Data trail worksheet

The UK Polar Data Centre (UK PDC)

The UK PDC collects Arctic and Antarctic environmental data in the UK and makes it available online. You can read about it here.

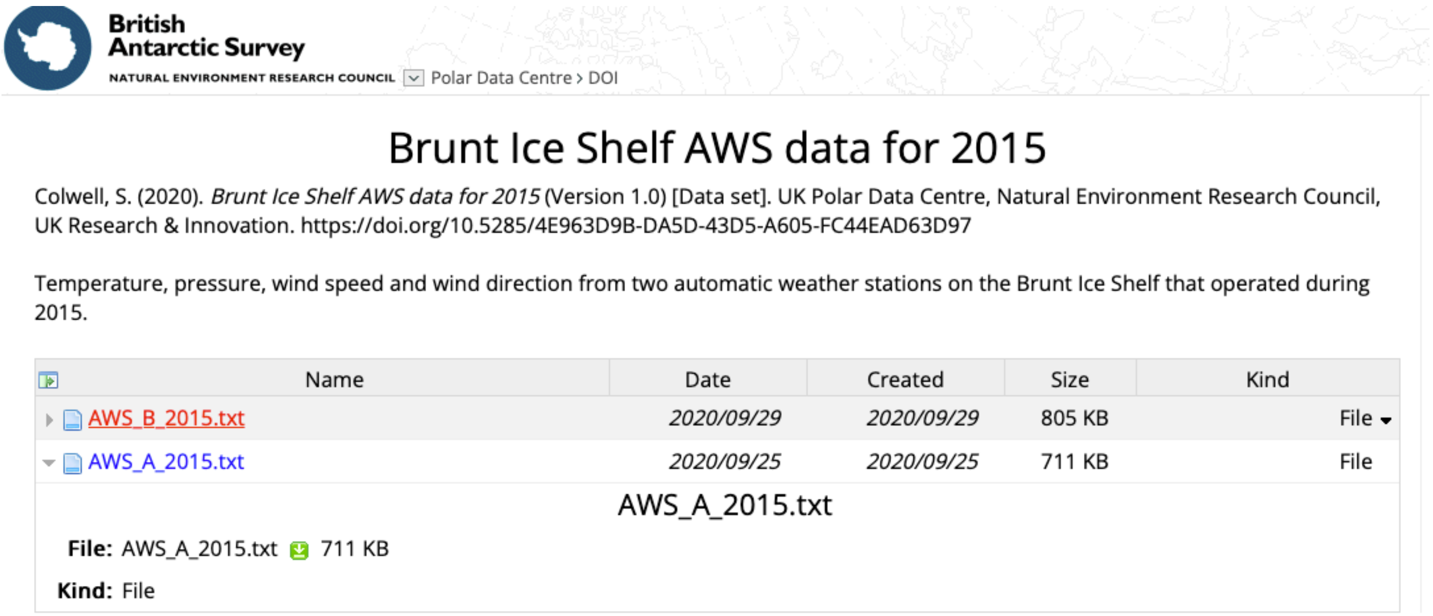
What is a data trail?

The reason Data Centres exist is to make the information they collect accessible to anyone who wants to use it. The data has been collected over many years by researchers and is added to constantly. A data trail gives you a guided example of making use of the data. Once you've answered these trail questions, you may have questions of your own to follow up on.

How to start

First, you need to get some data. We've chosen an example of environmental data from Antarctica collected in 2015.

You can download the data from the UK PDC from here. The data you need is labelled *AWS_A_2015.txt*. Open the drop-down arrow next to this label and download the file.



Brunt Ice Shelf AWS data for 2015

Colwell, S. (2020). *Brunt Ice Shelf AWS data for 2015* (Version 1.0) [Data set]. UK Polar Data Centre, Natural Environment Research Council, UK Research & Innovation. https://doi.org/10.5285/4E963D9B-DA5D-43D5-A605-FC44EAD63D97

Temperature, pressure, wind speed and wind direction from two automatic weather stations on the Brunt Ice Shelf that operated during 2015.

Name	Date	Created	Size	Kind
AWS_B_2015.txt	*2020/09/29*	*2020/09/29*	805 KB	File
AWS_A_2015.txt	*2020/09/25*	*2020/09/25*	711 KB	File

AWS_A_2015.txt

File: AWS_A_2015.txt 711 KB

Kind: File

This is a set of environmental data recorded on the Brunt Ice Shelf in Antarctica. It is recorded in a plain text file which you could read if you printed it out (but there is a lot of it, so to save paper if you try that, just print the first page!). To let the computer decode the items in the file, some coding is needed–which this worksheet will introduce to you.

Doing the data trail involves answering a set of questions from the data, and to do this we'll bring it into a coding environment.

Processing the data using Snap!

You might have used Scratch to do coding. If you have, then the Snap! system will seem very familiar. It's a blocks-based language, so you build code by arranging blocks as in Scratch, but it has extra features which we're going to use in a set of processing steps. If you've not used Scratch, then think of both it and Snap! as Lego for coding. And think of Snap! as the Lego Technic version of Scratch.

Before you start, have a look at the Data trail questions below to get an idea of what you'll be finding out.

Data trail questions

1. What information is recorded in the environmental data file (name the types of information recorded)?

2. How many data points are there?
3. What is the minimum temperature recorded over the year's data?
4. What is the maximum wind speed over the year's data?

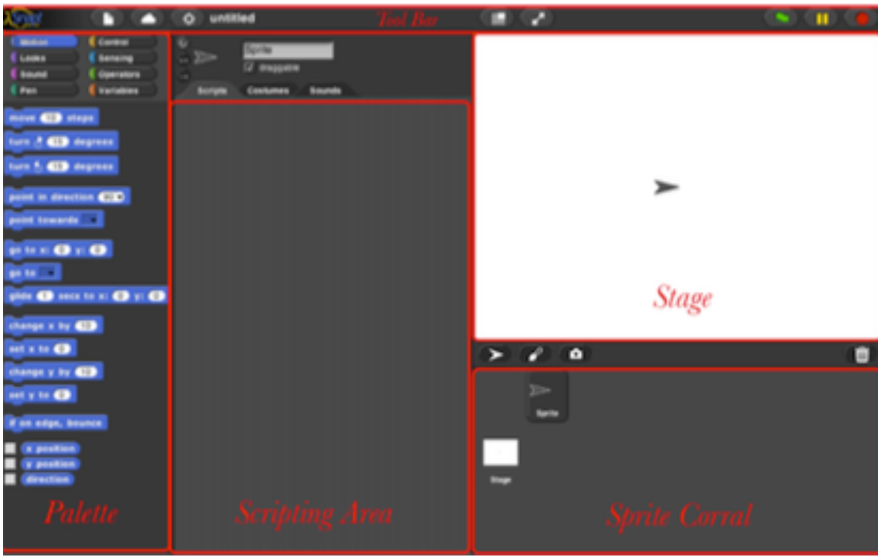
Getting the data into Snap!

First of all, start up Snap! by opening this page in your browser

Open Snap!

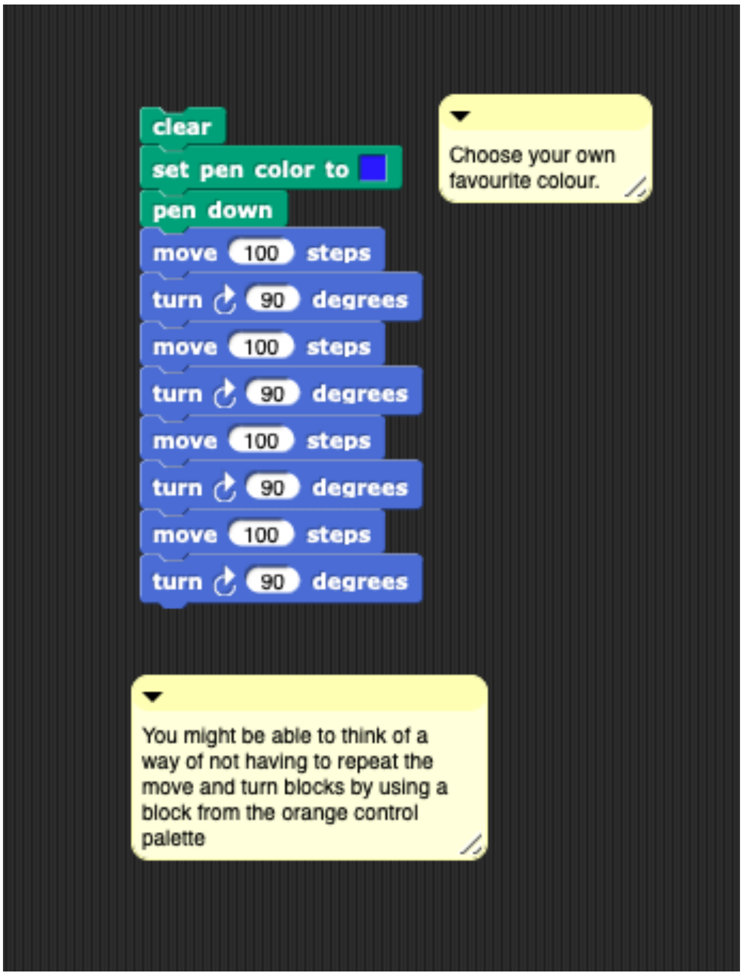
> ***Teacher note*** *What to do if you aren't able to access the Snap! software site? You can get a copy of the software onto a computer with internet access and distribute it locally (maybe on an internal school network or memory stick)–see Teacher's notes section.*

You will see some regions on the screen; including the Stage, Scripting Area, and Palette.



If you've used Scratch before, you might want to convince yourself that Snap! is like Scratch by getting it to draw a square on the stage using blocks from the *pen* and *motion* palette. Just try it, or look at the hint below:

Hint: Drag blocks from the palettes to make

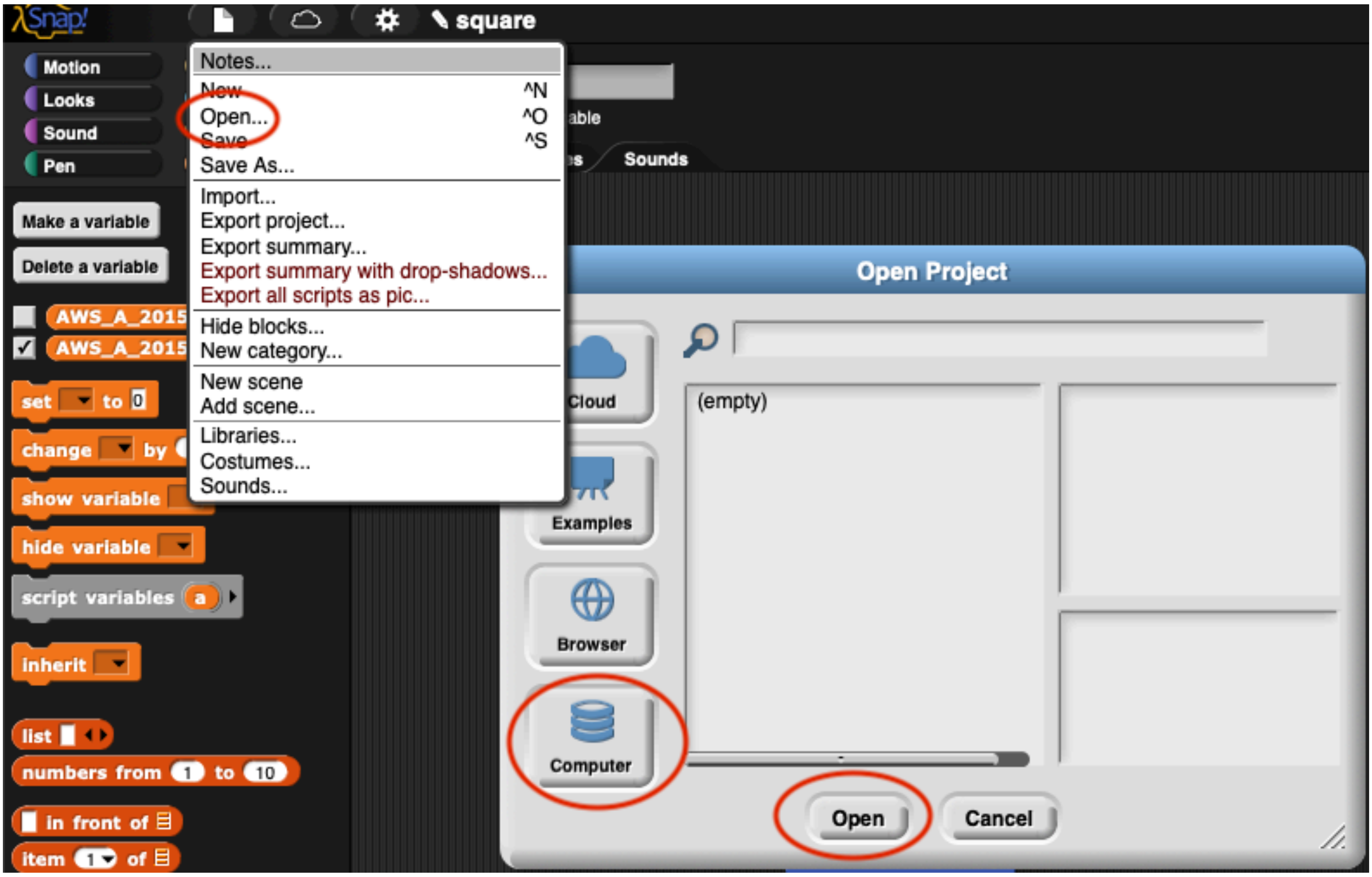


```
clear
set pen color to
pen down
move (100) steps
turn ↻ (90) degrees
move (100) steps
turn ↻ (90) degrees
move (100) steps
turn ↻ (90) degrees
move (100) steps
turn ↻ (90) degrees
```

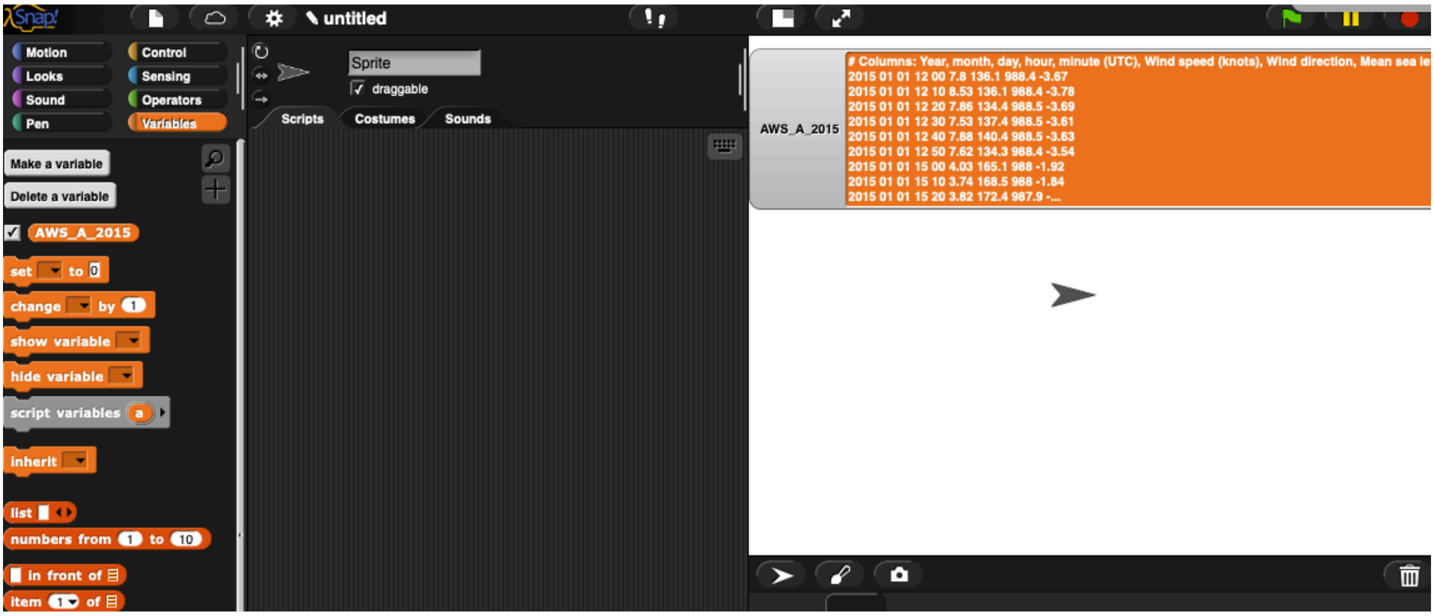
Choose your own favourite colour.

You might be able to think of a way of not having to repeat the move and turn blocks by using a block from the orange control palette

Get your data file into Snap! by using the file menu>Open , selecting *Computer* in the *Open Project* dialogue and finding your downloaded data file (which might be in a folder called *Downloads* on your computer) and then *Open*.

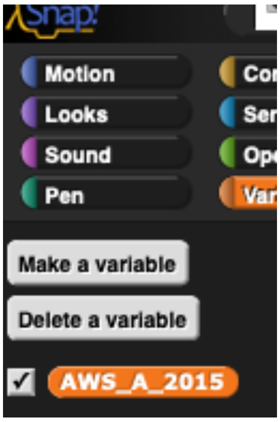


Your stage should look at bit like this:



```
# Columns: Year, month, day, hour, minute (UTC), Wind speed (knots), Wind direction, Mean sea le
2015 01 01 12 00 7.8 136.1 988.4 -3.67
2015 01 01 12 10 8.53 136.1 988.4 -3.78
2015 01 01 12 20 7.86 134.4 988.5 -3.69
2015 01 01 12 30 7.53 137.4 988.5 -3.61
2015 01 01 12 40 7.88 140.4 988.5 -3.63
2015 01 01 12 50 7.62 134.3 988.4 -3.54
2015 01 01 15 00 4.03 165.1 988 -1.92
2015 01 01 15 10 3.74 168.5 988 -1.84
2015 01 01 15 20 3.82 172.4 987.9 -...
```

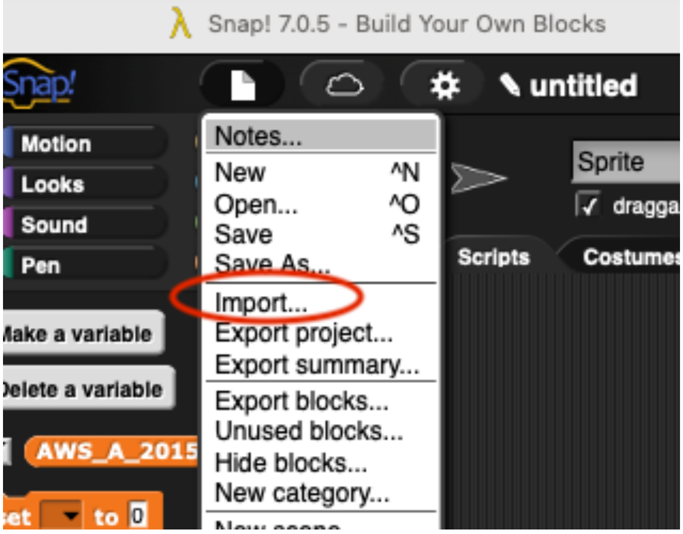
On the stage you will now see a variable, with the first part of the data displayed. You should also see the name of the variable holding your data *AWS_A_2015* in the top part of the palette.



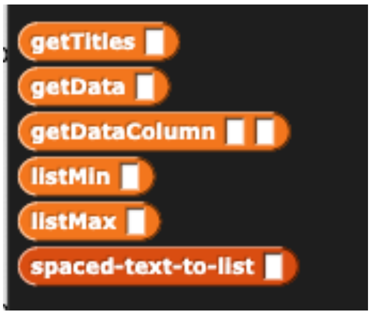
But before we answer any data trail questions, we need to see what information the data is about, and then get it into a table so that we can do calculations with it.

Looking at the data

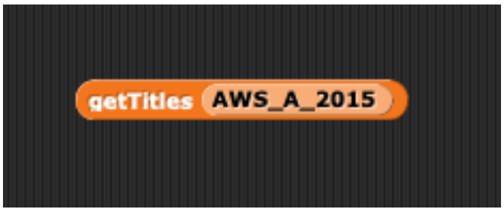
Load the special file *PDC data trail blocks.xml* into Snap! by using the file menu>Import… option and finding that file on your computer.



In the palette, make sure you've selected the *Variables* view and you will see some blocks which we can use to look at the data. For now look at *getTitles* and *getData*. There is also a helper block called *spaced-text-to-list* which *getData* uses.



These blocks can take the data variable and look inside. Try dragging the *getTitles* block into the scripting area (which makes a copy of it). Notice that there is a blank space inside it. This accepts a variable, so try dragging our data variable into the blank space–it highlights when you have the variable in the right place. You should see something like:



Click on the getTitles block in the scripting area, what do you see? This block should report the titles of the data columns in a 'speech bubble'. (Because this sort of block reports an answer, it's called a *reporter block*.) Can you answer the 1st trail question now?

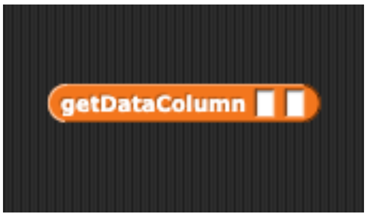
Try the same thing but now using the *getData* reporter block. What do you see? Can you use the information displayed to read off the first line of weather data for the year 2015?

Now scroll down through the information. Can you now answer the 2nd question (**hint:** ignore the very last line which contains only 1 blank cell).

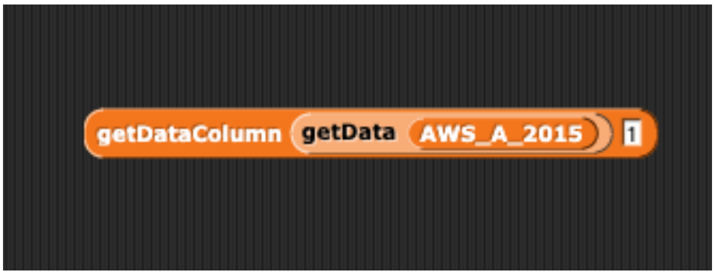
Doing calculations on the data

You can see that using *getData* gives us a table of the weather data. Have a look at the remaining special reporter blocks at the end of the palette.

We'll need to look at individual slices or columns of our data to look at one particular measurement. There is a block:



which has two inputs; the first is a data table, and the second is the number of column we want to extract to give a vertical slice through the data. Try an example, build this code and click on it to see what it reports. What is the title of the result?



Try changing the column number to see another set of results reported.

There are a final two blocks which are used to do calculations on our data; they take a list of numbers (which is what getDataColumn gives us) and then can pick the minimum (*listMin*) and maximum (*listMax*) values it finds in the list. Note that *listMin* and *listMax* have a single gap which you need drag your column of data of interest into.

Can you find the answers to the last two questions, using getDataColumn to pick out the weather data items you need and listMin or listMax to calculate a minimum or maximum value?