

- Profile of Dragit & Install Dragit ----- P1
- Use Dragit ----- P4
- Lesson 1: Move Your Car Forward ----- P4
- Lesson 2: Turn Your Car Around ----- P6
- Lesson 3: Control the Camera ----- P9
- Lesson 4: Control Your Car with Keyboard ----- P13
- Exploration ----- P16

Profile of Dragit

Install Dragit

Before any operations, you need to log into the Raspberry Pi remotely first. Here's how:

- 1 · Get the IP address of the Raspberry Pi:

<https://www.raspberrypi.org/documentation/remote-access/ip-address.md>

- 2 · Log in with the IP address, by ssh or VNC:

For ssh: <https://www.raspberrypi.org/documentation/remote-access/ssh/README.md>

For VNC: <https://www.raspberrypi.org/documentation/remote-access/vnc/README.md>

Now you're on the RPi already. Open a terminal for command lines. Let's see what next.

```

pi@raspberrypi: ~
login as: pi
pi@192.168.10.53's password:
Linux raspberrypi 3.12.25+ #700 PREEMPT Thu Jul 24 17:51:46 BST 2014 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Jul 27 14:59:17 2014 from 192.168.10.50
pi@raspberrypi ~ $

```

1. Type in the command below:

```
wget https://s3.amazonaws.com/sunfounder/Raspberry/dragit_installer.py
```

```
pi@rpi-picar-s:~$ wget https://s3.amazonaws.com/sunfounder/Raspberry/dragit_installer.py
--2017-05-11 10:52:01-- https://s3.amazonaws.com/sunfounder/Raspberry/dragit_installer.py
Resolving s3.amazonaws.com (s3.amazonaws.com)... 52.216.17.155
Connecting to s3.amazonaws.com (s3.amazonaws.com)|52.216.17.155|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10785 (11K) [text/x-python]
Saving to: 'dragit_installer.py'

dragit_installer.py      100%[=====] 10.53K  41.2KB/s  in 0.3s
2017-05-11 10:52:03 (41.2 KB/s) - 'dragit_installer.py' saved [10785/10785]
```

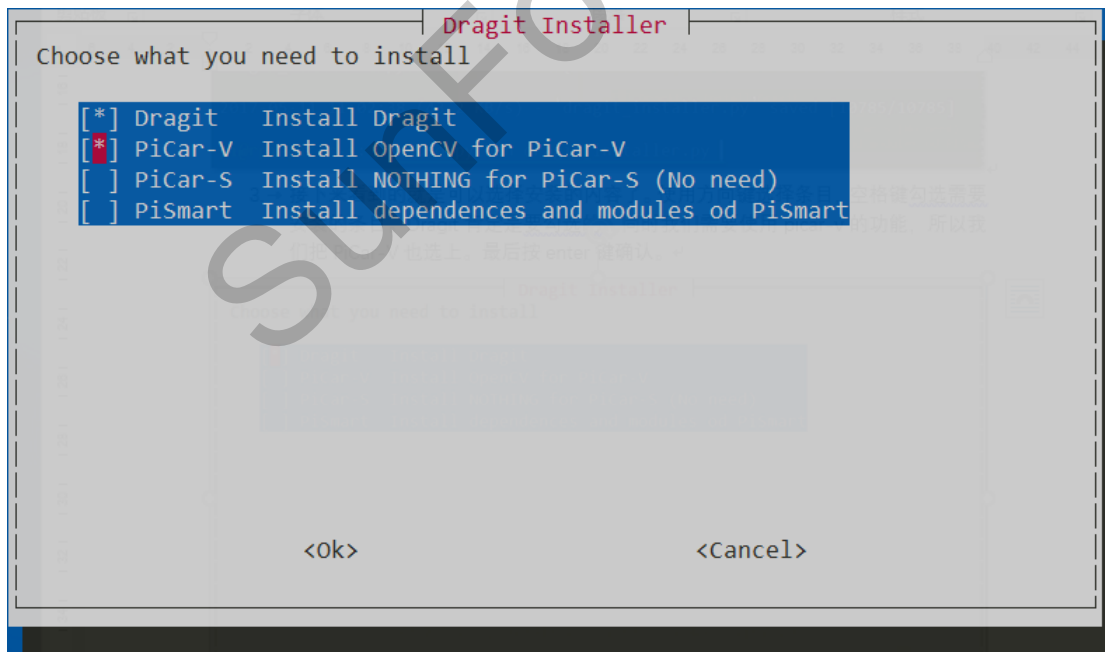
2. After download is done, type in the command to run:

```
sudo python dragit_installer.py
```

```
2017-05-11 11:02:38 (43.2 KB/s) - 'dragit_installer.py' saved [10785/10785]
pi@rpi-picar-s:~$ sudo python dragit_installer.py |
```

3. Then you can see what to install. Select the one you'd like to install by the arrow key, and confirm by Space bar.

Of course you need to select **Dragit**; meanwhile we need to use it on **PiCar-V**, so select it too. Then click OK or press Enter to confirm.



Then the program will process the installation. You need to wait for a while, and pay attention not to disconnect the network during the process.

```
Installing PiCar-V...
Reading package lists... Done
Building dependency tree
Reading state information... Done
python-opencv is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 75 not upgraded.
Done
pi@rpi-picar-s:~ $ |
```

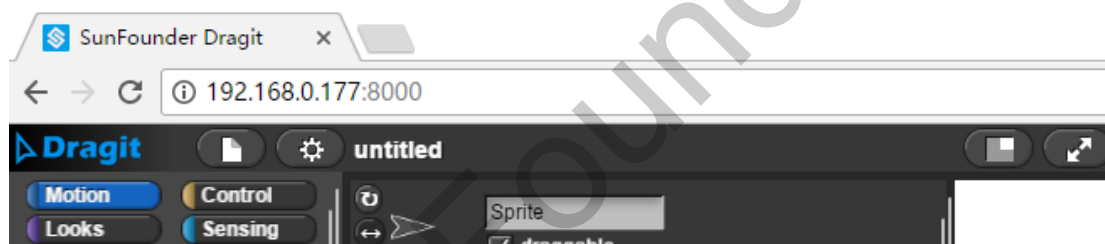
When the prompt appears as shown above, it means the installation is done.

After installation, reboot:

```
sudo reboot
```

On your computer/tablet, open a web browser (Chrome/Firefox/Safari recommended), enter the PI address of the PiCar-V and then 8000:

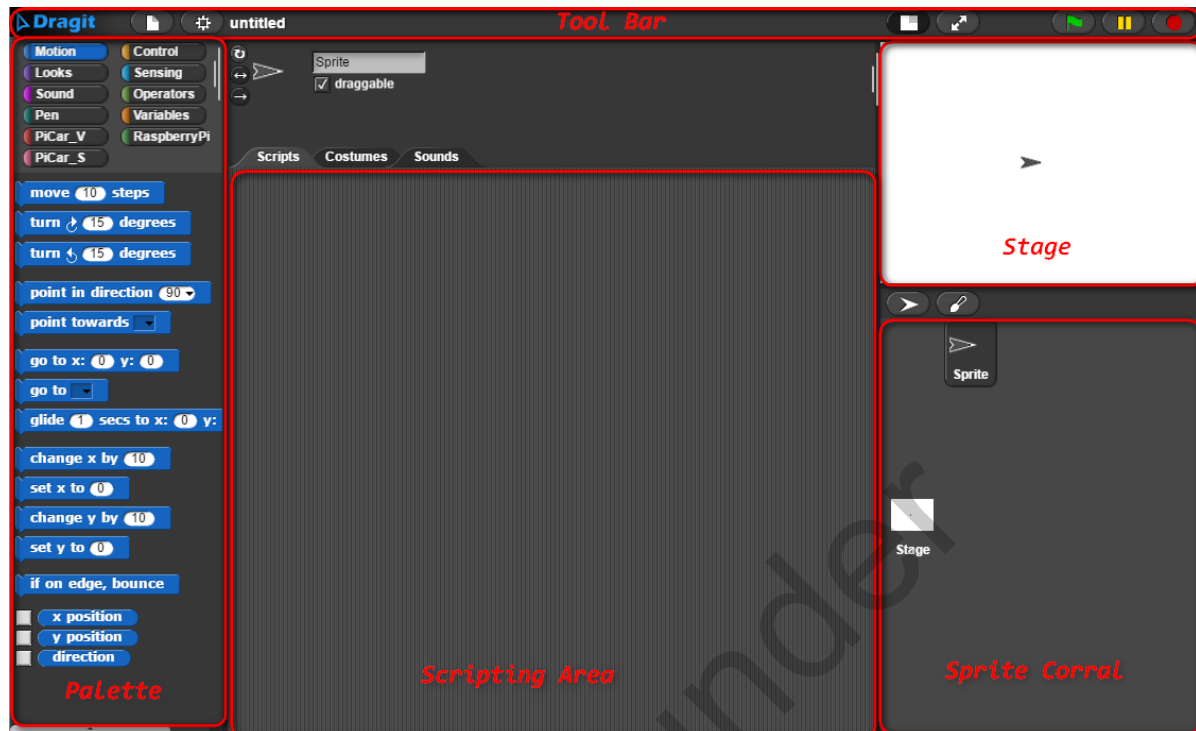
E.g. 192.168.0.177:8000



Then you're in the Dragit window.

Use Dragit

After opening Dragit, you can see the following window:



There can be one or more scripts for a project in Dragit. The scripts are shown in the scripting area and constructed by blocks in the palette.

In this tutorial for how to use Dragit for PiCar-V, we'll focus on the category of PiCar-V, basics of using Dragit, and how to control PiCar-V in this software.

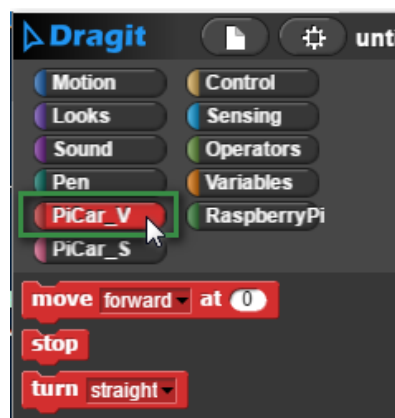
Lesson 1: Move Your Car Forward

The basic movements of the PiCar-V, just like any other car robots, are to go forward and backward. So let's start learning from its moving ahead and back.

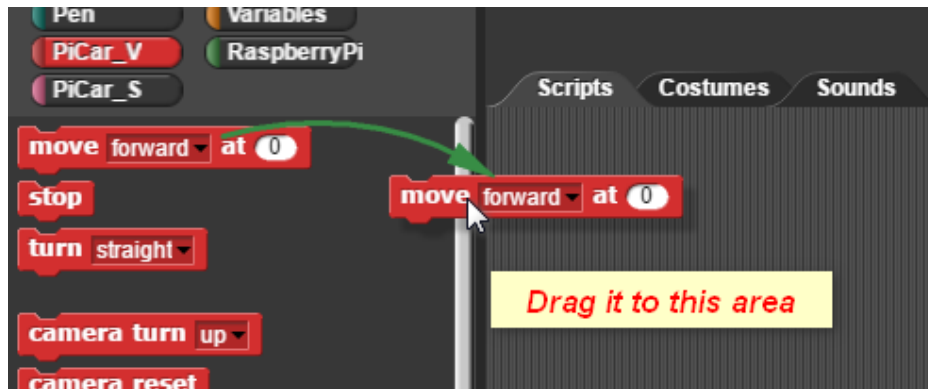
Before operation, in case the car runs out of the desk and falls off, we prop it off the surface.

And then take these steps:

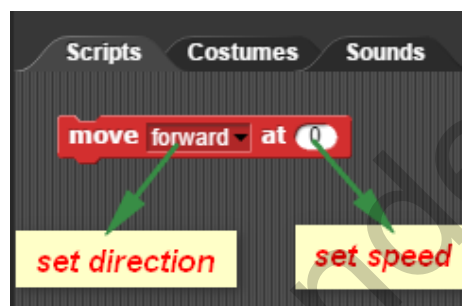
- 1 · In the **palette** area, click to select the **PiCar_V** category.



2 · Then click and drag **move forward** at 0 into the **scripting** area.



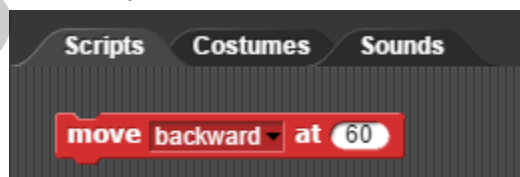
3 · So now in the scripting area there's already a block to control the robot to run forward and backward. Let's check how it is done with the block!



In this block, there are two parameters for control: **direction** and **speed**.

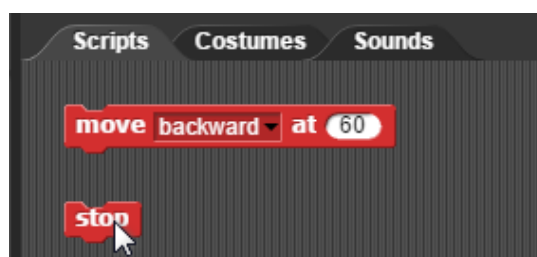
Click the input of direction by the **triangle** sign and select a value at the drop-down list, and then click the speed and enter a number ranging between 0 and 100. Try 60 for example. So now your block includes the speed and direction which control the car to move.

Click this block and see what will happen to the car. Yes! The car's back wheels start to spin and it will run on the table if you take it down.



Click the direction input and select backward, click the block again, and the back wheels will spin reversely, so the car will go backward if you put it on the table.

OK, now the robot may get tired after running and you can get it to stop. Click and drag **stop** to the scripting area and click the block. Good, have a good rest.



Special Notes

Generally your PiCar-V will move as you control. But, something will possibly go wrong. For example, when you set it to move forward, it goes backward; input backward, it may go forward. Or, even unexpectedly, the two back wheels spin in different directions...

Chillax! You can calibrate them right with these blocks: **set left wheel forward to 0** and **set right wheel forward to 0**.

If the left back wheel goes the wrong direction, drag **set left wheel forward to 0** into the scripting area, change the input, and click the block; if the right wheel is not functioning well, drag **set right wheel forward to 0** similarly; just drag them both if two back wheels need to correct.

For example, I need to change my left back wheel, I'll drag **set left wheel forward to 0**, set input to 1, click the block and the wheel will reverse the spinning. If I happen to dislike the adjustment (though I don't know why LoL), I'll click the block and the direction will change back again.



So now it's your turn to adjust the spinning direction of your car wheels. Just try!

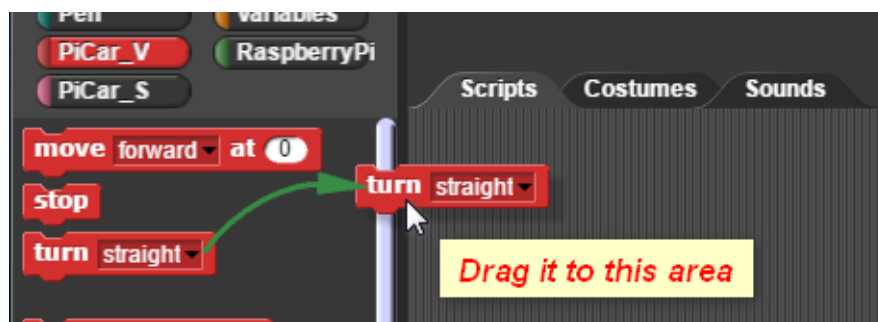
Lesson 2: Turn Your Car Around

Now after the Lesson 1, you should be able to control the car to go forward and back and make it stop. Then, you may wonder: how can it turn around?

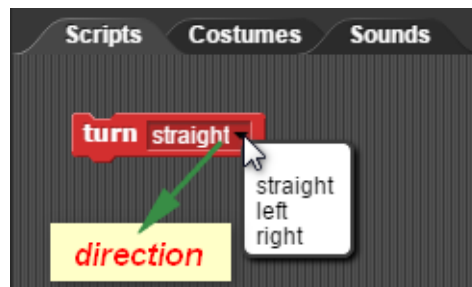
So, next we'll move on to turning of the car.

As before, prop the car and leave its wheels off the ground in case of running out of the table. For its turning around, the block **turn straight** will be used. Very similar steps as in Lesson 1.

- 1 · Click **PiCar_V** in the palette area. Or you can just stay there if you're already there.
- 2 · Click and drag **turn straight** into the scripting area.



- 3 · Now in the scripting area there's a block to control the car to turn. Let's check its details.



Click the triangle sign in the block and you can see the values on the drop-down list. Just pick an input to make the car turn in that direction.

For example, select left. Click the block and observe the car.

Yes, the car's front wheels will turn left accordingly, well, because we just made them do.

In the same way, change the direction to straight and right and click the block to run. Then, how are the front wheels going? Pretty easy, right?

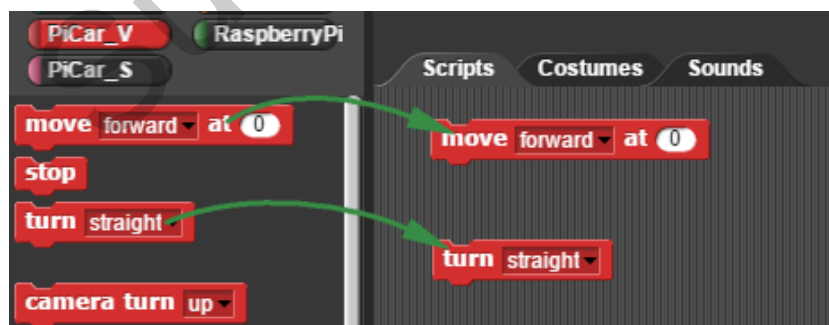
Hm, that may not be what you really want because the wheels just keep turning. You want the car to turn a direction when it's running forward or backward? Of course that's easy to make! No need to rush and we're just about to talk about making it by combining two or more blocks.

❖ Combined Blocks

To make the car turn left or right when it's running, we need the car to go forward/backward + turn left/right/straight... which means to make the blocks in Lesson 1 and 2 work at the same time.

So, how to combine these two?

- 1 · Drag **move forward at 0** and **turn straight** blocks into the scripting area.



- 2 · Drag **turn straight** under **move forward at 0**, approaching until a white line appears between them.

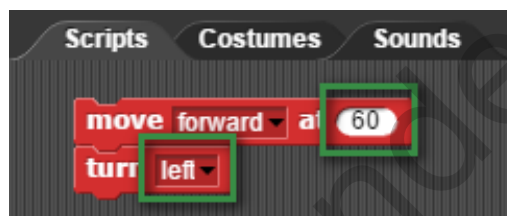


- 3 · Release the block and you'll see the two are combined!



To run the combined block, just as to do a single one, simply click it - *you can take it as a whole, just a bigger block.*

- 4 · Try to change the inputs in this larger block, click it, and see what happens. As shown in the figure below, as set to "move forward" at "60" speed and "turn left", so car is running slightly fast when turning left.



Or, change "left" to "straight" and click the block again. Then the car will go straight.



Things can be better if you have other ideas. If I do not feel like combining them, I can just split the two blocks by dragging the lower one to somewhere else, like this:



Special Notes:

When the car turns "straight", the front wheels are almost oriented to the front. But some minor errors may exist there which can be neglected. However, if it obviously deviates from

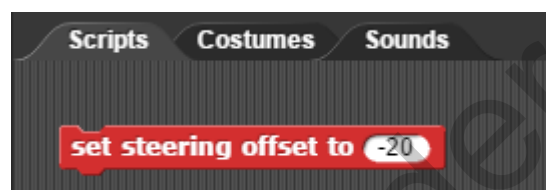
the right track, you can adjust by **set steering offset to 0**.

- 1 · Drag **set steering offset to 0** into the scripting area.



- 2 · Input a value in the block.

If the front wheels deviate to the left, input a **positive** value, like 20; if to the right, give a **negative** one like -20 as shown below.



- 3 · Then click the block and check the front wheels. Still deviations? Try to increase or decrease the value then, based on whether the angle is reduced or over-adjusted compared to the previous status. Just keep calibrating it till you feel it's good.

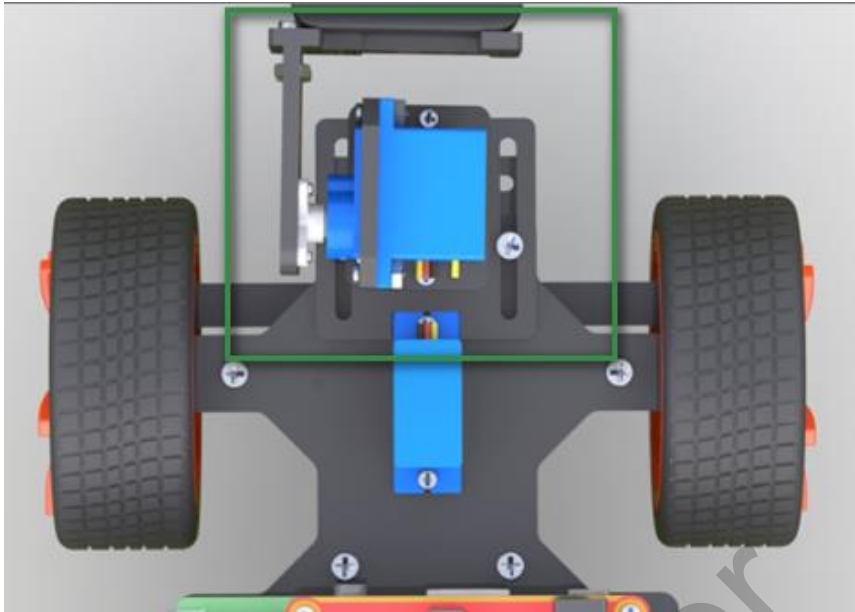
Lesson 3: Control the Camera

Through Lesson 1 and 2, now I think you can make the car run as you want easily. So those are just basic control of the robot via the Dragit. But definitely that's not all you can learn! To move on, let's check using the software to control the camera on the car. The **pan-&-tilt** structure enables the car to film at different angles, just like a naïve kids looking around 360 degrees curiously for new things in the wonderful world.

So in this lesson, let's try the pan&tilt.

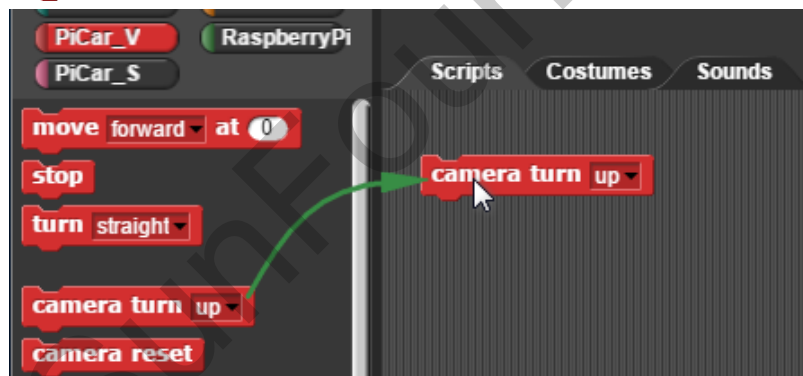
Well, here's some difference from the lessons before. First we need to offset the pan&tilt, by the **set camera pan offset to 0** and **set camera tilt offset to 0**. The steps are just similar to what you've been through previously with **set steering offset to 0**. So, no more explanation.

And here's how we'd like the pan&tilt to be:



After the adjustment is done, you can start dragging the block to control the pan&tilt. Try to make the camera get a glance now!

1 · Find and drag **camera turn up** into the scripting area:



2 · Pick a direction, like, "up":

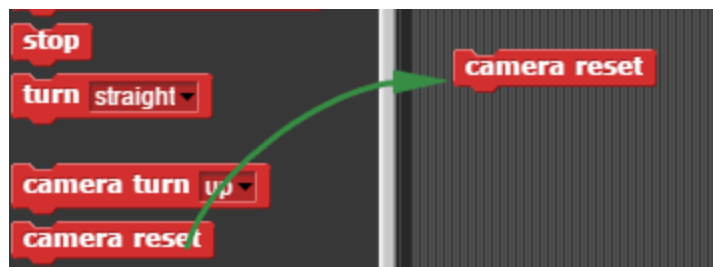


3 · Click the block and check how the camera is moving. Yep, it "looks" up.

Then try changing a direction. E.g., give it a "down" value and see what will happen. Bingo! The camera will just "looks" down as you told it to.

Have more trials for each input. Now, the camera may be "looking" up and down, or "looking" about you and other stuff!

4 · Well, if you want to restore the original position of the cam, try this block: **camera reset**



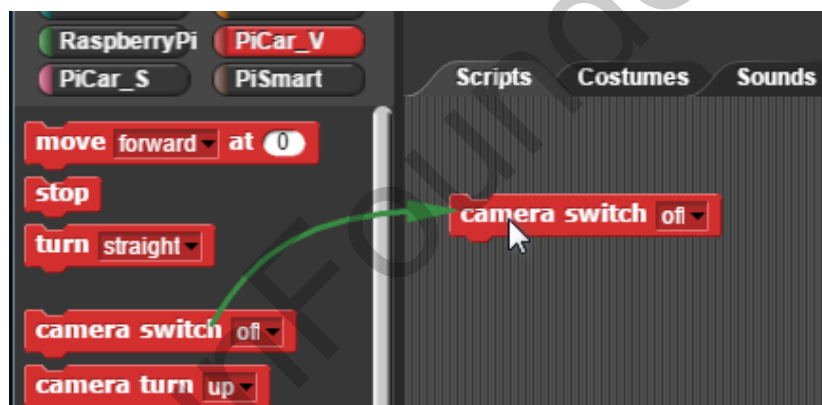
Click the block and everything, I mean, the camera, will return to its looking at first.

❖ Turning ON/OFF Cam

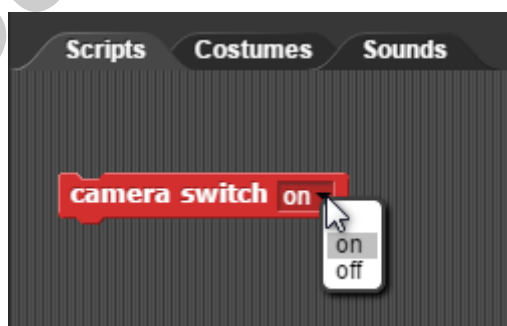
PiCar-V can transfer the images taken by the camera to its server, meaning when you turn on the camera, you can visit it on a web page and view what it "sees" also.

Here use the block **camera switch off** for camera power up and down.

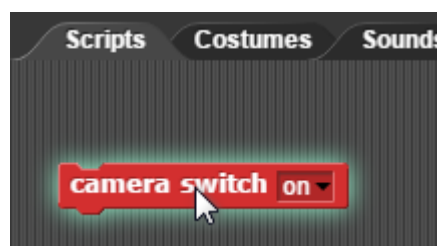
1 · Drag **camera switch off** into the scripting area.



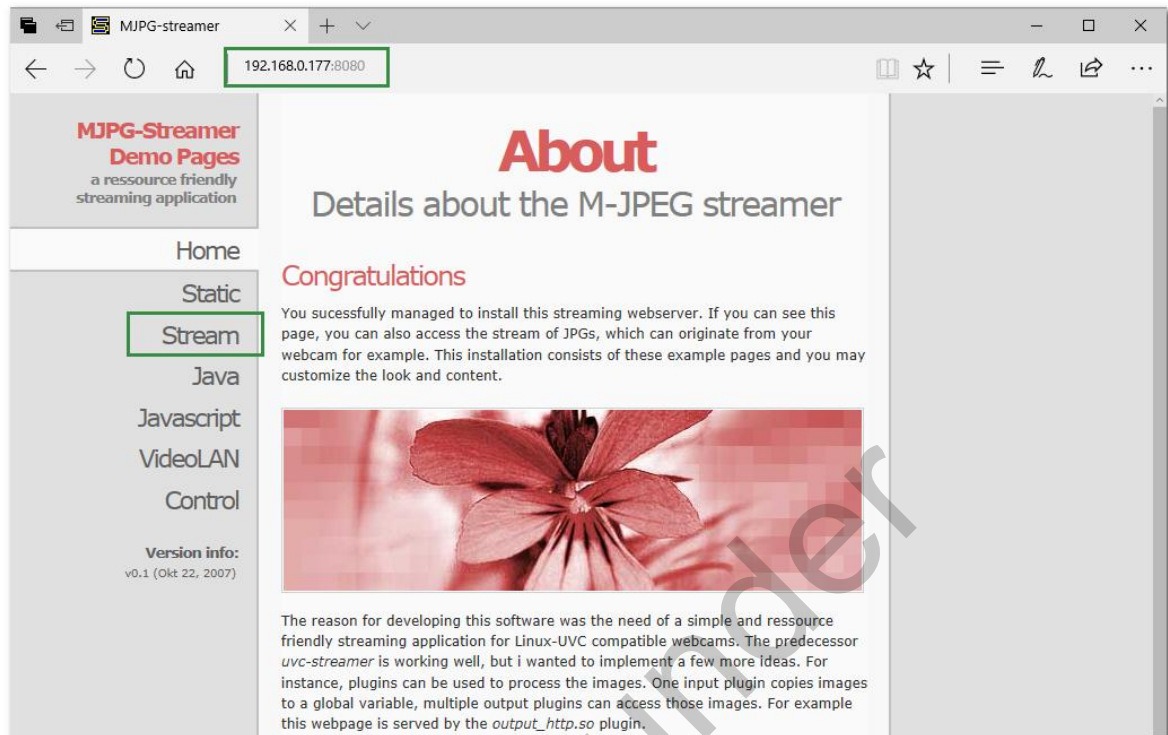
2 · Select **on** from the drop-down list.



3 · Click the block to start the service of the camera.



- After the camera is in use, open a web browser to visit it. Enter IP address of your PiCar-V in the address bar, with the port number 8080. For example, mine is 192.168.0.177, so I enter 192.168.0.177:8080.



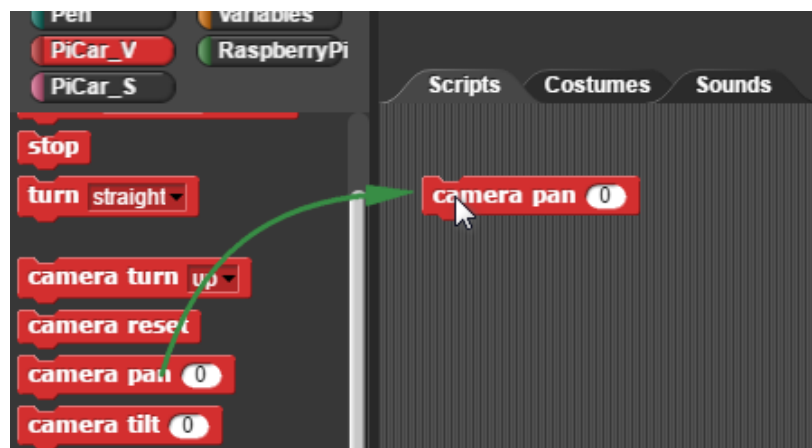
Now the camera is in service normally, and you can see the page above. Click **Stream** and you'll see what the camera is filming.

If you don't need the camera open, just select off in the block of camera. Then the page cannot be refreshed.

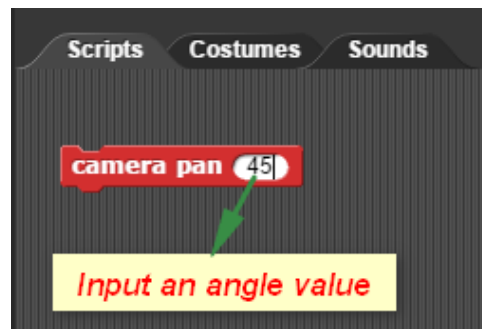
Special Notes

In normal cases, you can let the camera turn in each directions just by **camera turn up**. Yet, the angle in this block may not meet all your needs. Good news is that we prepare another two blocks for you to fine tune the angle: **camera pan** and **camera tilt**.

- Drag **camera pan** into scripting area.



- 2 · Input a value in the block, indicating the angle you want the pan to turn by. For instance, input 45:



Click the block and check the turning of the camera. See, it turns to about 45 degrees at the **left**.

Or, try another value. Let's say, a negative one -45. Again click the block and the camera should turn to the same degrees position at **right**.

Why not try your own input? Click the block every time you change the value and it's interesting to see the change come true each clicking!

- 3 · **camera tilt 0** is similar to the **camera pan 0**. Just enter an angle, click the block, and the camera will turn around as you want.

If you enter a positive value, the camera will "look" up; a negative one, it "bends" the head down.

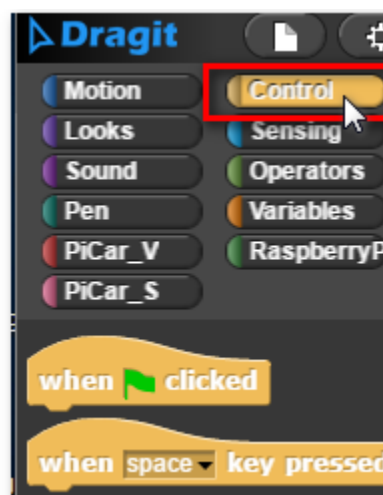
Lesson 4: Control Your Car with Keyboard

OK then. Here we are - done learning the control of each part on the PiCar-V robot. It's easy and simple to do by just dragging and clicking the block to run the code. Hold on, a little bit boring? Don't go crazy - a keyboard can save you!

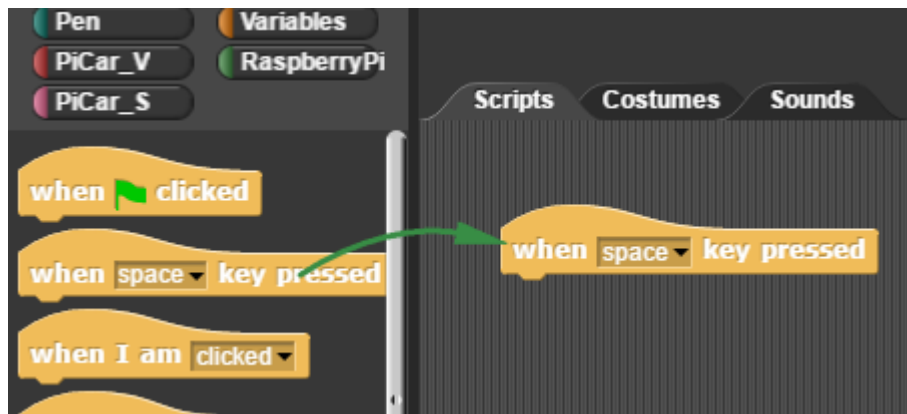
To control the car with keyboard, we need this block: **when space key pressed**

Similarly before the operations, prop the car with wheels off the ground.

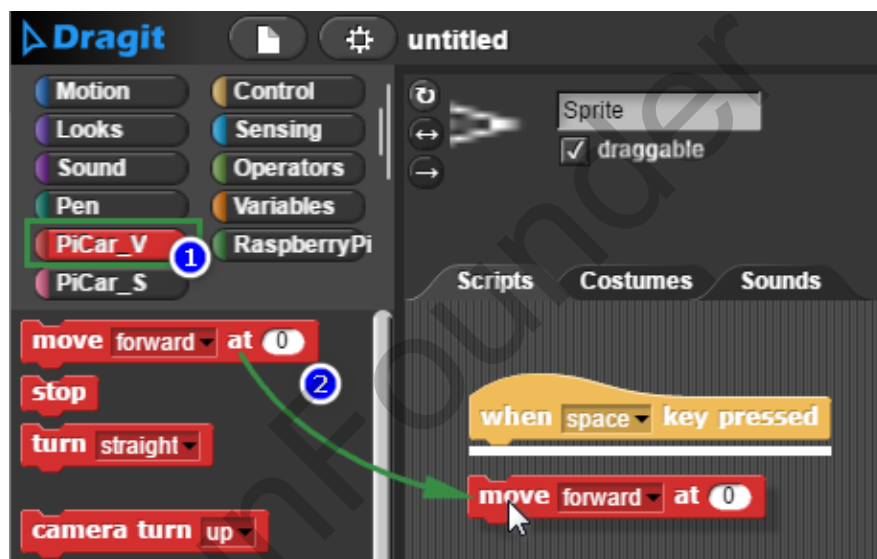
- 1 · In the palette area, click the block **Control**.



- 2 · Click **when space key pressed** and drag it into the scripting area.




- 3 · In the palette area, click on PiCar_V, click and drag the "move" block to under the "when" block and combine the two.



- 4 · Click the sign of "when" and select "w" on the drop-down list, enter a value, like 60, for speed.



- 5 · Ok, now let's try the control by keyboard. Press the key , and the car's back wheels are spinning forward. If you put the car down on the table, it will just run away quickly!

6 · Try more keys based on the table below:

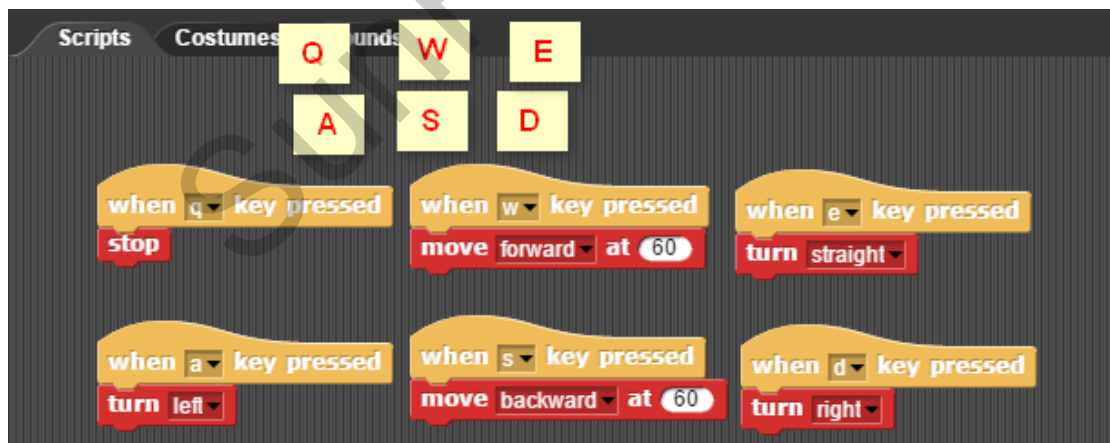
Key	Car Action
W	Forward
S	Backward
A	Turn left
D	Turn right
Q	Stop
E	Straight

Or you can try another simpler way: copy the block:

Right click on the top block, namely **when space key pressed**, select **duplicate** on the drop-down list, and paste this combined block anywhere you like.



After fixing the block, change the key value in the "when key pressed" block, direction in the "move" and "turn" blocks, so it should be like this:



Now you should be able to control the car to go forward/back, turn left/right and go straight easily by the w, s, a, d, q, and e keys. Sounds as cool as playing a game. Have a try!

Wow, you're control the car as snug as a bug in a rug then. Great job!

Exploration

So, that's all for this tutorial. Bet you have learnt a great lot!

How about making your code? All you need to do is use your imagination and make the program with Dragit - much hopefully more than we have introduced!

Some tips:

If you feel like it, try to build the code for this function: add the control of the cam with the five keys mentioned in Lesson 4.

Or, try to use variables and then control the speed change by keyboard.

Welcome to share with us your ideas on our [FORUM](#).

SunFounder