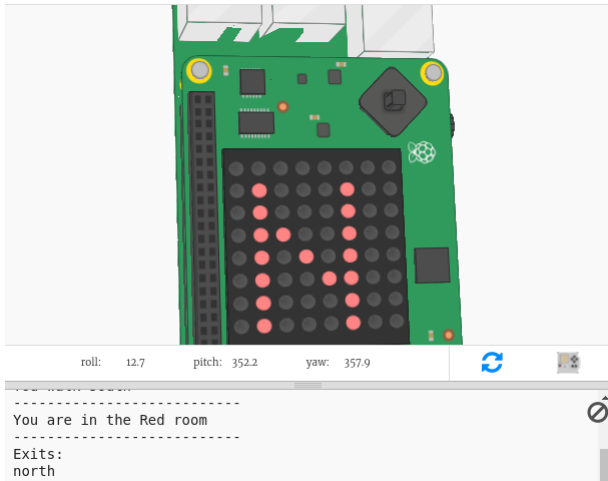




Compass Maze

Introduction:

In this project you will use the the Sense HAT as a compass to navigate out of a maze of colourful rooms. You will need to point the Sense HAT in the direction you want to move and then press the button in the middle of the joystick to make a move.



To play the game press Run and read the text that appears in the trinket output window.

Your current compass direction will appear on the Sense HAT display (N, S, E or W.) You can change direction by moving the Sense HAT in the emulator.

When you are facing in the direction you want to go, press the middle button on the joystick by pressing enter on the keyboard.

Additional information for club leaders

If you need to print this project, please use the Printer friendly version (<https://projects.raspberrypi.org/en/projects/compass-maze/print>).

Club leader notes

Introduction:

In this project, children will learn how to use the Sense HAT magnetometer (compass), and use the compass direction to navigate through a maze.

Online Resources

This project uses Python 3. We recommend using Trinket (<https://trinket.io/>) to write Python online. This project contains the following Trinkets:

- 'Compass Maze' Starter Trinket – jumpto.cc/compass-go (<http://jumpto.cc/compass-go>)

There is also a trinket containing the completed project:

- 'Compass Maze' Finished – trinket.io/python/d11bf21615 (<https://trinket.io/python/d11bf21615>)

Offline Resources

This project can also be completed offline (<https://www.codeclubprojects.org/en-GB/https://projects-static.raspberrypi.org/projects/compass-maze/07a6b994556711d71ee399d8a6d17dfd5dd7dab4/en/resources/physical-sense-hat/>) on a Raspberry Pi computer with a Sense HAT. You can access the project resources by clicking the

'Project Materials' link for this project. This link contains a 'Project Resources' section, which includes resources that children will need to complete this project offline. Make sure that each child has access to a copy of these resources. This section includes the following files:

- [compass-maze/main.py](#)
- [compass-maze/maze.py](#)

You can also find a completed version of this project in the 'Volunteer Resources' section, which contains:

- [compass-maze-finished/main.py](#)
- [compass-maze-finished/maze.py](#)

(All of the resources above are also downloadable as project and volunteer .zip files.)

Learning Objectives

- How to use the Sense HAT magnetometer (`get_compass()`)

This project covers elements from the following strands of the Raspberry Pi Digital Making Curriculum (<http://rpf.io/curriculum>):

- Combine programming constructs to solve a problem.
(<https://www.raspberrypi.org/curriculum/programming/builder>)

Challenges

- "Reward the player" - Displaying an image on the LEDs at the end of the game;
- "Create your own maze" - edit the maze dictionary to create their own maze.

Project materials

Project resources

- .zip file containing all project resources (<https://projects-static.raspberrypi.org/projects/compass-maze/07a6b994556711d71ee399d8a6d17dfd5dd7dab4/en/resources/compass-maze-project-resources.zip>)
- Compass Maze starter project (<http://jumpton.cc/compass-go>)
- Offline starter Python file (<https://projects-static.raspberrypi.org/projects/compass-maze/07a6b994556711d71ee399d8a6d17dfd5dd7dab4/en/resources/compass-maze-main.py>)
- Offline starter Python file containing the maze code (<https://projects-static.raspberrypi.org/projects/compass-maze/07a6b994556711d71ee399d8a6d17dfd5dd7dab4/en/resources/compass-maze-maze.py>)

Club leader resources

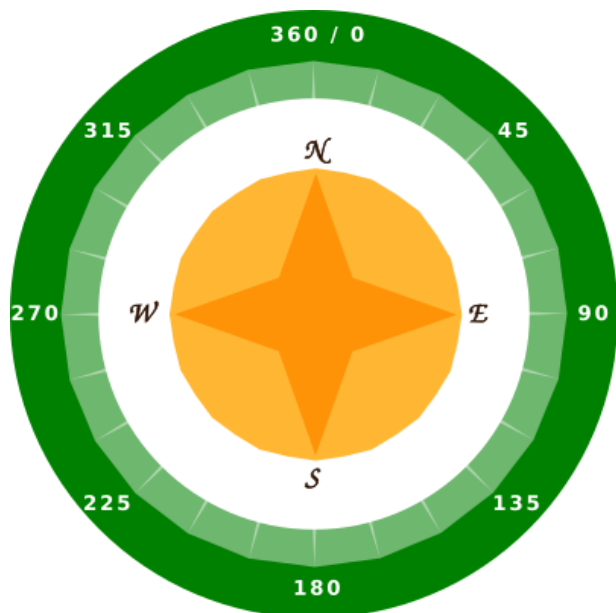
- .zip file containing all completed project resources (<https://projects-static.raspberrypi.org/projects/compass-maze/07a6b994556711d71ee399d8a6d17dfd5dd7dab4/en/resources/compass-maze-volunteer-resources.zip>)
- Online completed Trinket Compass Maze project (<https://trinket.io/python/0c8cdacd70>)
- [compass-maze-finished/main.py](#) (<https://projects-static.raspberrypi.org/projects/compass-maze/07a6b994556711d71ee399d8a6d17dfd5dd7dab4/en/resources/compass-maze-finished-main.py>)
- [compass-maze-finished/maze.py](#) (<https://projects-static.raspberrypi.org/projects/compass-maze/07a6b994556711d71ee399d8a6d17dfd5dd7dab4/en/resources/compass-maze-finished-maze.py>)

Finding the compass direction

The Sense HAT contains a magnetometer that can be used to work out which direction is North.

In the emulator North corresponds to the top of your screen. The Sense HAT reports a compass heading in degrees from North.

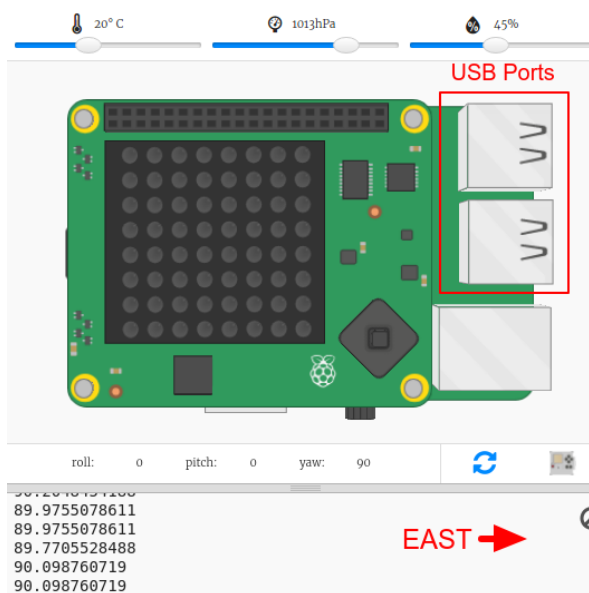
Here's a reminder of the points of a compass:



- Open the Compass Maze Starter Trinket: jumpto.cc/compass-go (<http://jumpto.cc/compass-go>).
- Let's find out which direction the Sense HAT is pointing in. Add the following code to the bottom of `main.py`:

```
while True:
    print(sense.get_compass())
```

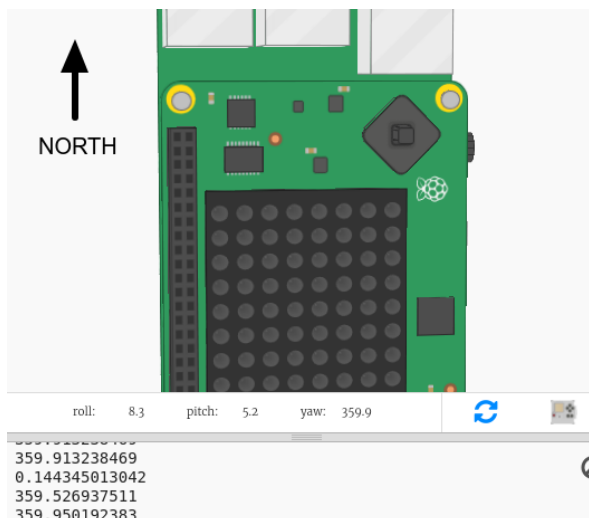
- Run your code to see the compass heading - how many degrees you are from facing north.



In its starting position the Sense HAT is facing east and you should see values of about 90 degrees.

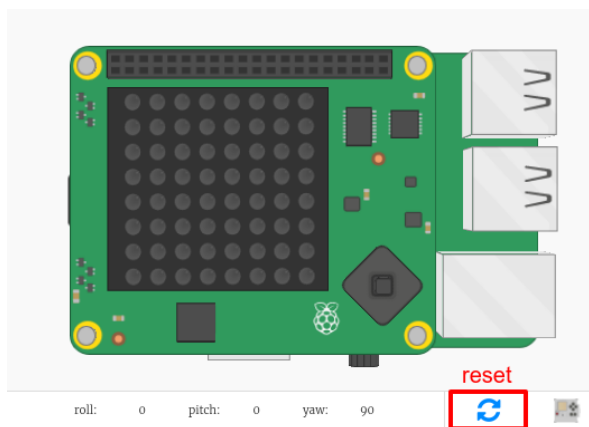
The direction is based on the USB ports.

- Drag the Sense HAT around to change its direction.



Try finding different directions:

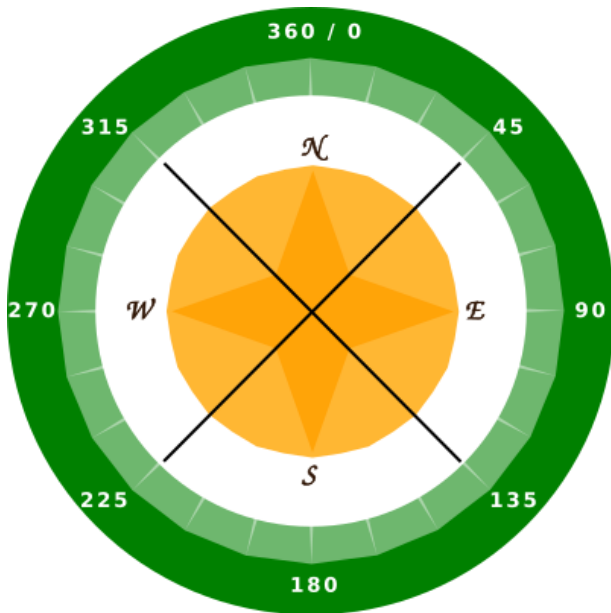
- North: Around 360 or 0 degrees
- East: Around 90 degrees
- South: Around 180 degrees
- West: Around 270 degrees
- If you get in a muddle you can always click the reset button to put the Sense HAT back into its starting position.



Showing the compass direction

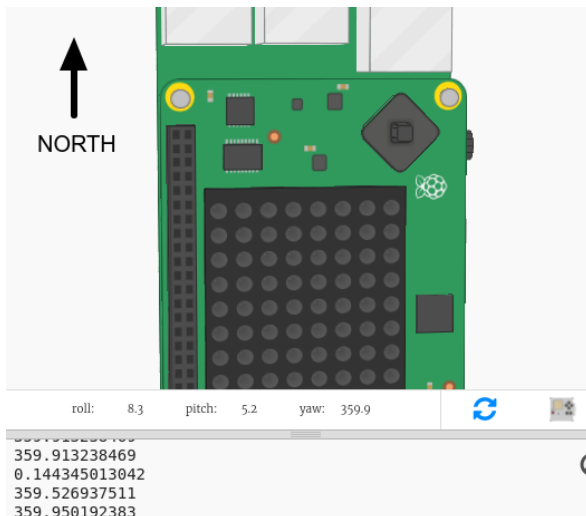
Next let's show the compass direction, N, E, S or W, on the Sense HAT screen.

If the compass heading in degrees is between 315 and 45 then the Sense HAT is pointing North and you want to display an 'N'. If the heading is between 45 and 315 then you want to display an 'E' and so on.



- First let's show an N on the screen when the Sense HAT is facing north.

Remember that when the Sense HAT is facing North the USB ports are at the top:



- Change your code to display an 'N' when the compass heading is between 45 and 135:

```
while True:
    heading = sense.get_compass()
    if heading < 45 or heading > 315:
        sense.show_letter('N')
```

- Drag the Sense HAT to North (USB ports at the top of the screen) to test your compass.



The 'N' won't disappear, you need to add code for the other directions.

- Hmm, the 'N' is sideways. It would make more sense to have the letter facing in the same direction as the USB ports.

Add the following code to rotate the Sense HAT display.

```
sense = SenseHat()
sense.clear()

sense.set_rotation(90)

while True:
    heading = sense.get_compass()
    if heading < 45 or heading > 315:
        sense.show_letter('N')
```

Now the compass letter will be lined up with the USB ports which makes more sense when using the Sense HAT as a compass.

- Now let's show an E on the screen when the Sense HAT is facing east. If you're not facing north then the heading must be more than 45 degrees so you can just check that it's less than 315:

```
while True:
    heading = sense.get_compass()
    if heading < 45 or heading > 315:
        sense.show_letter('N')
    elif heading < 135:
        sense.show_letter('E')
```

- Add the code for south. Look at the compass to work out what the condition needs to be.
- Your code should look like this:

```
while True:
    heading = sense.get_compass()
    if heading < 45 or heading > 315:
        sense.show_letter('N')
    elif heading < 135:
        sense.show_letter('E')
    elif heading < 225:
        sense.show_letter('S')
```

- Now add the code for west. If it's not north, east or south then it must be west! You can just use an 'else'.

```
while True:
    heading = sense.get_compass()
    if heading < 45 or heading > 315:
        sense.show_letter('N')
    elif heading < 135:
        sense.show_letter('E')
    elif heading < 225:
        sense.show_letter('S')
    else:
        sense.show_letter('W')
```

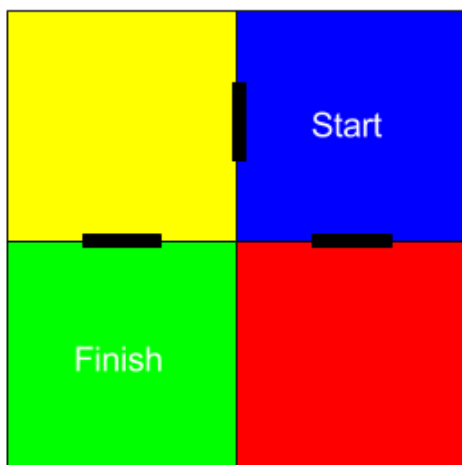
- Test your code by dragging the Sense HAT around.

You've made a Sense HAT compass!

Navigating the maze

Now let's use the compass to navigate around a maze.

The rooms and doors in the Maze are as shown in this map:



You start in the Blue room and need to find the Green room to escape.

- The code for creating a simple adventure game (like the one in the RPG project) is in maze.py in your project.

maze.py includes some functions to help you write a maze game:

- `maze.start()` - starts the game
- `maze.escaped()` - tells you whether the player has escaped the maze

- `maze.walk(dir)` - moves the player in the given direction
- `maze.getColour()` - gives you the colour of the current room

You'll need to import `maze.py`:

```
from sense_hat import *
import maze

sense = SenseHat()
sense.clear()
```

- Start the game with `maze.start()`:

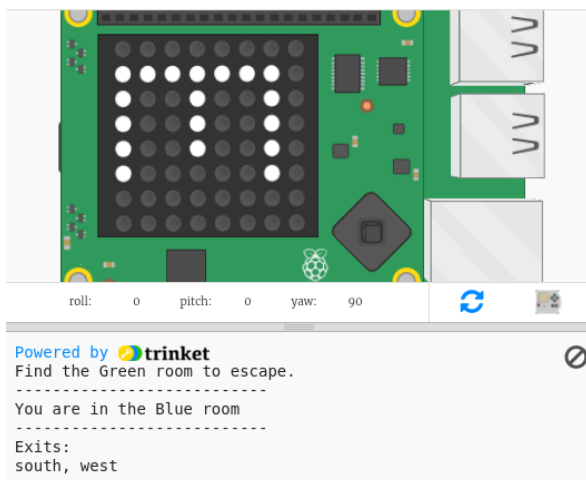
```
from sense_hat import *
import maze

maze.start()

sense = SenseHat()
sense.clear()

sense.set_rotation(90)
```

- You'll see the game instructions appear below the Sense HAT.



- To move around the maze you need to use `maze.walk(dir)` with the direction you want to move in.

Put the current compass direction in a `dir` variable, you'll need to set it for each compass direction:

```
if heading < 45 or heading > 315:
    dir = "north"
    sense.show_letter('N')
elif heading < 135:
    dir = "east"
    sense.show_letter('E')
elif heading < 225:
    dir = "south"
    sense.show_letter('S')
else:
    dir = "west"
    sense.show_letter('W')
```

- Now let's have the player move in the direction the Sense HAT compass is pointing when they press the middle button on the joystick.


```

else:
    dir = "west"
    sense.show_letter('W')

for e in sense.stick.get_events():
    if e.action == ACTION_PRESSED and e.direction == DIRECTION_MIDDLE:
        maze.walk(dir)

```

- Try moving around the maze using the compass.

To press the joystick you need to click in the Sense HAT window and then press Enter (Return) on the keyboard.

- Test your project by moving the Sense HAT to the direction you want to move in and then tapping Enter on the keyboard.

Look at the map if you need help to find the Green room.

- When the player reaches the Green room they have managed to escape the maze. Let's turn the screen green when they win and end the game:

```

for e in sense.stick.get_events():
    if e.action == ACTION_PRESSED and e.direction == DIRECTION_MIDDLE:
        maze.walk(dir)

if maze.escaped():
    sense.clear(0, 255, 0)
    break;

```

The `break` finished the loop to end the game.

Add colours

It would be better if you could tell which room you were in just by looking at the Sense HAT.

Let's display the compass letter in the colour of the current room.

For example, if you're in the Blue room and facing South you should see a blue letter S.

- You'll need to provide a `text-colour` to `sense.show_letter`. Rather than do that four times, change the code to use the `dir` variable to work out the letter to show on the Sense HAT.

`dir[0].upper()` takes the first letter of a string and turns it into a capital so "north" gives you 'N'.

Change your compass code to use `show_letter` once:

```

if heading < 45 or heading > 315:
    dir = 'north'
    sense.show_letter('N')
elif heading < 135:
    dir = 'east'
    sense.show_letter('E')
elif heading < 225:
    dir = 'south'
    sense.show_letter('S')
else:
    dir = 'west'
    sense.show_letter('W')

sense.show_letter(dir[0].upper())

```

- Your compass code should look like this:

```

if heading < 45 or heading > 315:
    dir = 'north'
elif heading < 135:
    dir = 'east'
elif heading < 225:
    dir = 'south'
else:
    dir = 'west'

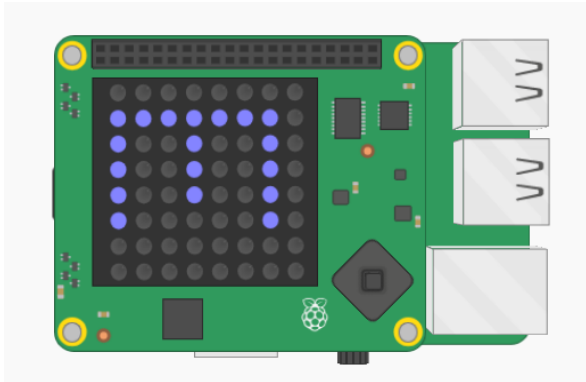
sense.show_letter(dir[0].upper())

```

- Now use the colour of the current room when you display the compass letter:

```
sense.show_letter(dir[0].upper(), text_colour=maze.getColour())
```

- Test your code and you should find that you can tell which room you're in from the colour of the letter.



Challenge: Reward the player

Can you reward the player with a cool display on the Sense HAT when they manage to escape?

You'll need to replace the highlighted code with your own code:

```
if maze.escaped():
    sense.clear(0, 255, 0)
    break;
```

Challenge: Create your own maze

Create your own colourful maze and get a friend to try and find their way out using the Sense HAT compass.

You'll need to:

- Edit the rooms dictionary in maze.py
- Edit the colours dictionary in maze.py

It might help to draw your map on paper first.

Published by Raspberry Pi Foundation (<https://www.raspberrypi.org>) under a Creative Commons license (<https://creativecommons.org/licenses/by-sa/4.0/>).

View project & license on GitHub (<https://github.com/RaspberryPiLearning/compass-maze>)