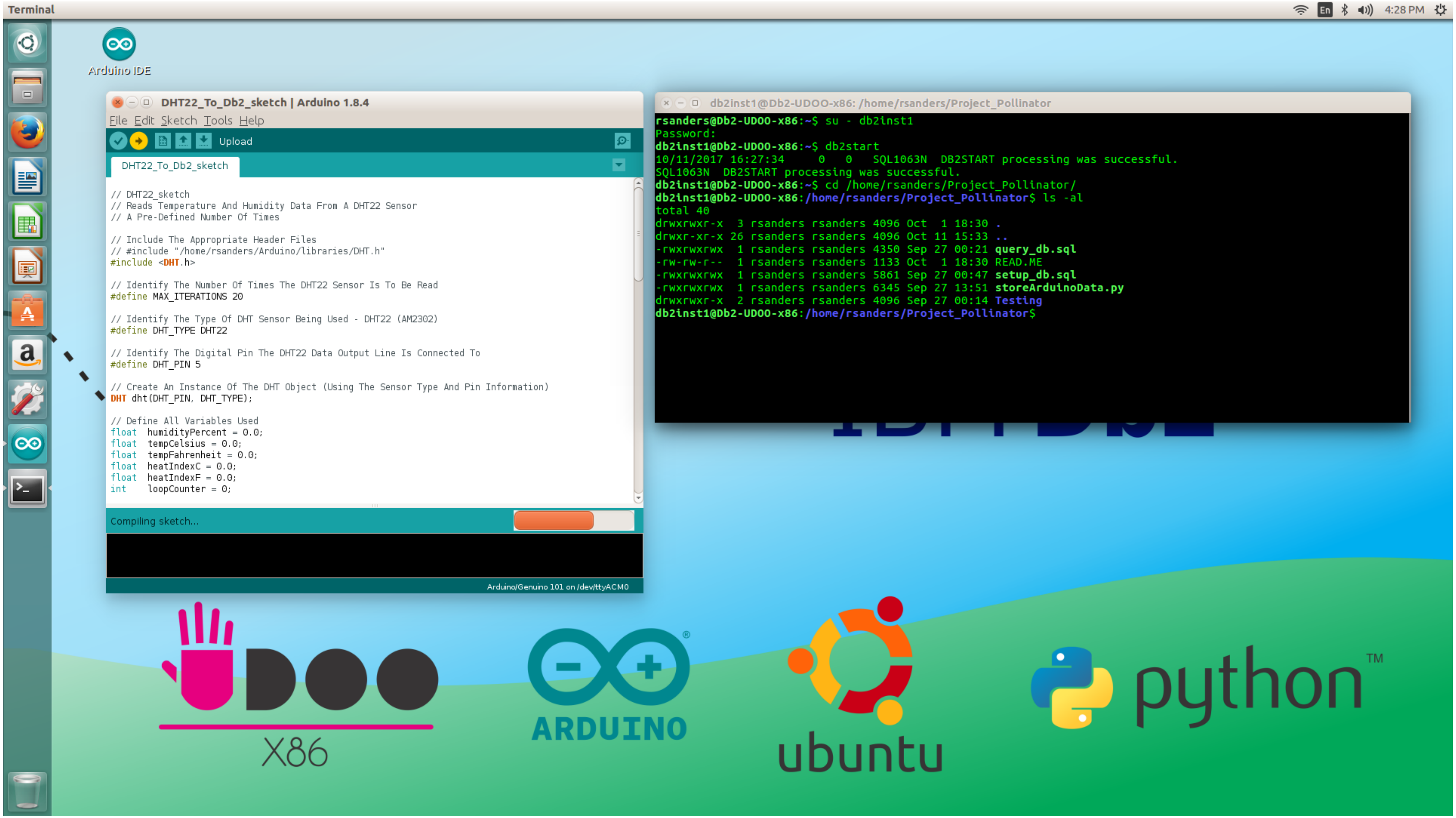
db2inst1@Db2-UDOO-x86: /home/rsanders/Project_Pollinator

```
rsanders@Db2-UDOO-x86:~$ su - db2inst1
Password:
db2inst1@Db2-UDOO-x86:~$ db2start
10/11/2017 16:27:34     0   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.
db2inst1@Db2-UDOO-x86:~$ cd /home/rsanders/Project_Pollinator/
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$ ls -al
total 40
drwxrwxr-x  3 rsanders rsanders 4096 Oct  1 18:30 .
drwxr-xr-x 26 rsanders rsanders 4096 Oct 11 15:33 ..
-rwxrwxrwx  1 rsanders rsanders 4350 Sep 27 00:21 query_db.sql
-rw-rw-r--  1 rsanders rsanders 1133 Oct  1 18:30 READ.ME
-rwxrwxrwx  1 rsanders rsanders 5861 Sep 27 00:47 setup_db.sql
-rwxrwxrwx  1 rsanders rsanders 6345 Sep 27 13:51 storeArduinoData.py
drwxrwxr-x  2 rsanders rsanders 4096 Sep 27 00:14 Testing
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$ 
```


Terminal
4:28 PM
Arduino IDE
DHT22_To_Db2_sketch | Arduino 1.8.4
File Edit Sketch Tools Help
Upload
DHT22_To_Db2_sketch
Arduino/Genuino 101 on /dev/ttyACM0
db2inst1@Db2-UDOO-x86: /home/rsanders/Project_Pollinator
UDOO X86
ARDUINO
ubuntu
python™

```
// DHT22_sketch
// Reads Temperature And Humidity Data From A DHT22 Sensor
// A Pre-Defined Number Of Times

// Include The Appropriate Header Files
// #include "/home/rsanders/Arduino/libraries/DHT.h"
#include <DHT.h>

// Identify The Number Of Times The DHT22 Sensor Is To Be Read
#define MAX_ITERATIONS 20

// Identify The Type Of DHT Sensor Being Used - DHT22 (AM2302)
#define DHT_TYPE DHT22

// Identify The Digital Pin The DHT22 Data Output Line Is Connected To
#define DHT_PIN 5

// Create An Instance Of The DHT Object (Using The Sensor Type And Pin Information)
DHT dht(DHT_PIN, DHT_TYPE);

// Define All Variables Used
float  humidityPercent = 0.0;
float  tempCelsius = 0.0;
float  tempFahrenheit = 0.0;
float  heatIndexC = 0.0;
float  heatIndexF = 0.0;
int    loopCounter = 0;
```

```
rsanders@Db2-UDOO-x86:~$ su - db2inst1
Password:
db2inst1@Db2-UDOO-x86:~$ db2start
10/11/2017 16:27:34     0   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.
db2inst1@Db2-UDOO-x86:~$ cd /home/rsanders/Project_Pollinator/
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$ ls -al
total 40
drwxrwxr-x  3 rsanders rsanders 4096 Oct  1 18:30 .
drwxr-xr-x 26 rsanders rsanders 4096 Oct 11 15:33 ..
-rwxrwxrwx  1 rsanders rsanders 4350 Sep 27 00:21 query_db.sql
-rw-rw-r--  1 rsanders rsanders 1133 Oct  1 18:30 READ.ME
-rwxrwxrwx  1 rsanders rsanders 5861 Sep 27 00:47 setup_db.sql
-rwxrwxrwx  1 rsanders rsanders 6345 Sep 27 13:51 storeArduinoData.py
drwxrwxr-x  2 rsanders rsanders 4096 Sep 27 00:14 Testing
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$
```

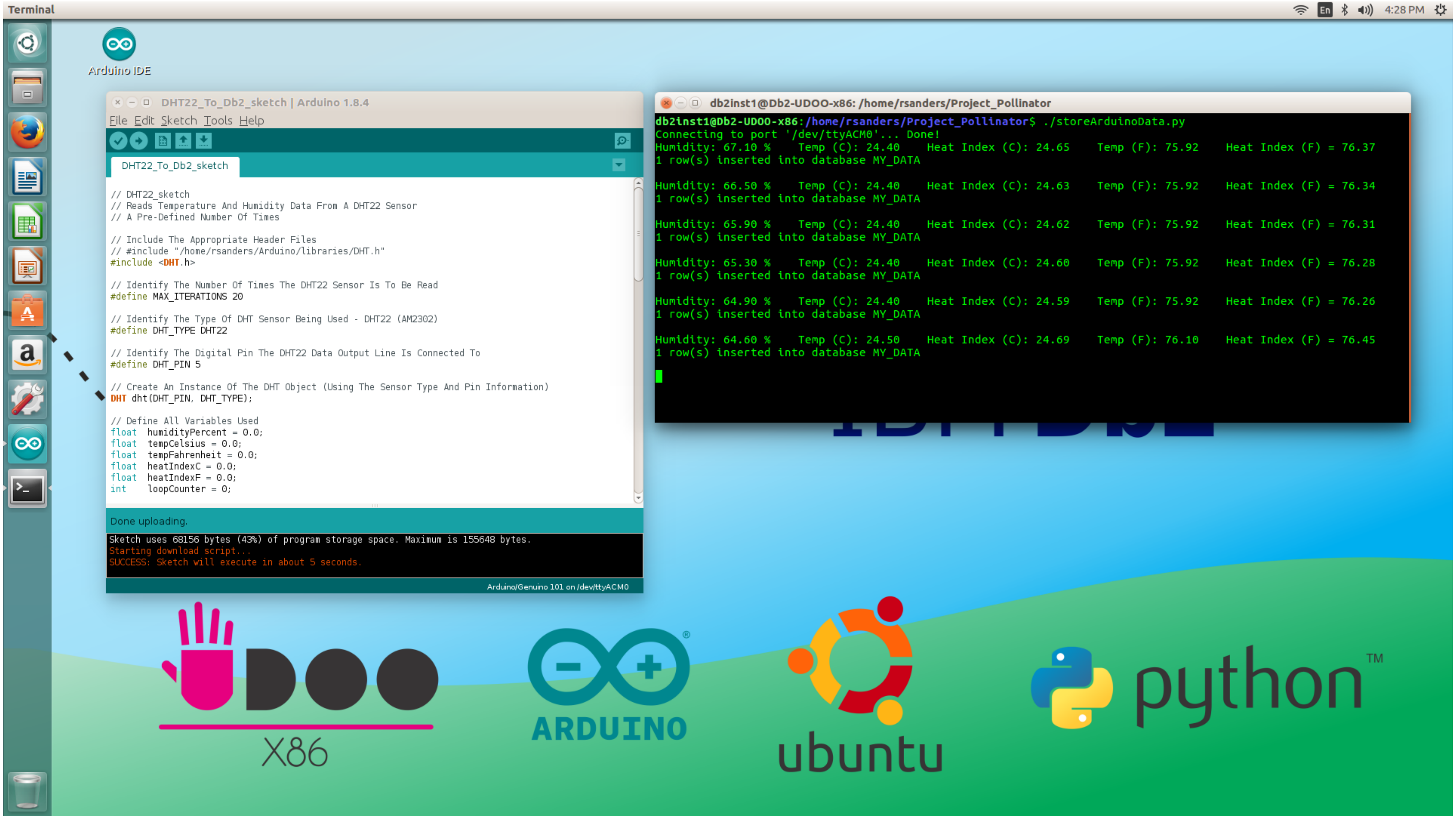

Terminal
En
4:28 PM
Arduino IDE
DHT22_To_Db2_sketch | Arduino 1.8.4
File Edit Sketch Tools Help
Upload
DHT22_To_Db2_sketch
db2inst1@Db2-UDOO-x86: /home/rsanders/Project_Pollinator
X86
ARDUINO
ubuntu
python™

```
// DHT22_sketch
// Reads Temperature And Humidity Data From A DHT22 Sensor
// A Pre-Defined Number Of Times

// Include The Appropriate Header Files
// #include "/home/rsanders/Arduino/libraries/DHT.h"
#include <DHT.h>

// Identify The Number Of Times The DHT22 Sensor Is To Be Read
#define MAX_ITERATIONS 20

// Identify The Type Of DHT Sensor Being Used - DHT22 (AM2302)
#define DHT_TYPE DHT22

// Identify The Digital Pin The DHT22 Data Output Line Is Connected To
#define DHT_PIN 5

// Create An Instance Of The DHT Object (Using The Sensor Type And Pin Information)
DHT dht(DHT_PIN, DHT_TYPE);

// Define All Variables Used
float  humidityPercent = 0.0;
float  tempCelsius = 0.0;
float  tempFahrenheit = 0.0;
float  heatIndexC = 0.0;
float  heatIndexF = 0.0;
int    loopCounter = 0;
```

Compiling sketch...

Arduino/Genuino 101 on /dev/ttyACM0

```
rsanders@Db2-UDOO-x86:~$ su - db2inst1
Password:
db2inst1@Db2-UDOO-x86:~$ db2start
10/11/2017 16:27:34     0   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.
db2inst1@Db2-UDOO-x86:~$ cd /home/rsanders/Project_Pollinator/
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$ ls -al
total 40
drwxrwxr-x  3 rsanders rsanders 4096 Oct  1 18:30 .
drwxr-xr-x 26 rsanders rsanders 4096 Oct 11 15:33 ..
-rwxrwxrwx  1 rsanders rsanders 4350 Sep 27 00:21 query_db.sql
-rw-rw-r--  1 rsanders rsanders 1133 Oct  1 18:30 READ.ME
-rwxrwxrwx  1 rsanders rsanders 5861 Sep 27 00:47 setup_db.sql
-rwxrwxrwx  1 rsanders rsanders 6345 Sep 27 13:51 storeArduinoData.py
drwxrwxr-x  2 rsanders rsanders 4096 Sep 27 00:14 Testing
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$
```



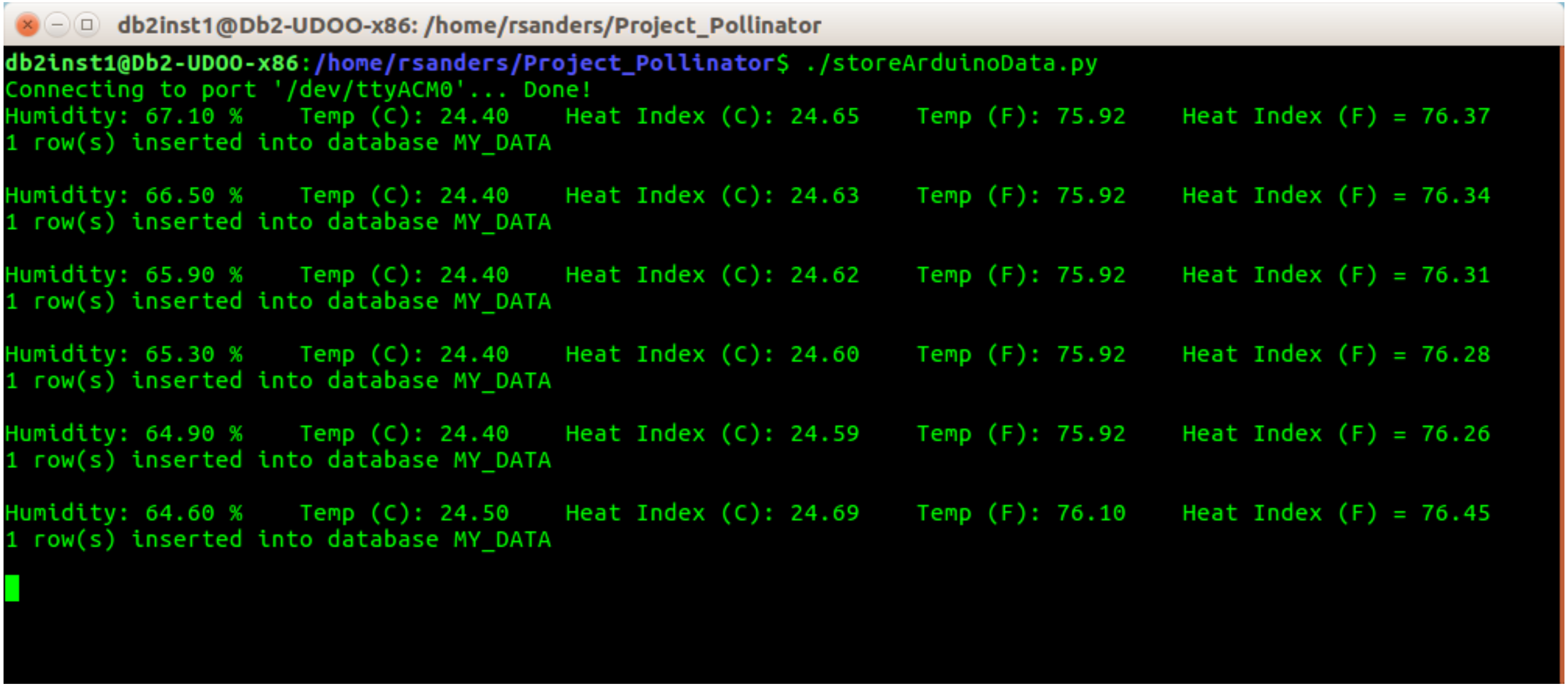
Terminal
4:28 PM
Arduino IDE
DHT22_To_Db2_sketch | Arduino 1.8.4
File Edit Sketch Tools Help
DHT22_To_Db2_sketch
Arduino/Genuino 101 on /dev/ttyACM0
db2inst1@Db2-UDOO-x86: /home/rsanders/Project_Pollinator
X86
ARDUINO
ubuntu
python™

```
// DHT22_sketch
// Reads Temperature And Humidity Data From A DHT22 Sensor
// A Pre-Defined Number Of Times

// Include The Appropriate Header Files
// #include "/home/rsanders/Arduino/libraries/DHT.h"
#include <DHT.h>

// Identify The Number Of Times The DHT22 Sensor Is To Be Read
#define MAX_ITERATIONS 20

// Identify The Type Of DHT Sensor Being Used - DHT22 (AM2302)
#define DHT_TYPE DHT22

// Identify The Digital Pin The DHT22 Data Output Line Is Connected To
#define DHT_PIN 5

// Create An Instance Of The DHT Object (Using The Sensor Type And Pin Information)
DHT dht(DHT_PIN, DHT_TYPE);

// Define All Variables Used
float  humidityPercent = 0.0;
float  tempCelsius = 0.0;
float  tempFahrenheit = 0.0;
float  heatIndexC = 0.0;
float  heatIndexF = 0.0;
int    loopCounter = 0;
```

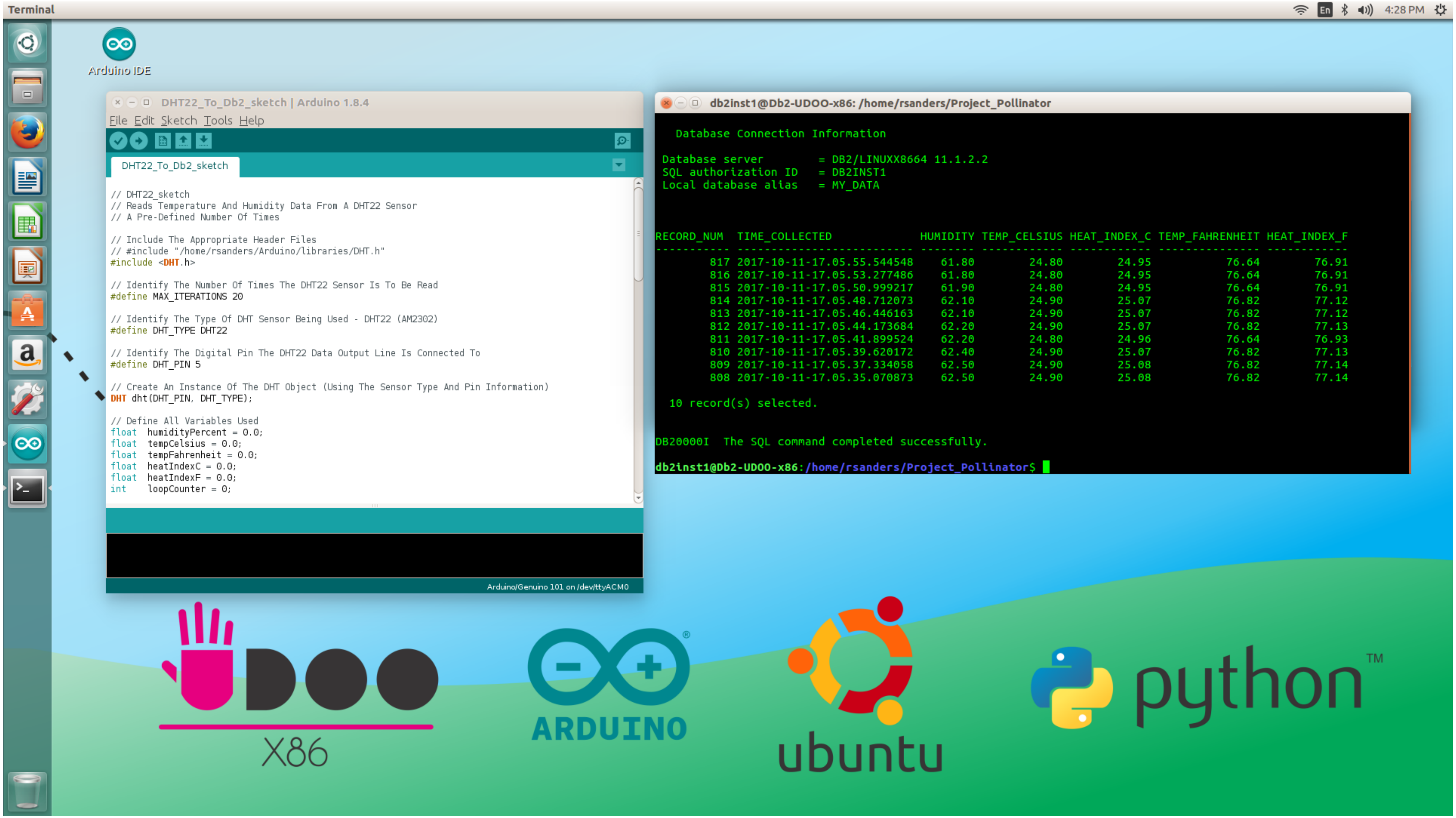
Done uploading.

```
Sketch uses 68156 bytes (43%) of program storage space. Maximum is 155648 bytes.
Starting download script...
SUCCESS: Sketch will execute in about 5 seconds.
```

```
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$ ./storeArduinoData.py
Connecting to port '/dev/ttyACM0'... Done!
Humidity: 67.10 %    Temp (C): 24.40    Heat Index (C): 24.65    Temp (F): 75.92    Heat Index (F) = 76.37
1 row(s) inserted into database MY_DATA

Humidity: 66.50 %    Temp (C): 24.40    Heat Index (C): 24.63    Temp (F): 75.92    Heat Index (F) = 76.34
1 row(s) inserted into database MY_DATA

Humidity: 65.90 %    Temp (C): 24.40    Heat Index (C): 24.62    Temp (F): 75.92    Heat Index (F) = 76.31
1 row(s) inserted into database MY_DATA

Humidity: 65.30 %    Temp (C): 24.40    Heat Index (C): 24.60    Temp (F): 75.92    Heat Index (F) = 76.28
1 row(s) inserted into database MY_DATA

Humidity: 64.90 %    Temp (C): 24.40    Heat Index (C): 24.59    Temp (F): 75.92    Heat Index (F) = 76.26
1 row(s) inserted into database MY_DATA

Humidity: 64.60 %    Temp (C): 24.50    Heat Index (C): 24.69    Temp (F): 76.10    Heat Index (F) = 76.45
1 row(s) inserted into database MY_DATA
```

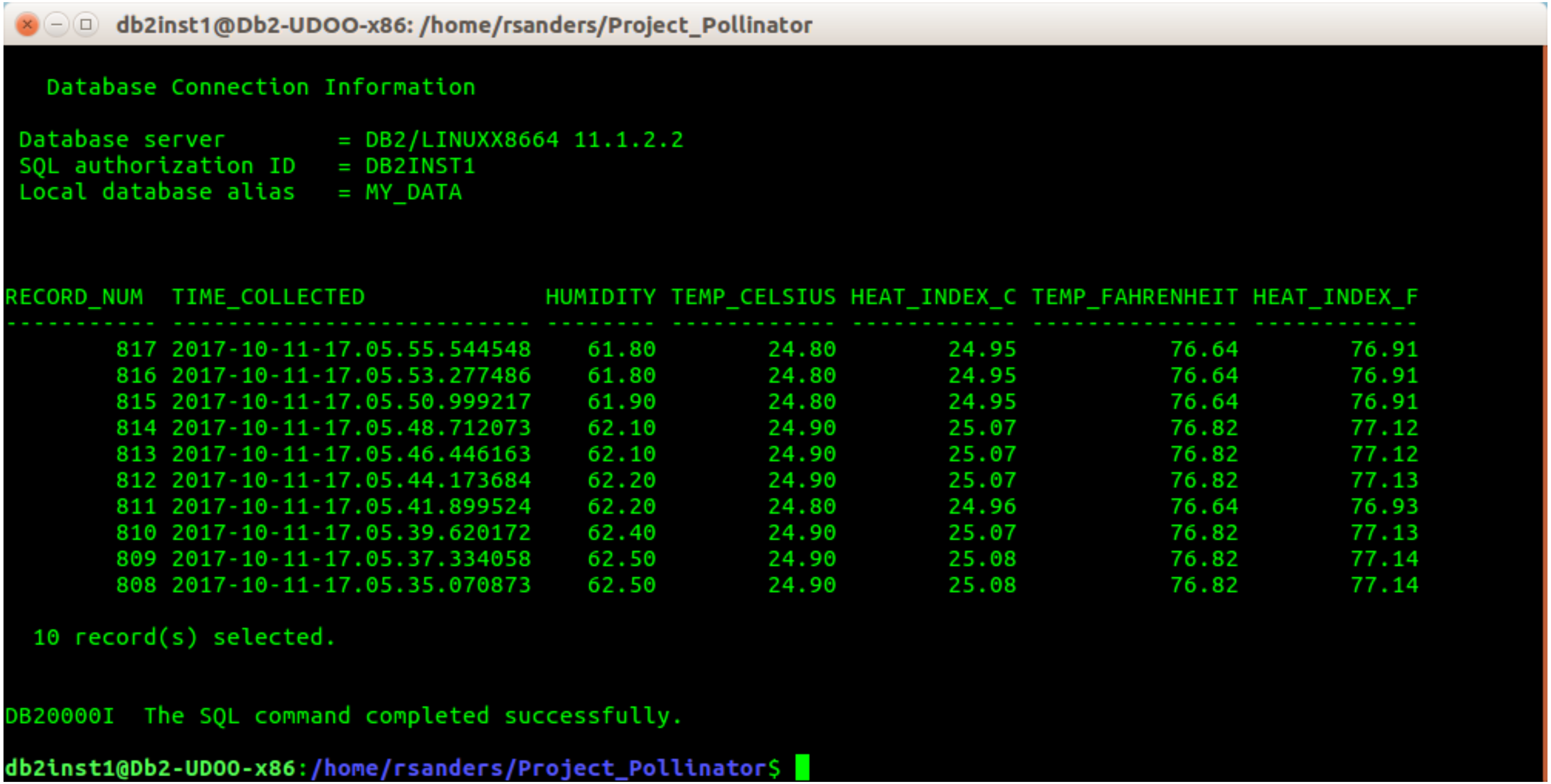


```
db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$ ./storeArduinoData.py
Connecting to port '/dev/ttyACM0'... Done!
Humidity: 67.10 %    Temp (C): 24.40    Heat Index (C): 24.65    Temp (F): 75.92    Heat Index (F) = 76.37
1 row(s) inserted into database MY_DATA

Humidity: 66.50 %    Temp (C): 24.40    Heat Index (C): 24.63    Temp (F): 75.92    Heat Index (F) = 76.34
1 row(s) inserted into database MY_DATA

Humidity: 65.90 %    Temp (C): 24.40    Heat Index (C): 24.62    Temp (F): 75.92    Heat Index (F) = 76.31
1 row(s) inserted into database MY_DATA

Humidity: 65.30 %    Temp (C): 24.40    Heat Index (C): 24.60    Temp (F): 75.92    Heat Index (F) = 76.28
1 row(s) inserted into database MY_DATA

Humidity: 64.90 %    Temp (C): 24.40    Heat Index (C): 24.59    Temp (F): 75.92    Heat Index (F) = 76.26
1 row(s) inserted into database MY_DATA

Humidity: 64.60 %    Temp (C): 24.50    Heat Index (C): 24.69    Temp (F): 76.10    Heat Index (F) = 76.45
1 row(s) inserted into database MY_DATA

```


Terminal
4:28 PM
En
Arduino IDE
DHT22_To_Db2_sketch | Arduino 1.8.4
File Edit Sketch Tools Help
DHT22_To_Db2_sketch
Arduino/Genuino 101 on /dev/ttyACM0
db2inst1@Db2-UDOO-x86: /home/rsanders/Project_Pollinator
X86
ARDUINO
ubuntu
python™

```
// DHT22_sketch
// Reads Temperature And Humidity Data From A DHT22 Sensor
// A Pre-Defined Number Of Times

// Include The Appropriate Header Files
// #include "/home/rsanders/Arduino/libraries/DHT.h"
#include <DHT.h>

// Identify The Number Of Times The DHT22 Sensor Is To Be Read
#define MAX_ITERATIONS 20

// Identify The Type Of DHT Sensor Being Used - DHT22 (AM2302)
#define DHT_TYPE DHT22

// Identify The Digital Pin The DHT22 Data Output Line Is Connected To
#define DHT_PIN 5

// Create An Instance Of The DHT Object (Using The Sensor Type And Pin Information)
DHT dht(DHT_PIN, DHT_TYPE);

// Define All Variables Used
float  humidityPercent = 0.0;
float  tempCelsius = 0.0;
float  tempFahrenheit = 0.0;
float  heatIndexC = 0.0;
float  heatIndexF = 0.0;
int    loopCounter = 0;
```

```
  Database Connection Information

 Database server        = DB2/LINUXX8664 11.1.2.2
 SQL authorization ID   = DB2INST1
 Local database alias   = MY_DATA


RECORD_NUM  TIME_COLLECTED             HUMIDITY TEMP_CELSIUS HEAT_INDEX_C TEMP_FAHRENHEIT HEAT_INDEX_F
----------- -------------------------- -------- ------------ ------------ --------------- ------------
        817 2017-10-11-17.05.55.544548    61.80        24.80        24.95           76.64        76.91
        816 2017-10-11-17.05.53.277486    61.80        24.80        24.95           76.64        76.91
        815 2017-10-11-17.05.50.999217    61.90        24.80        24.95           76.64        76.91
        814 2017-10-11-17.05.48.712073    62.10        24.90        25.07           76.82        77.12
        813 2017-10-11-17.05.46.446163    62.10        24.90        25.07           76.82        77.12
        812 2017-10-11-17.05.44.173684    62.20        24.90        25.07           76.82        77.13
        811 2017-10-11-17.05.41.899524    62.20        24.80        24.96           76.64        76.93
        810 2017-10-11-17.05.39.620172    62.40        24.90        25.07           76.82        77.13
        809 2017-10-11-17.05.37.334058    62.50        24.90        25.08           76.82        77.14
        808 2017-10-11-17.05.35.070873    62.50        24.90        25.08           76.82        77.14

  10 record(s) selected.


DB20000I  The SQL command completed successfully.

db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$
```



```
   Database Connection Information

 Database server        = DB2/LINUXX8664 11.1.2.2
 SQL authorization ID   = DB2INST1
 Local database alias   = MY_DATA



RECORD_NUM  TIME_COLLECTED             HUMIDITY TEMP_CELSIUS HEAT_INDEX_C TEMP_FAHRENHEIT HEAT_INDEX_F
----------- -------------------------- -------- ------------ ------------ --------------- ------------
        817 2017-10-11-17.05.55.544548    61.80        24.80        24.95           76.64        76.91
        816 2017-10-11-17.05.53.277486    61.80        24.80        24.95           76.64        76.91
        815 2017-10-11-17.05.50.999217    61.90        24.80        24.95           76.64        76.91
        814 2017-10-11-17.05.48.712073    62.10        24.90        25.07           76.82        77.12
        813 2017-10-11-17.05.46.446163    62.10        24.90        25.07           76.82        77.12
        812 2017-10-11-17.05.44.173684    62.20        24.90        25.07           76.82        77.13
        811 2017-10-11-17.05.41.899524    62.20        24.80        24.96           76.64        76.93
        810 2017-10-11-17.05.39.620172    62.40        24.90        25.07           76.82        77.13
        809 2017-10-11-17.05.37.334058    62.50        24.90        25.08           76.82        77.14
        808 2017-10-11-17.05.35.070873    62.50        24.90        25.08           76.82        77.14

  10 record(s) selected.


DB20000I  The SQL command completed successfully.

db2inst1@Db2-UDOO-x86:/home/rsanders/Project_Pollinator$
```

Call to Action:

Tell us about a project you’d like to do with a “Project Pollinator” starter kit.

http://ibm.biz/D-Bee-2

Thank You