**Objective**: In this lab you will learn about Azure and connect the Raspberry Pi to Azure IoT services.

**Required Setup**: Connect GrovePi+ board to RPi and have all GrovePi+ libraries installed.

**Parts:**

* RPi 3 B

**Part A. Introduce Microsoft Azure IoT**

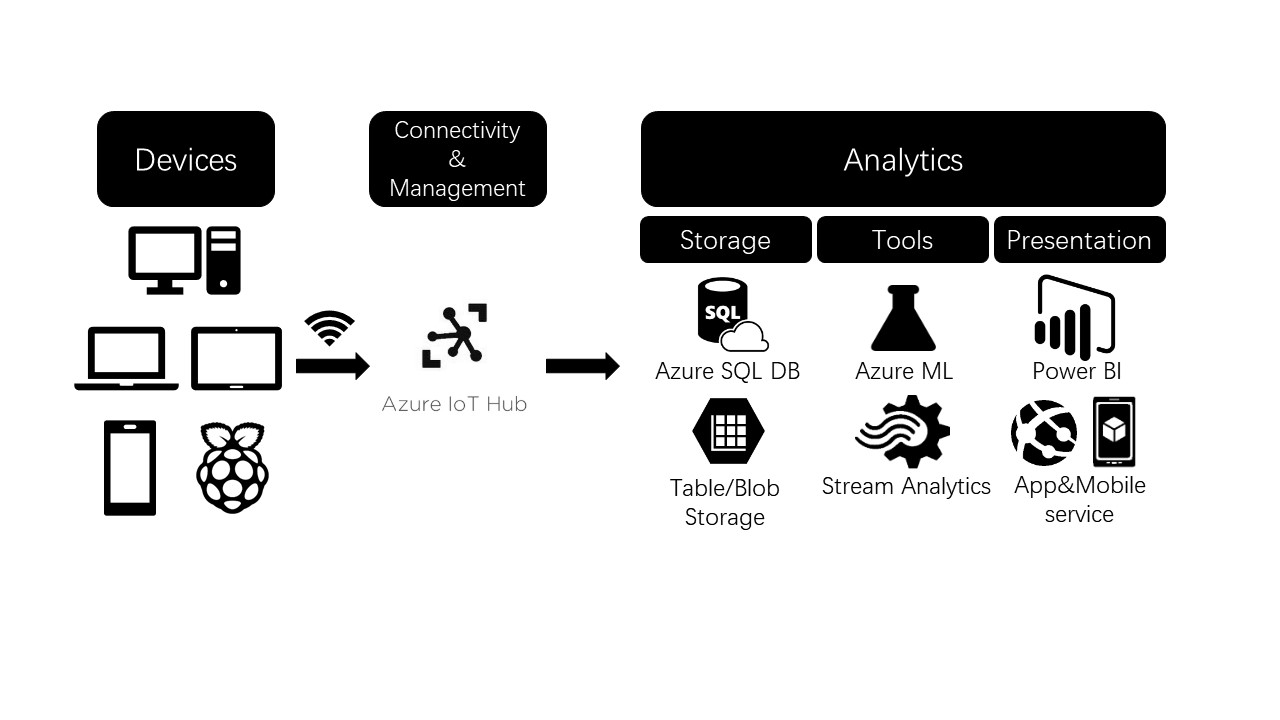
Internet of things is a new wave of computing that builds on small devices/sensors and many cloud services. IoT is a fast-growing technique that you can have the ability to remote control devices and collect data in the real world which were never possible before. There are a lot of advantages of IoT applications and you will soon see the wide range of possibilities of them. Here are four possible scenarios designed by Microsoft.

Microsoft connected Vehicle Platform: https://www.youtube.com/watch?v=wk2hflbXM6I&feature=youtu.be

Microsoft IoT for smart buildings: https://www.youtube.com/watch?v=d55rBuB9D7s

Microsoft connected Water Filler: <https://www.youtube.com/watch?v=eM1Nbz7DggI&feature=youtu.be>

Microsoft smart Waste Management: https://www.youtube.com/watch?v=NS6yd21xnhk&feature=youtu.be



Azure provides tons of IoT services to help us achieve or accelerate our applications. The above flow chart explains the whole flow of the framework in the high level. First, we can access all those cloud services on basically any platform, including raspberry pi, through a cloud gateway ---- Azure IoT hub. It maintains the communication between the devices and the Azure cloud to support data ingestion, command or control, and the device management.

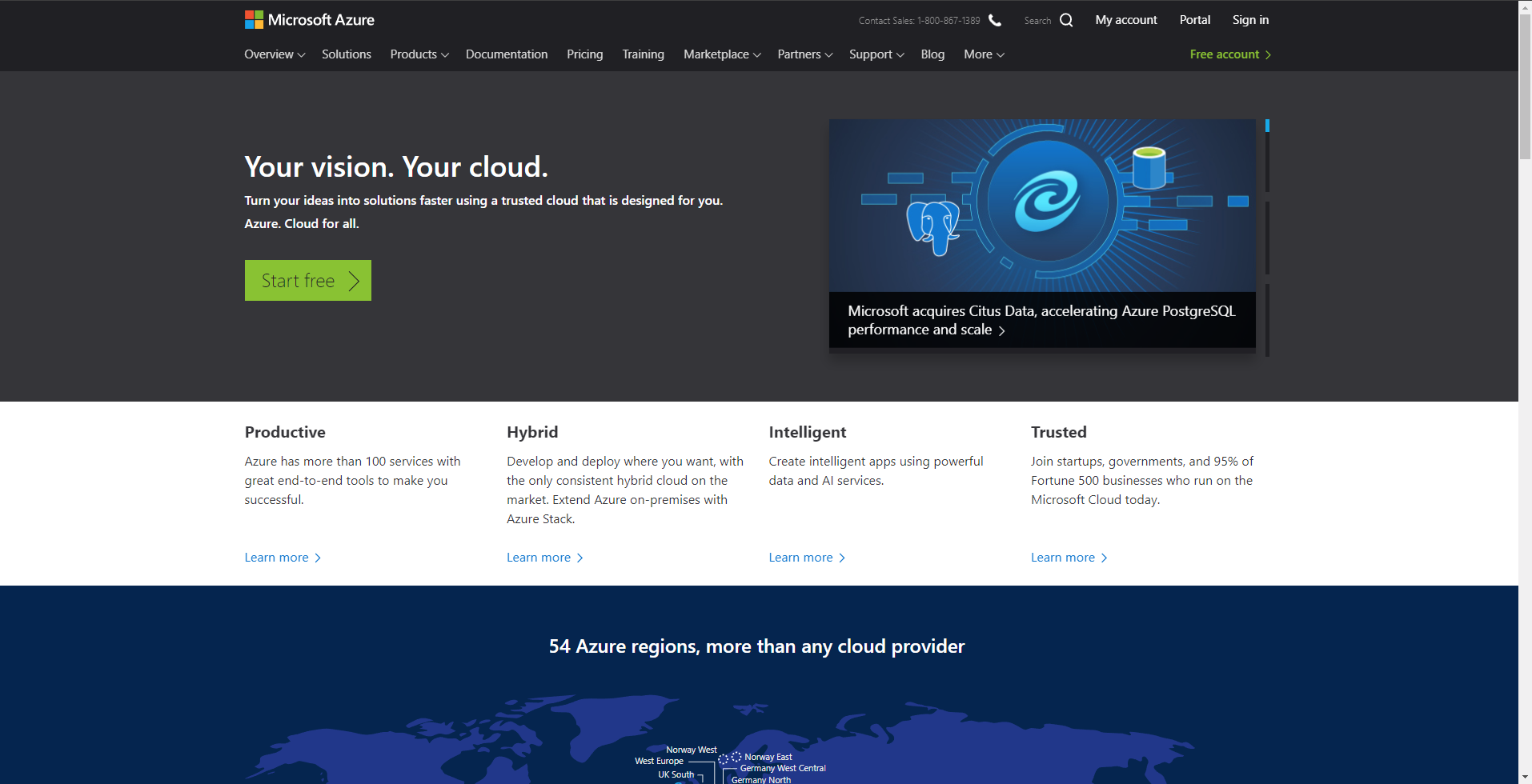
After that, you can start your analysis using all the related Azure services. For example, one of the tools is Azure Machine Learning to actually derive insights from the data you want to analyze. This is called “cold path analytics” which is a common method where the data comes in, you store the data into SQL database and then do analysis on it. The other one is called “hot path analytics” which you do real-time analysis on the data flow. In this scenario, Azure provides this service called “Stream Analytics”. It’s easy to set up and response quickly. We are going to implement this service with our raspberry Pi project.

For storage, you can implement regular relational database(Azure SQL DB), big data, key-valued database, documented database, and unstructured database at massive scale(Table/Blob Storage). For presentation, Power BI provides a rich, interactive dashboard for visualization. Furthermore, App&Mobile service allows us to quickly create website, web-based applications, and even mobile apps on vary platforms. Since we are using 100% Azure service in this course, those services are going to be communicating with each other flawlessly.

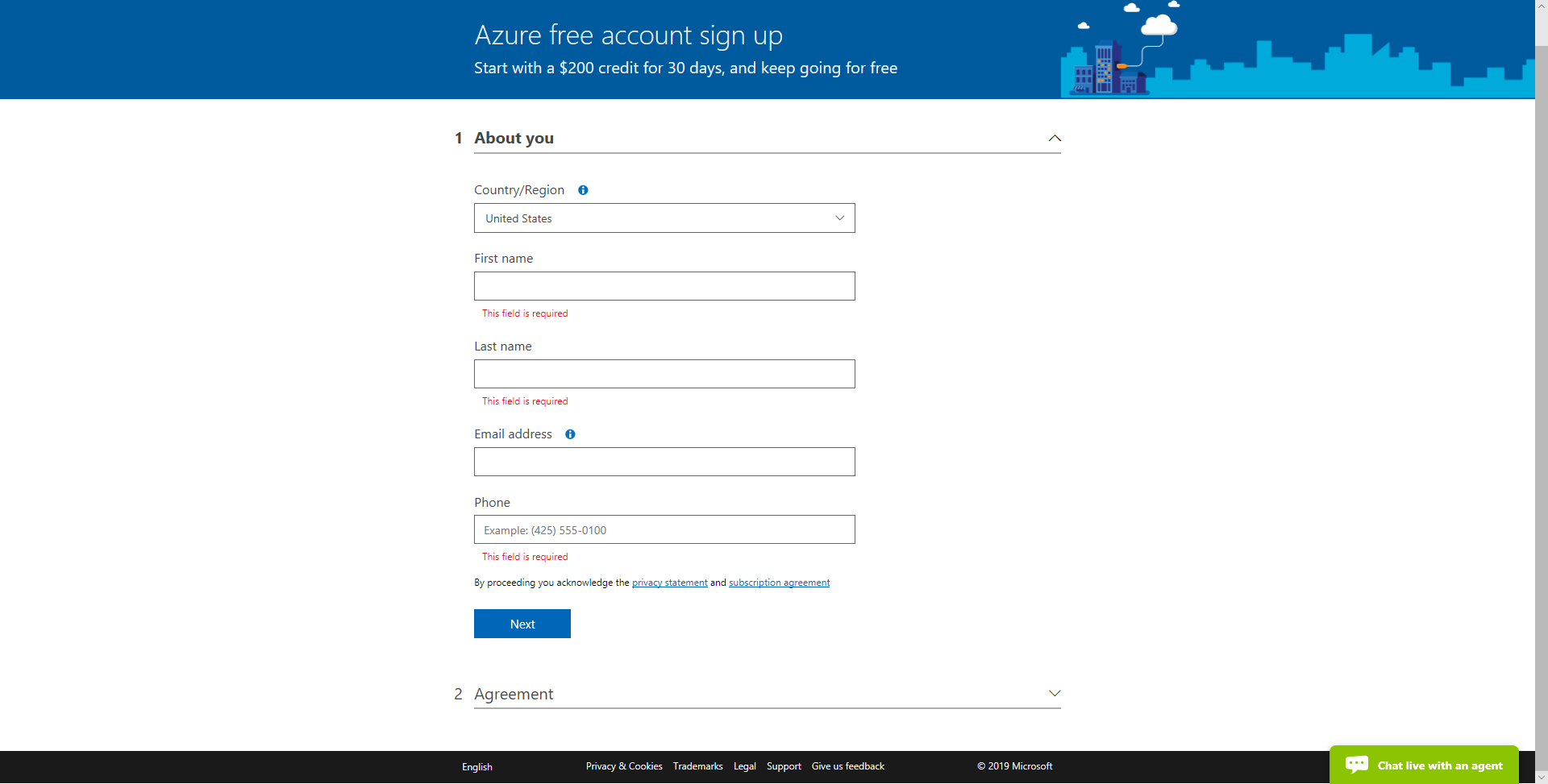
Now, you should have some base knowledge of the Azure IoT reference architecture. There are a lot more services available on Azure than what I just mentioned. Last week, we used the raspberry Pi and sensors to collect data and build a SQL database. We are going to connect the Raspberry Pi to the Azure IoT first for part B. Then We are going to explore the Azure dashboard and more services, then re-construct the database using Azure SQL database.

**Part B. Connect Raspberry Pi to the Azure IoT**

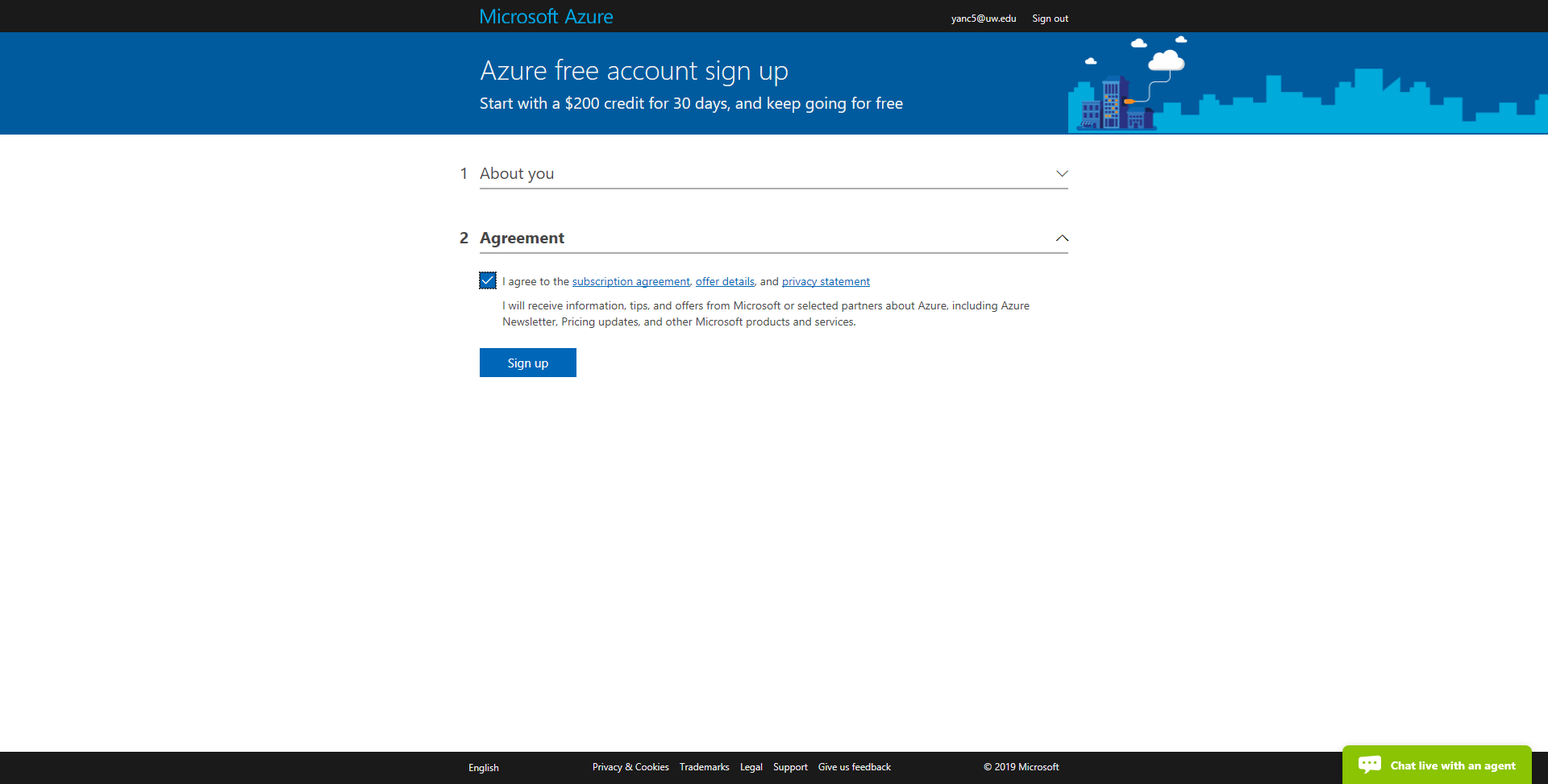
**Step 1.** Click on the following link to create a free Microsoft Azure account: <https://azure.microsoft.com/en-us/>. Then Click on ‘Start Free’. After that Click on ‘Start free’ again.



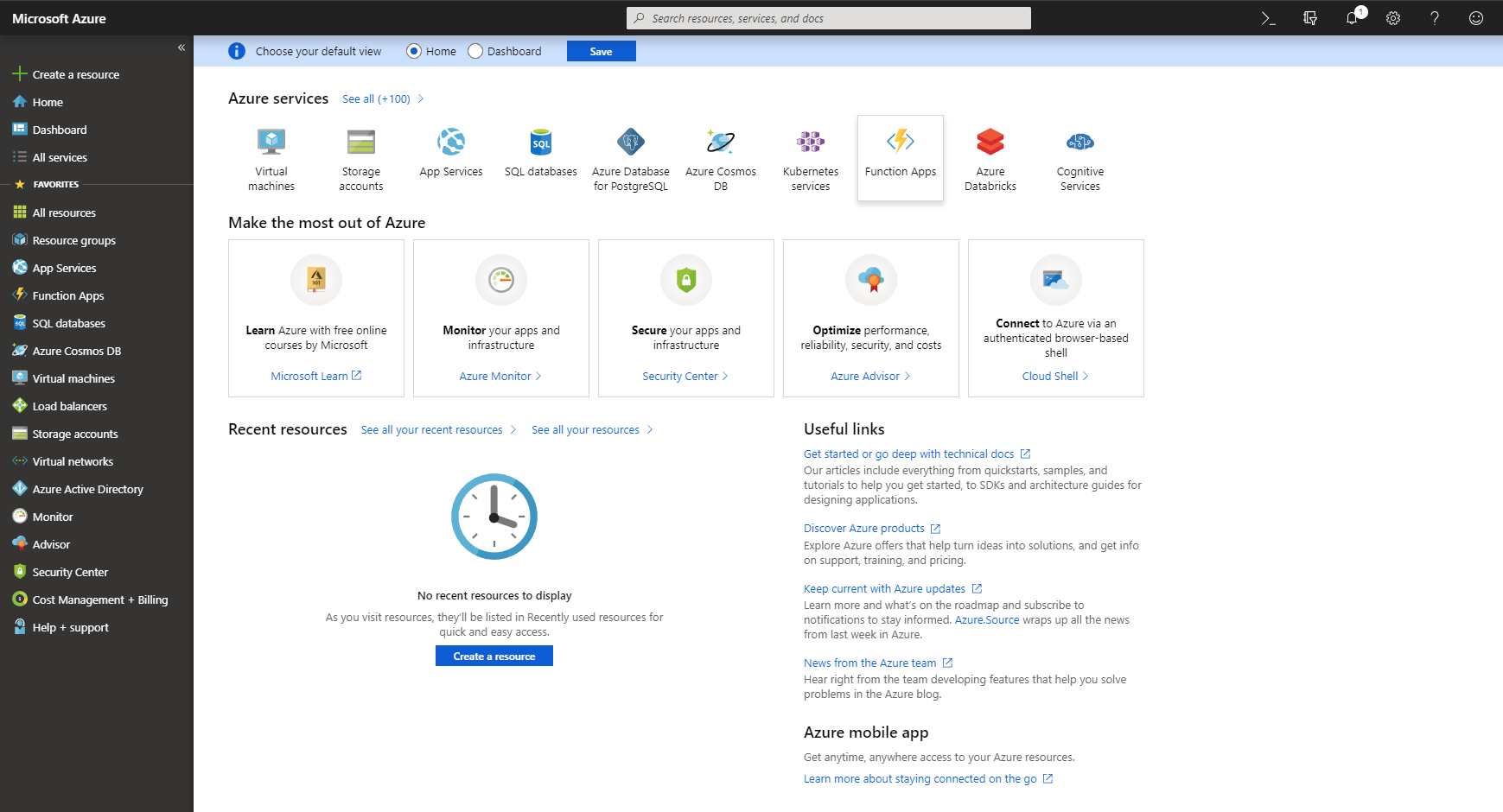
**Step 2.** The browser will prompt a sign-in portal for a Microsoft account. If you have one, go ahead and sign in; if you don’t, click on ‘Create one!’ and follow the step to create a Microsoft account. It’s highly recommended that you use your school/organization email to sign up the account. After you eventually sign in with your Microsoft account, you will be prompted to fill up a few more information.



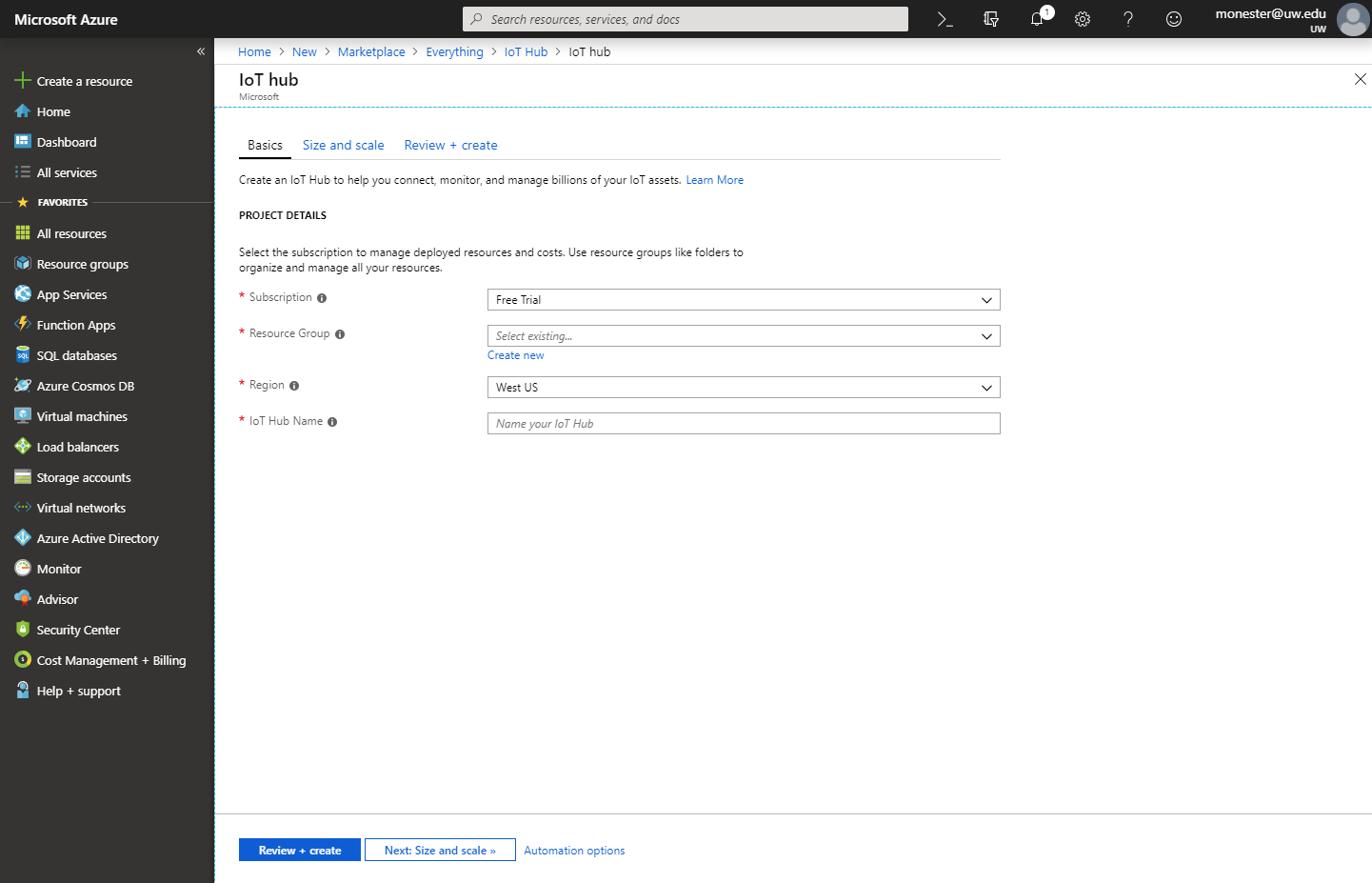
**Step 3.** Then check the box and click on the ‘sign up’ button.



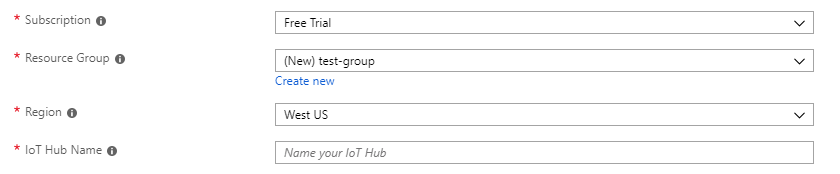
**Step 4.** You should be getting a verification/confirmation email. Now you are able to log in with your account you just created. (Notice, there might be some follow-up emails about offering lectures of Azure. You will be provided enough materials to keep you on the track of this course.) Now you should see this portal.



**Step 5.** Click on ‘+Create a resource’, then search “IoT hub”. Choose the first thing from the list. Click ‘create’ and you should see the similar screen as following.

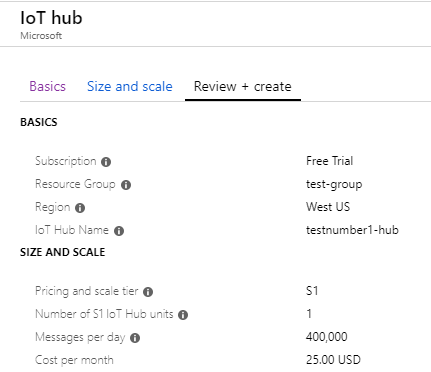


**Step 6.** For resource group, click on ‘Create new’. Then put ‘test-group’ as Name. Next, pick any name as your IoT hub Name(Noticed, the name must be globally unique). Leave other as they are initially. Click on ‘Next: Size and scale’. 

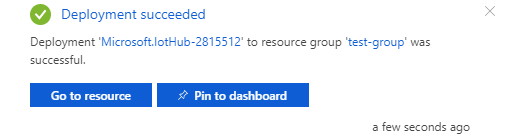


**Step 7.** On the next screen, simply click ‘Review + create’ button on the bottom.

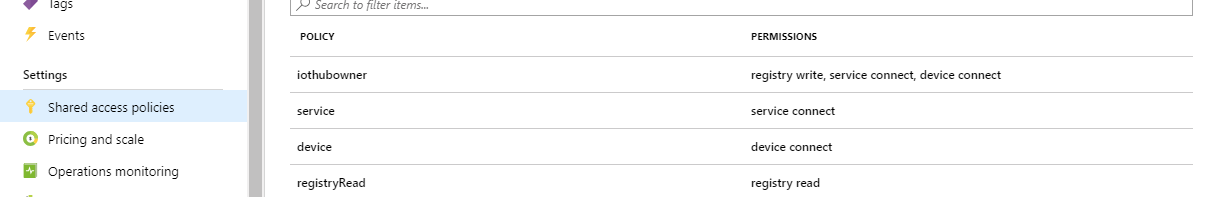
**Step 8.** Click ‘Create’ on the bottom of this page again. It might take a while to complete the deployment.



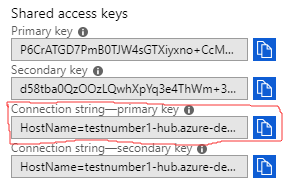
**Step 9.** When it’s completed, you will see the following in the notification. Click on ‘Go to resource’.



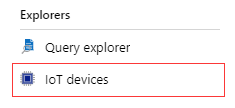
**Step 10.** Choose ‘Shared access policies’ in the settings from the IoT-hub panel. Select ‘iothubowner’.



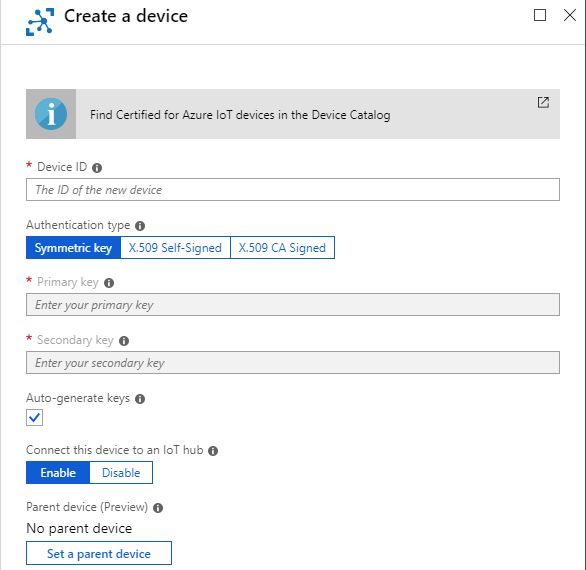
**Step 11.** From the panel and under ‘Shared access keys’, copy the Connection string-primary key to use later.



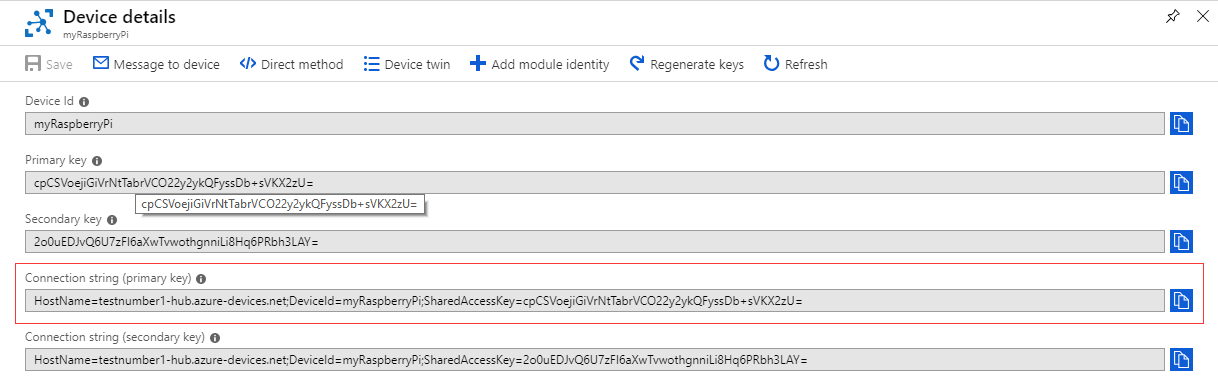
**Step 12.** Now we need to continue to set up a new device in the IoT hub. Choose ‘IoT devices’ in the ‘Explorers’ from the IoT-hub menu.



**Step 13.** Click ‘+Add’ to register a new device. You should see the following page. Put ‘myRaspberryPi’ in the Device ID and click ‘Save’.

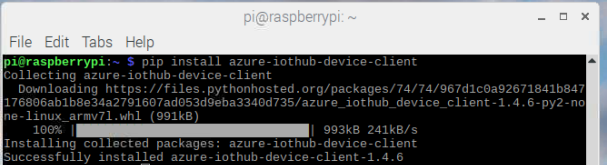


**Step 14.** After the new device is created, open the device from the list in the panel. Copy the Connection string (primary key) to use later.



**Step 15.** Now we are done with the Azure IoT hub set up. Let’s set up our Raspberry Pi. Open the terminal, then run the following command to install the Azure IoT hub device SDK.





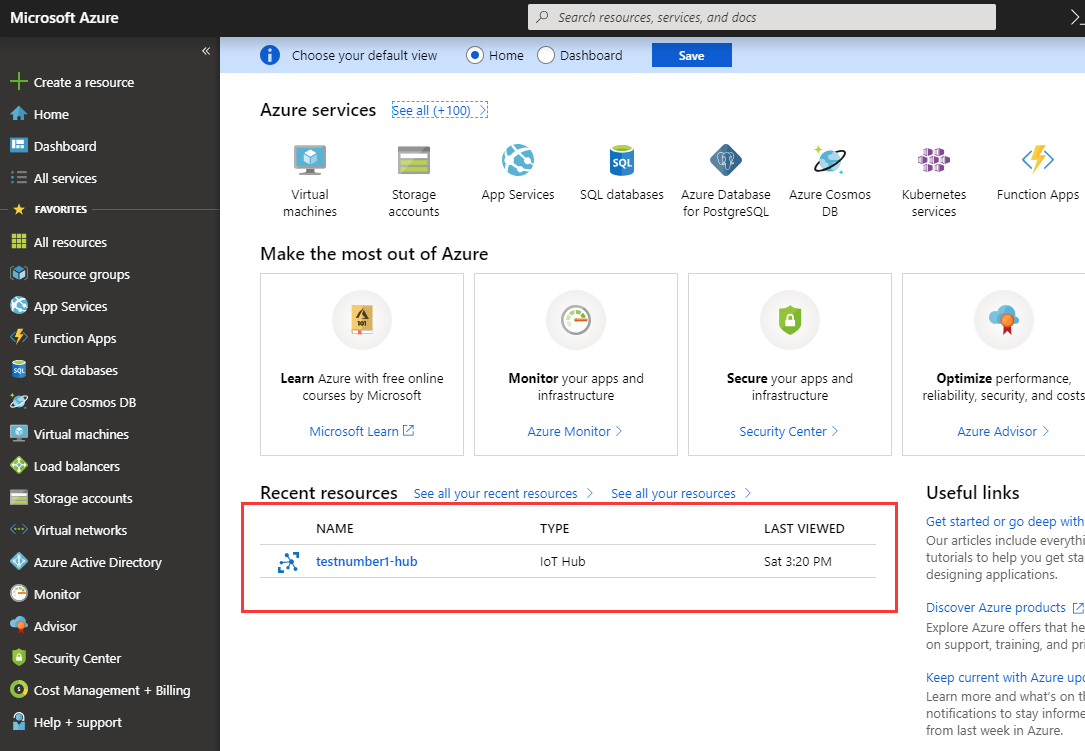
**Step 16.** Run the following command to update your raspberry Pi and libraries. Enter ‘y’ if anything prompts.

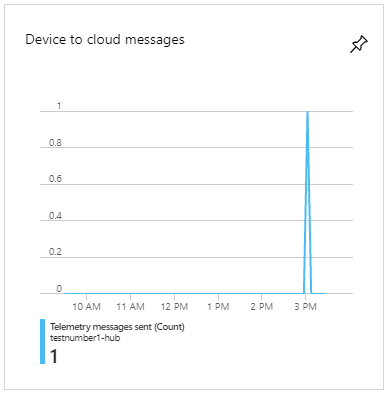


**Step 17.** Create an empty python script using Thonny. Copy paste all the code from ‘starter\_sample.py’ to it. You need to change the ‘Connection string’ variable to your connection string in the code. After running it, you should see the following output.



Step 18. Login to the Azure portal and click on the IoT hub on the home page.



**Step 19.** Scroll down IoT hub Overview. You will see a change in ‘Device to cloud messages’ usage. (You might wonder where to see the sent message. You need to do a message routing to pass the message to a storage location to be able to see it. Since this example is just to show case, we are not going to show the whole process here.) Congratulations! You just sent a message to the Azure IoT hub from your Raspberry Pi.

There are two ways for developer to connect to the Azure IoT hub. The first one is to use the Azure IoT hub device/service SDK with some supported programming language of your choice, which is exactly what we just did in this lab. Secondly, you can connect directly to the Azure IoT hub’s MQTT endpoint, which is basically using the REST API. We would introduce another way in the course later. Next week, we will continue exploring the IoT hub and try to achieve streaming sensor data in real-time with Stream Analytics and Power BI.

Reference:

1. <https://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/setting-software/>

2. <http://download.microsoft.com/download/A/4/D/A4DAD253-BC21-41D3-B9D9-87D2AE6F0719/Microsoft_Azure_IoT_Reference_Architecture.pdf>

3. <https://courses.edx.org/courses/course-v1:Microsoft+DEV225x+3T2018/course/#block-v1:Microsoft+DEV225x+3T2018+type@sequential+block@dea6cca6-87fa-c235-3c16-b9b8d83fc60f>

4. <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-raspberry-pi-kit-node-get-started>

5. <https://github.com/Azure/azure-iot-sdk-python/tree/master/device/samples>

6. <https://github.com/Azure/azure-iot-sdk-python>