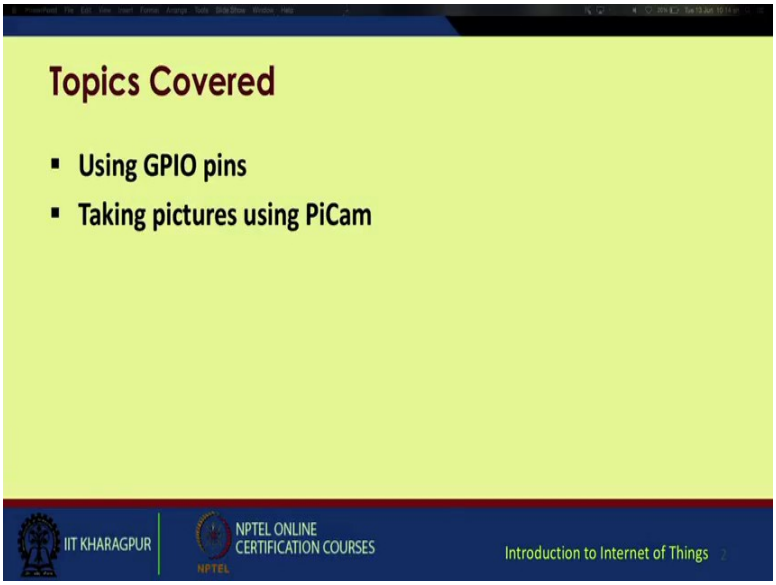
Introduction to Internet of Things
Prof. Sudip Misra
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

Lecture - 29
Introduction to Raspberry Pi- II

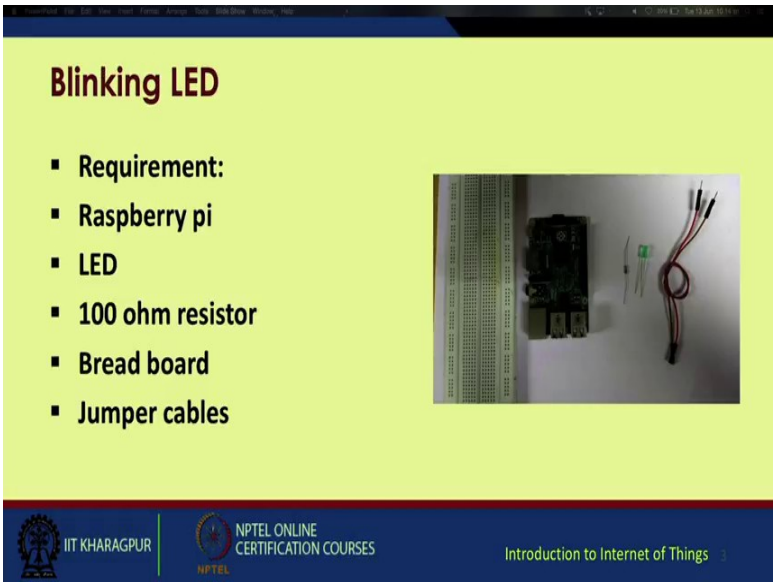
In this lecture, we will cover the bit more detailed introduction to raspberry pi, so that you will get a concise idea about how you can go about integrating various sensors and devices with a raspberry pi base system, and maybe programmatically access and manipulate them.

(Refer Slide Time: 00:41)



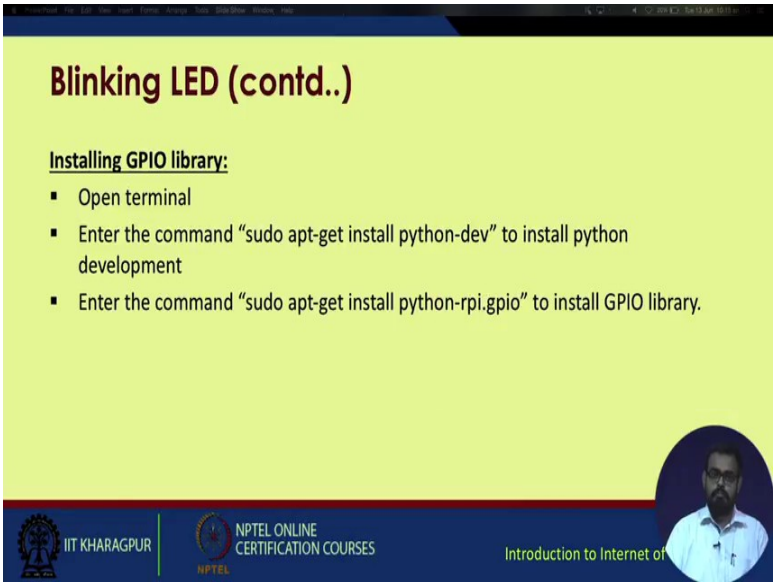
So, first of all we are going to cover how to use that basic GPIO pins and then you are going to cover how to integrate a pi camera with the raspberry pi for taking images.

(Refer Slide Time: 00:56)



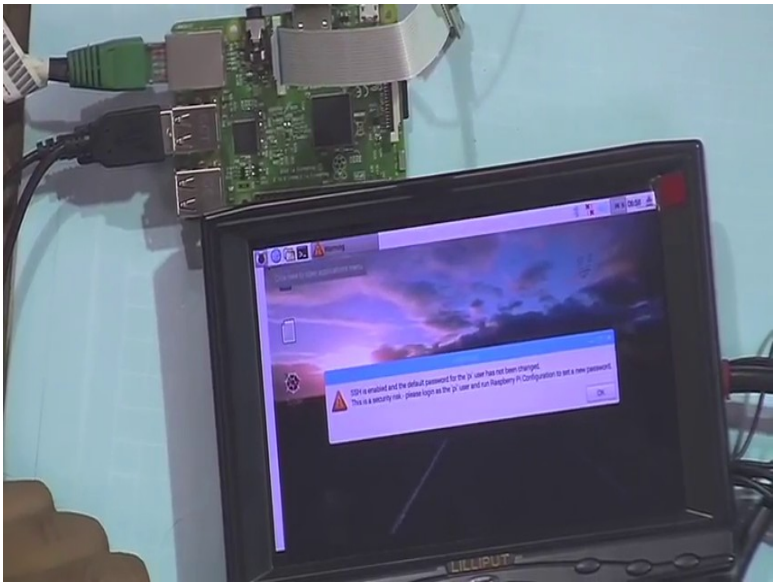
Point to note is we will be covering both of these things on python. So, the programming language will be python. So, since it is quite easy and easy to integrate with raspberry pi also, we will be preferring python So, first thing we are going to go about doing is a blinking LED based project. So, basic requirements are you are going to require a raspberry pi connected to the network, you are going to require an LED, a basic 100 ohm register breadboard and a few jumper cables.

(Refer Slide Time: 01:37)



So, before we go into this any further let us come back to the raspberry pi as we remember from the last lecture. As you can see I have not yet powered on this device yet, but I have attached a pi camera. So, this module is known as a pi camera and it goes over this side. So, you will see this is called a ribbon wire. So, it is already available with the pi cam and the blue sided should be pointed towards the Ethernet port, this is one point to remember. You take this white jack out plug this ribbon and push the white jack in and that is your that will attach your pi camera to your raspberry pi.

(Refer Slide Time: 01:45)



So, now we are going to power up this system. Now, since my device is booted up. I will check whether we have network access or not. Now, we have got a functional Ethernet configuration ready for operation. We have got the same IP as the previous example. Now, in the last example, in the last lecture, I showed you how to connect your raspberry pi over SSH from any PC. So, using SSH you can mainly execute and run various programs suppose you want to transfer some file this would not be this operation you would not be able to do using SSH. You can, but it is quite tiresome. So, the easier way out is using a ftp client. So, one I prefer is Filezilla it is available freely online.

So, once I open the Filezilla, the left hand panel generally shows my local PC contents and the right hand side will be my remote PC contents. So, I am connecting to the host the IP address was I put in the IP user name was pi password was raspberry, and under ports it give 22. So, this is fixed for ftp, to give port 22 and then quick connect. So, it will ask you will get

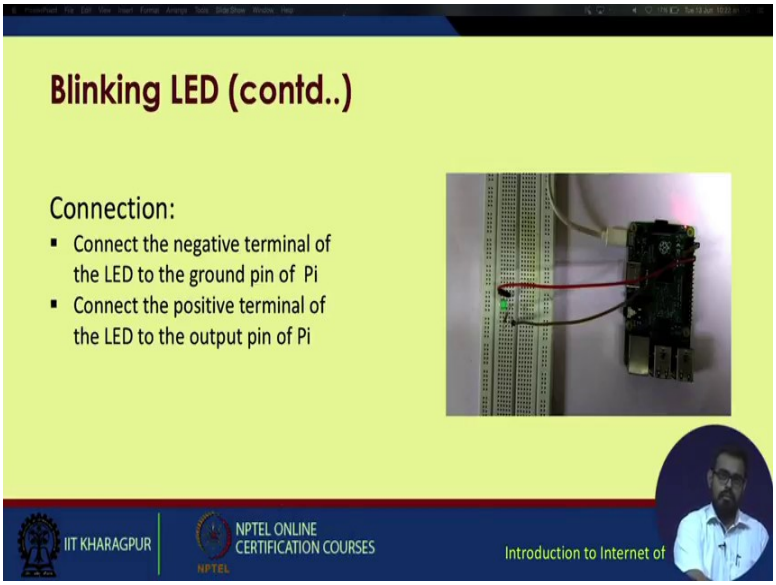
a security confirmation message, you click ok. Now, you are connect to the raspberry pi. You can just drag drop between the two file systems.

Suppose, you want to copy this folder I created this folder name IoT lets see what is inside it has three files two python files and one image file. So, just as an example I will transfer this to my desktop or better still I will transfer one file from a desktop to raspberry pi. Suppose I have got this one example1.py on my local machine and I want to transfer this to my raspberry pi. I just drag drop between the two interfaces, see file transfer was done, and now this example1.py is on my raspberry pi base system.

So, if you check again on the raspberry pi screen, now there is a folder name iot you click on it and there you have this example1.py on your raspberry pi base system. So, it is quite easy many a times programming bigger things on raspberry pi is quite cumbersome. So, I my preference is I basically program everything on my local PC and then transfer the eventual files into the raspberry pi and just do some minor configurations and changes.

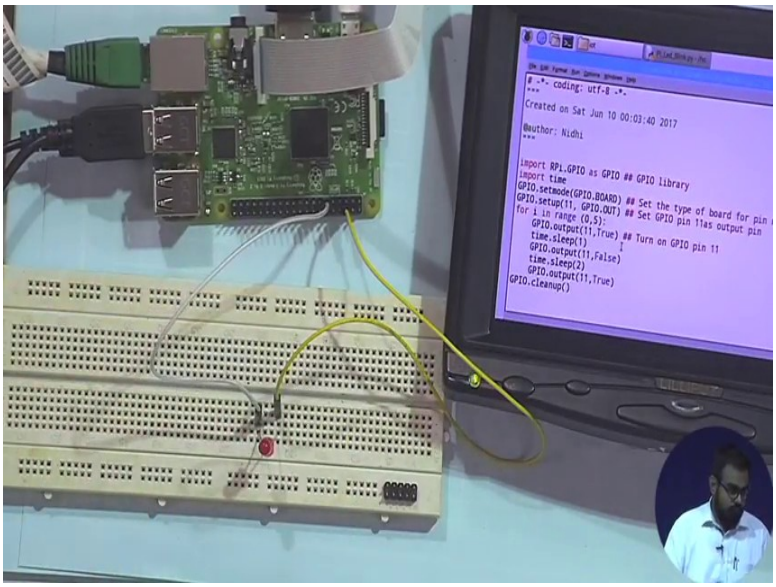
So, now coming back to the raspberry pi. So, for the blinking LED project since you are going to connected to the GPIO terminal, you will require two things first is the python-dev library and the other is the python-rpi.gpio library. So, you can easily access these two libraries over the net. So, once your pi is configured and ready to go on the terminal, you just type the commands sudo apt-get install python-dev. Once this operation is over, you type the commands sudo apt get install python-rpi.gpio.

(Refer Slide Time: 08:09)



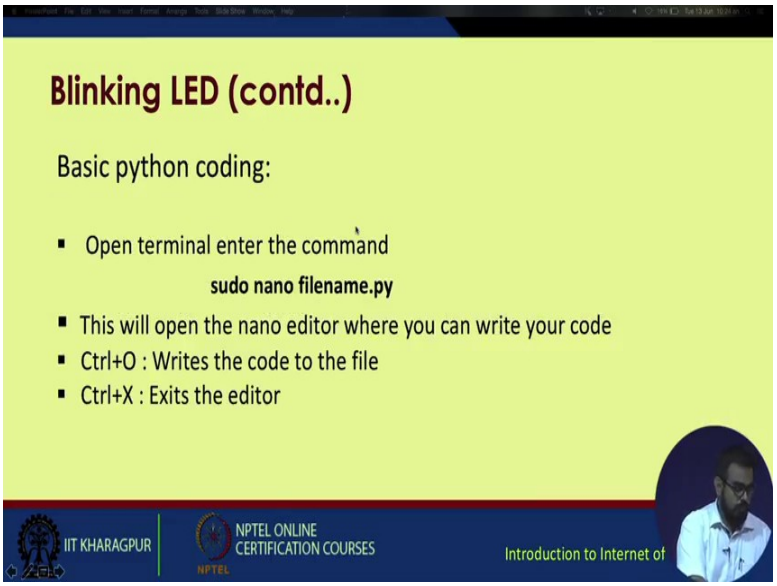
So, once these two files are installed on your raspberry pi base system, you are ready to go. So, for initial connection, I am taking an LED, I will check for the positive and negative terminals the larger one is the positive terminal and the smaller one is the negative one.

(Refer Slide Time: 08:36)



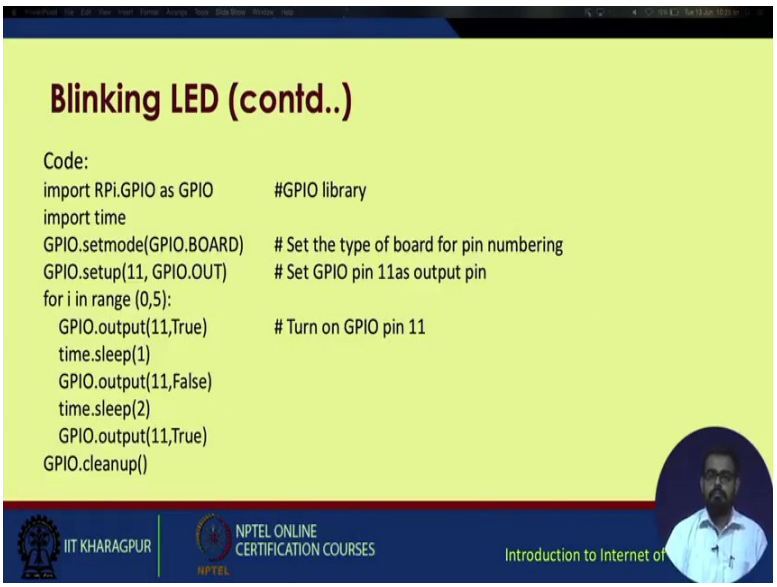
So, the white wire is connected to the positive terminal, the yellow wire is connected to the negative terminal. So, this negative terminal I will put to ground and the positive terminal for this I will have to check which port I have assigned. So, I have assigned port 11 GPIO pin 11. So, it is not quite straight forward as arduino boards for this you need to have that a reference table shown in the previous lecture, so that will give you a good reference for which pin is which GPIO pin. So, for GPIO pin 11, so actually 1, 2, 3, 4, 5, the 5th pin on the inner side no sorry is the 6th pin. So, now I have connected this. So, my connections are ready.

(Refer Slide Time: 10:07)



On the terminal I can; obviously, write my code on the terminal. So, best practice is there is in built editor on the raspberry pi in nano. So, you write sudo nano and then filename.py. So, if I want to name my file as LED blink I write LED blink.py. So, this will open a new terminal on your system on which you can write your code, python code since your file name is in the extension of your file name is dot py you have to write your code in python. And once your code is ready you press control plus O this will write the code to the file and you exit the nano editor by control pressing control plus X.

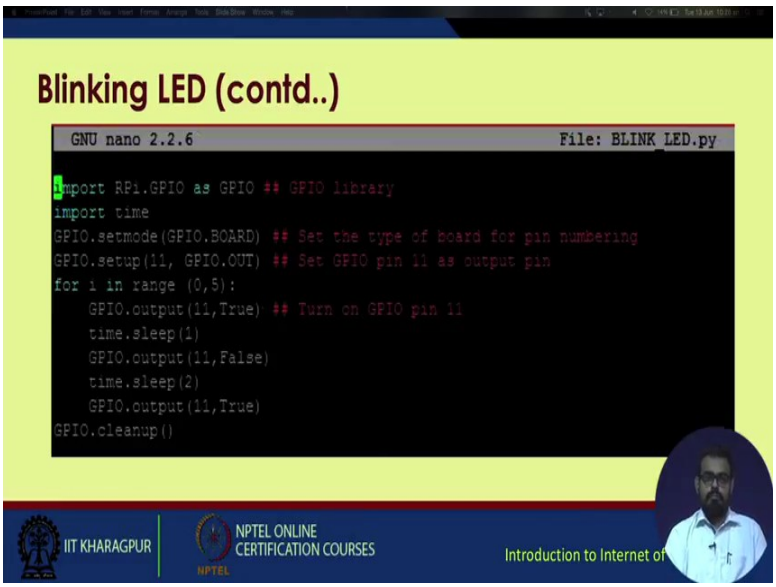
(Refer Slide Time: 11:00)



So, this is the code. So, first line is import RPi.GPIO as GPIO. So, this actually calls the GPIO library second one is import time, since you will be using a delay function on python this delay function is denoted by sleep. So, we use a time.sleep third line is GPIO.setmode. So, you put GPIO.BOARD. So, there are two options you can use it as a board mode and there is another option which we are not going to use now. So, you are going to strict to this board option in GPIO pin 11 is setup as the output pin. Then the main looping part just a simple loop for i in range 0 to 5 that means, it will iterate between 0, 1, 2, 3 and 4, 5 is because it is excluded we covered at this in the python programming part.

Now, GPIO.output pin 11 is true that means it will send a high bit to the output pin, it will sleep for 1 second and again it will send a low bit to the output pin. So, the physical output will be your once the high bit is reached, your LED will glow then it will wait for one second then once the lower bit is reached it will turn off. And again it will sleep for 2 seconds and then turn off. At the end you do this cleanup operation to release the GPIO pin.

(Refer Slide Time: 12:51)

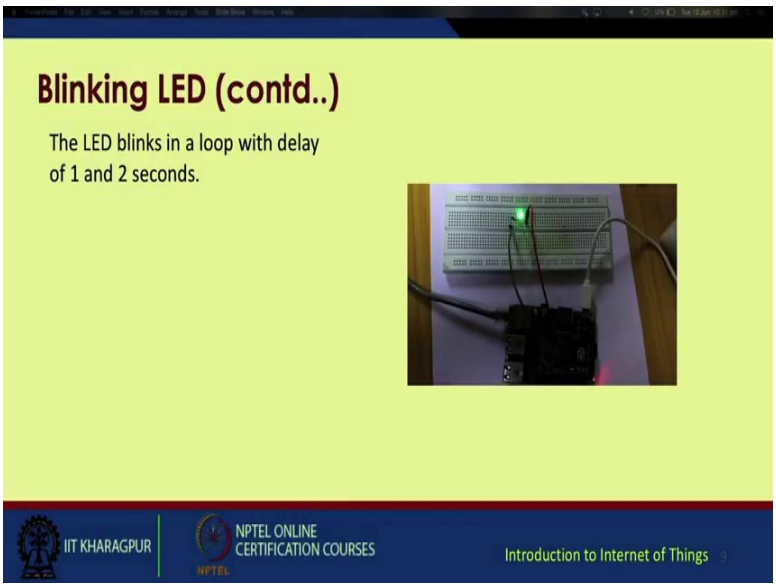


So, you have already seen on the console this operation. So, these are the codes. So, one more thing since the inbuilt python installation on raspberry pi OS are very rudimentary you would not be able to access advanced options without installing advance libraries. So, the very basic python idle interface and editor you will be able to access on raspberry pi. So, once you double click on this file this is my file of interest pi LED blink dot py, it opens up this code

right. So, let us suppose you do not have a visual display anymore let us try accessing this thing over the terminal.

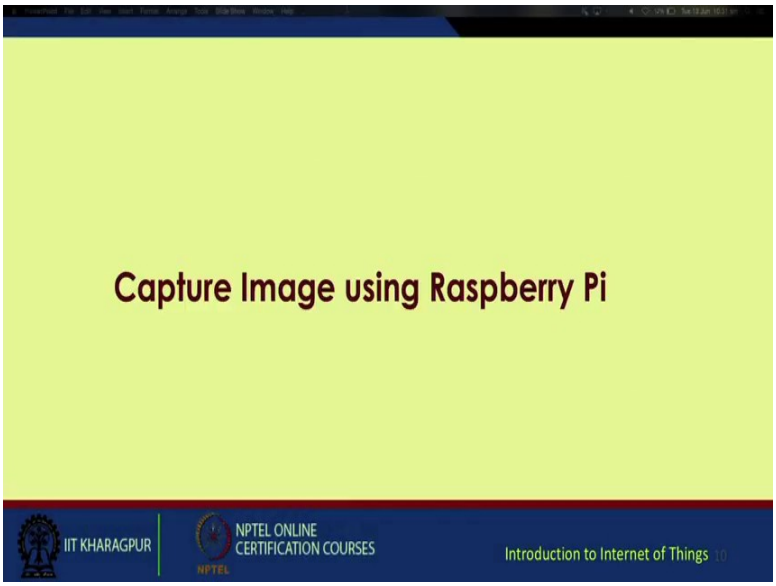
So, you login into my raspberry pi, I go to the directory IoT I am inside the directory I will check what are the files and folders inside the directory by writing ls it will list all the files and folders inside this iot directory. So, my code of interest is pi LED blink dot py. So, first let us see what is exactly inside this file I write sudo nano pi LED blink dot py. So, it opens up the editor this is the same code we showed on the slides it has already been saved. We exit this. Now, when I want to run this code I write python Pi Led_blink.py this was my Led blinking program you press enter as you can see the LED has started blinking so and it will keep on blinking. So, again run this program, I run this module, you can see LED blinks, program is executing. Now, once you get this triple arrow symbol for the console that means your program execution has finished. So, this was the basic integration and usage of GPIO pins on the raspberry pi.

(Refer Slide Time: 17:00)



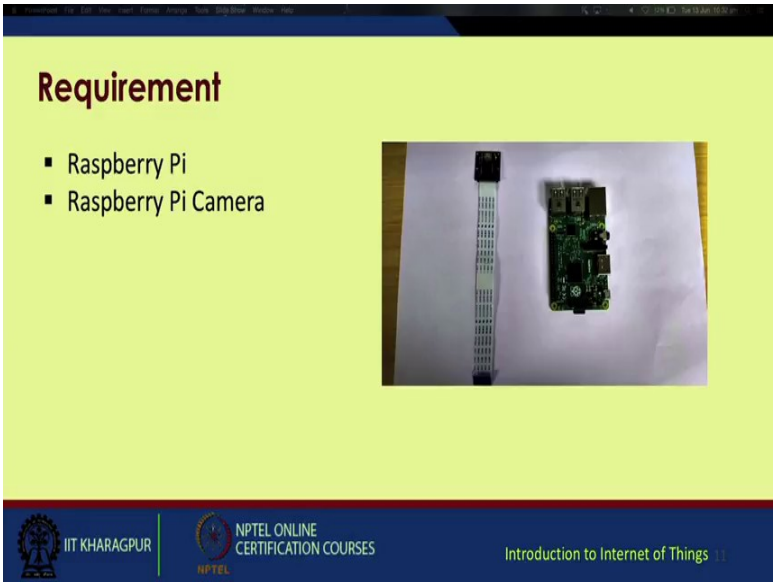
The next interesting topic you are going to cover is integration of the camera or the raspberry pi camera.

(Refer Slide Time: 17:03)



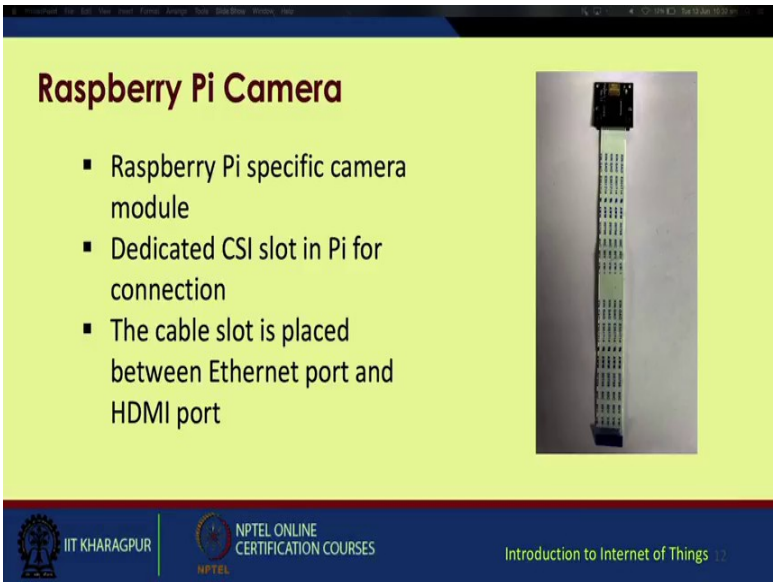
Here we are just going to show integration of raspberry pi camera.

(Refer Slide Time: 18:03)



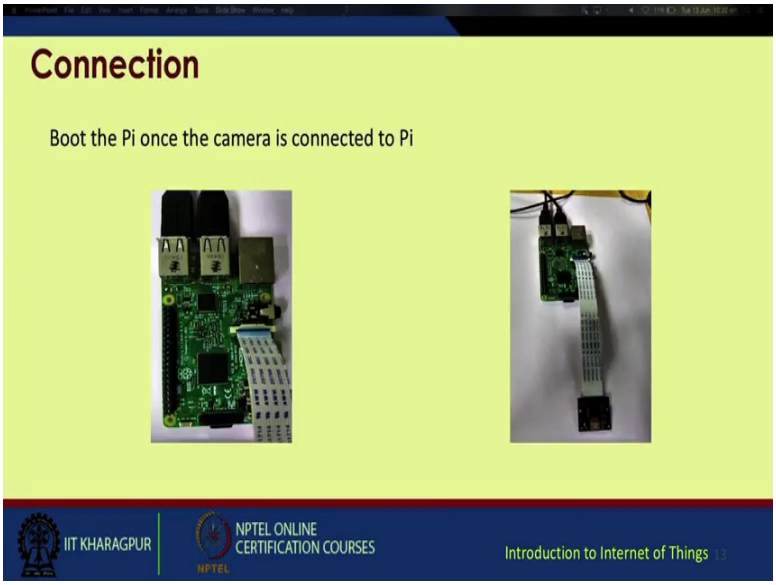
I am going to place my camera over here, my pi is booting up. Prior to this, we will cover the basics of the camera. So, you will need a pi cam and a raspberry pi.

(Refer Slide Time: 18:08)



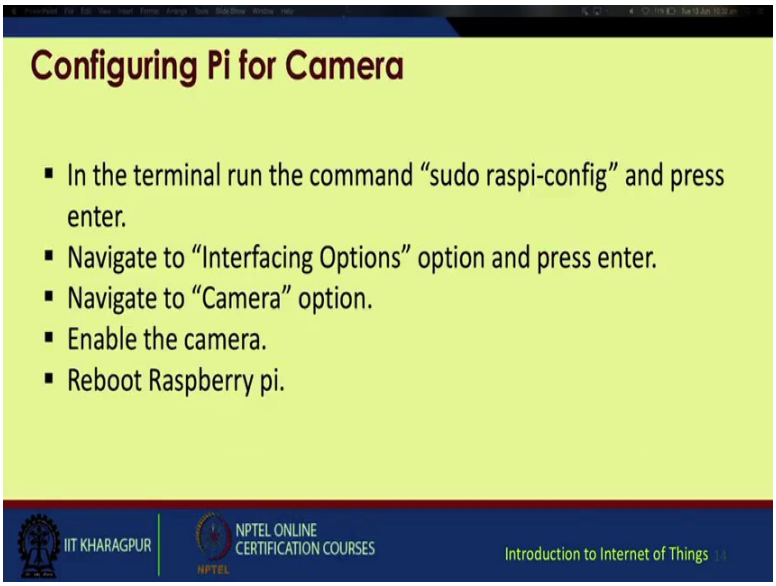
So, your raspberry pi specific camera module for on the raspberry pi you have a dedicated CSI slot for the connection and the cable slot is placed between the Ethernet port and the HDMI port told you before. The end of the ribbon cable you will find a blue tag and that has to be placed toward facing towards the Ethernet port.

(Refer Slide Time: 18:34)



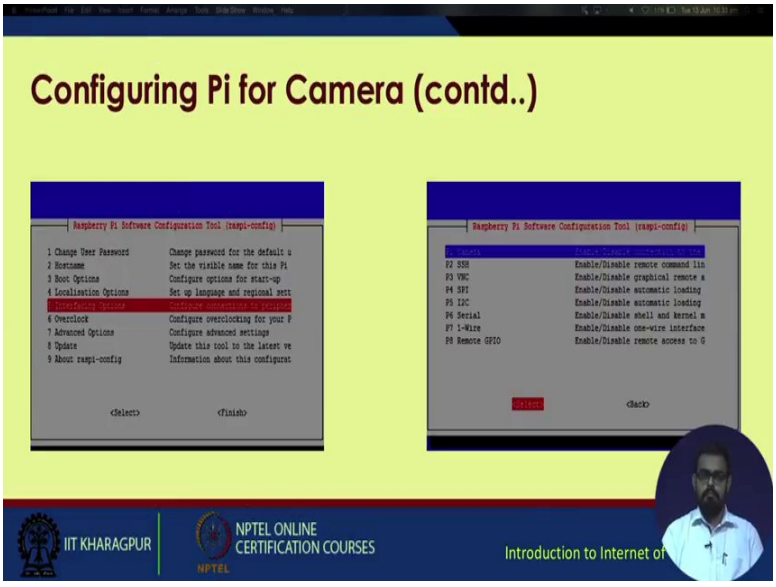
So, once the camera is connected we reboot the system. So, this reboot operation has already been done.

(Refer Slide Time: 18:43)



Now, in the terminal we run the command sudo raspi-config and press enter. So, this was covered in the first lecture on introduction to raspberry pi. I showed you there were various options one for enabling the camera another for enabling the SSH and so on. So, over there you enable the camera option and reboot your raspberry pi.

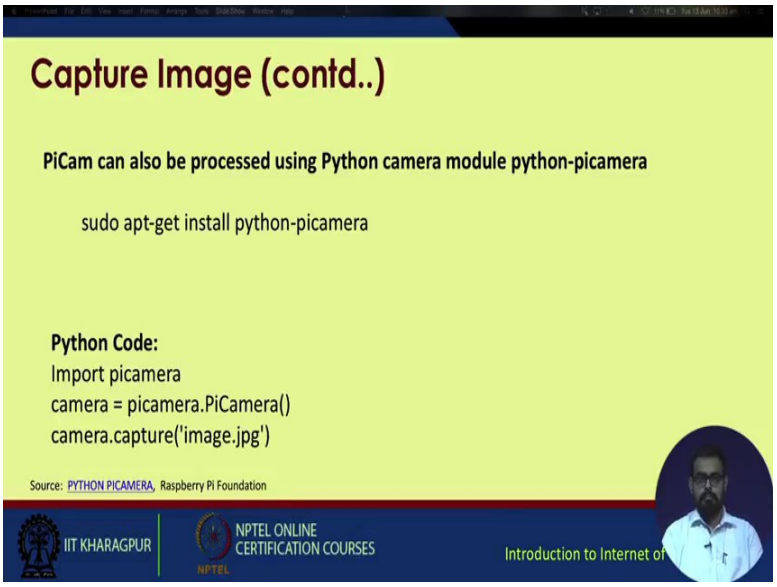
(Refer Slide Time: 19:08)



(Refer Slide Time: 19:11)

Now, once this has been done you can directly capture your images using the pi cam by writing on this terminal on this particular line of code raspistill -o space any image name.jpeg. So, this will click an image and store it as that "image name.jpg".

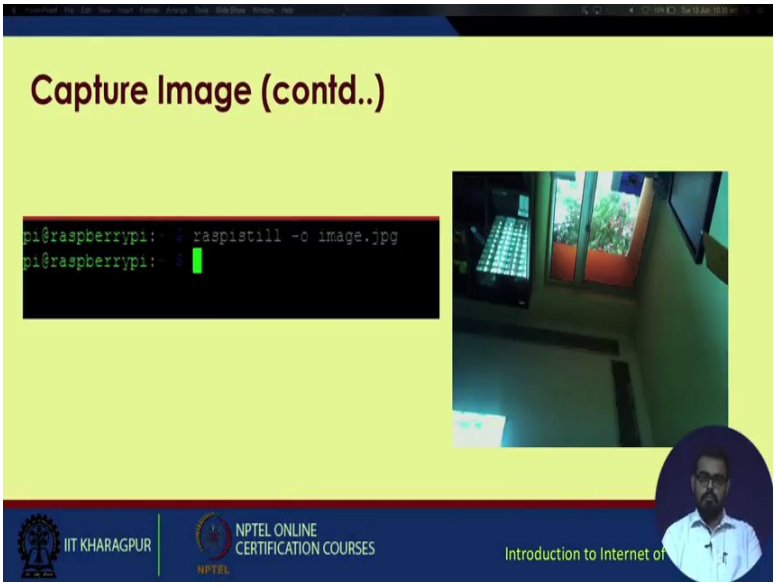
(Refer Slide Time: 19:40)



Now, this pi cam can also be integrated with python and the module particular module is known as python-picamera. So, again you will need to install this library because none of these libraries are by default available on your raspberry pi system. So, you write sudo apt-get install python-picamera. And this is the small code snippet for capturing your images

from the pi camera using a python code, it is a very short script. So, first of all you import this library import pi camera then you initiate an instance of camera by this name camera = picamera.PiCamera(). So, you must pay special attention to the way it has been written you see the first pi camera is all small then dot PiCamera P and C are capital. Once this has been initialized within camera you write camera.capture then filename.jpg. So, these tutorials are commonly available online, you can go for advanced versions options also. So, these things are commonly available on any online resource.

(Refer Slide Time: 21:03)

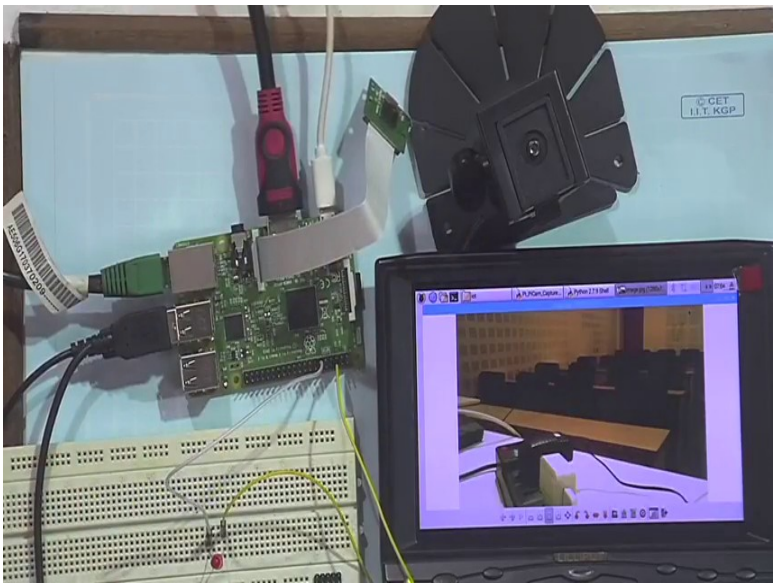


So, this is one screenshot. You write raspistill -o image.jpeg. So, your system will capture an image. So, coming back to the hardware part, so we have our camera ready. We have connected it to the network your raspberry pi to the network and it is connected to the monitor via HDMI port. Let us open the IoT folder. Now, we already have this code ready the three line snippet you saw earlier. So, this is the part of the code we are interested in. We simply run this code. Pay attention to the pi camera module, when it clicks an image then we have LED indicator going on. So, I run this module, you see this small LED indicator went off.

So, let us finish taking the image see it has taken the image of the roof. Let us try another image. We may delete this image also. Let us point the camera forward and run this code again. So, the code has executed, you see it has captured an image of the studio. So, this is just one of the basic examples of how you go about using GPIO based pins for integrating

various small sensors. And actuators with your raspberry pi as well as IO integrate larger systems with your raspberry pi more complicated devices like cameras, you can integrate keyboards mouse and so on.

(Refer Slide Time: 23:48)



I have already put up interfaced a keyboard and a mouse over here through which I am actually controlling raspberry pi interface, you can see the cursor move and using the mouse to move it. So, this is as good as a computer its additionally it is very low on power requirements as well as it is quite cheap as compared to regular computers available on the market. So, I hope you will be able to go for more complicated projects using this basic idea.

Thank you.