**Exercise 7**

**1)** Create a new Power BI Desktop file, and load the **Buildings** worksheet from the Excel workbook in the above folder

There are 3 columns giving the uses to which each building is put:

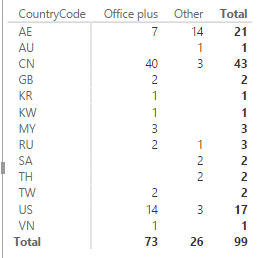


*The last 3 columns tell you how each building is used.*

Use the **IF** function to create a new column to show for each office its *use type*, following these rules:

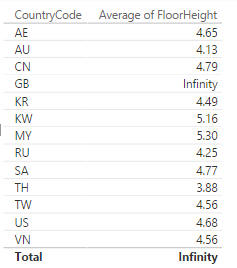
|  |  |
| --- | --- |
| **Rule** | **Use type** |
| If any of the 3 **Use** columns is **office** | **Office, etc** |
| Otherwise | **Other** |

Show the number of buildings per use type and country code in a matrix - you should get these results:



*AE - the United Arab Emirates - seems to be the place to go if you want to see skyscrapers without offices in them.*

Create a column giving the floor height per building (the height in metres divided by the number of floors), and use this to show the average floor height per country:

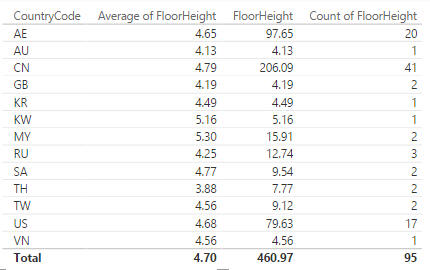


*It seems hard to believe British buildings are infinitely high; easier to believe that Wise Owl have added a rogue office to the list with nothing filled in for the number of floors, and this is giving a divide by zero error.*

Amend your floor height formula so that it gives a blank if there is no value in the **Floors**column.

*To do this you can either use the***DIVIDE***function, or use the***IFERROR***function to test for an error.*

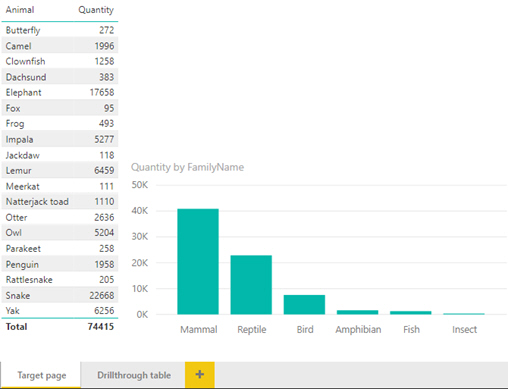
After correcting your formula, your table should now look like this:



*For Britain there are two buildings, but in calculating the average the one which returns a blank floor height is (correctly) ignored.*

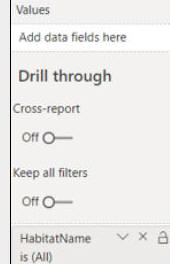
Save this file as **Kingsmoor**, and close down the Power BI instance you're using.

**2)** Start by creating the following table and chart using the **Drilling down to details.** (choose to import all of the tables listed).



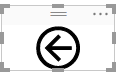
*This is our main data page and the one we will filter through to, when someone uses drill-through on a habitat.*

To make this into a page that will be filtered, drag the **HabitatName** into the **Drillthrough** **filters** section of the field well:



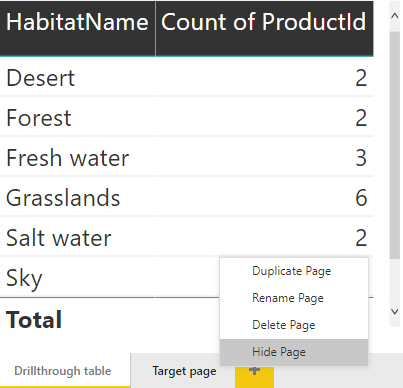
*Say that you will drill through by the value of the habitat field.*

Power BI will create a back button automatically after you add a drill-through field.



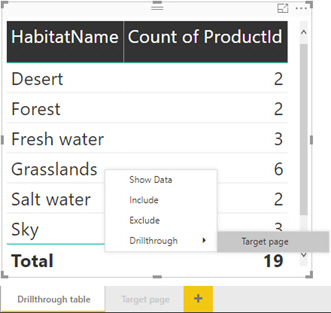
*The back button is a really handy tool. It returns the user to the page they came from last!*

Now to create the page to drill from. On a separate page called **Drillthrough table**create the following table:



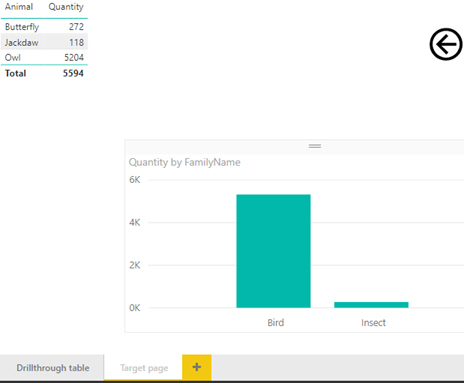
*You could also at this point hide the page called****Target page****that you want to drill to. This prevents anyone getting to that page without using the drill-through.*

Surprisingly that is all that you need to do. Now visuals that contain a field include in the **Drillthrough filters** can be right clicked, to navigate to the filtered page.



*As you add more pages that have a drillt-hrough filter on them, more options will appear in the****Drillthrough****section shown.*

Finally the result of all your hard work should be that the hidden report page is automatically filtered:



*So if I have 5,322 birds in the hand does that mean they are worth 10,644 in the bush?*

Optionally save this as **Who you calling a bird.pbix**and close it down.

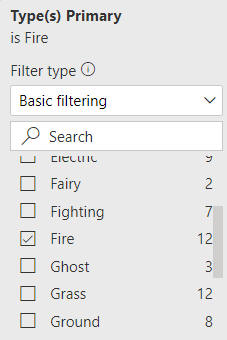
**3)**Load data from **Does this count as a Zoo** workbook

Try to make a report which has a page for each kind of creature. When you click the image of that creature you will see all the animals of the same type:

|  |  |
| --- | --- |
| *Gotta catch them all* |  |
| *Blue for water, green for grass and orange for fire.* | *Each page has a table filtered  to the creatures type.* |

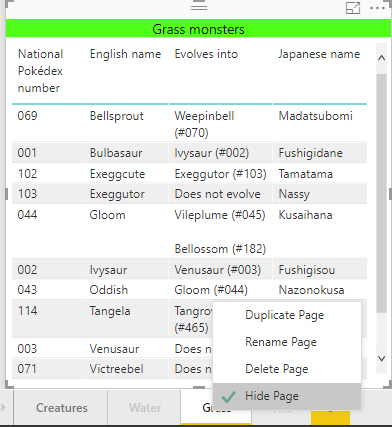
*Bookmarks can be attached to images, shapes and buttons. They can't be attached to text boxes!*

To get you started look in the above folder to find the three images used. Note the names of the images. Create a page with a table for each:



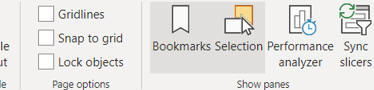
*Use a page or visual filter to make each page about only one type. This page is for the fire lizard.*

When the pages are all created, hide them so that the only way user can reach the page is by clicking the relevant image on the first page.



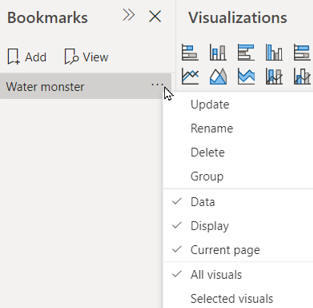
*Don't try and hide all the pages. One must remain visible - otherwise how will they get to the hidden pages!*

To create the bookmarks make sure you have the bookmark feature turned on in the **View** tab:



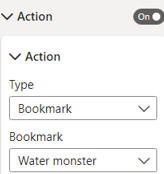
*If you plan on doing more bookmark exercises, turn on****Selection Pane****as well.*

Now choose the first page to create as a bookmark, and choose **Add** in the **Bookmark Pane**:



*Leave all the other options ticked for the time being. Name the bookmark after the page.*

To apply the bookmark, return to the page with the images. Click on one of the images (in this case the turtle) and assign an action to it:



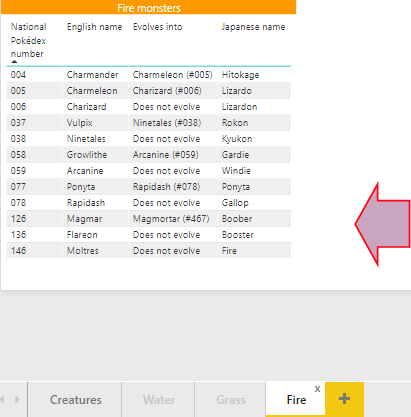
*Choose the****Type****as****Bookmark****and then select the bookmark you just created.*

Clicking on the turtle should now take you through to the water page we created earlier:



*Now repeat this for each of the other images and bookmarks!*

If you have time try adding a back arrow which returns you to the first page of the report. Use the type **Back** instead of **Bookmark** when creating the **Link**.



*Alternatively you could create a bookmark for the image page and link it to the back arrow.*

Optionally save this report as **Does this count as a zoo,**then close it down.

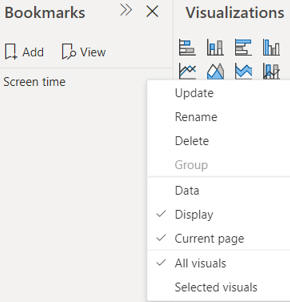
**4)** Load data from **Game of screen time**

Try and create a report which allows the user to see different Game of Thrones characters based on how long they appeared in the film:

|  |  |
| --- | --- |
|  |  |
| *Choose an importance level and click yes ...* | *To return a page showing  characters of that category* |

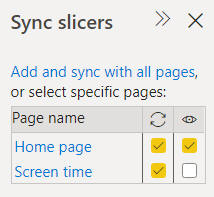
The categories are completely arbitrary and in no way reflect the current state of a character. So no spoilers ahead!

In the folder above is a **csv** file called **Game of Screen time**with all the data needed.  When creating the bookmark for **Screen time** make sure to un-tick **Data**:



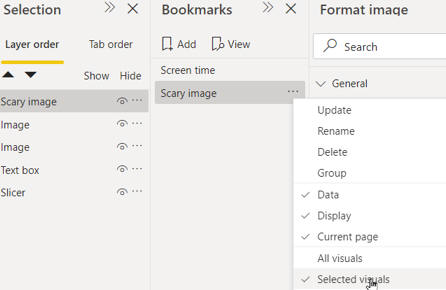
*If you leave****Data****ticked then Power BI Desktop will save the current data state of the page. Leaving it unticked means slicers can carry over.*

Attach the bookmark to the **YES** image found in the folder above. Now create a synced slicer which is hidden on the **Screen time** page but visible on the **Home page**.



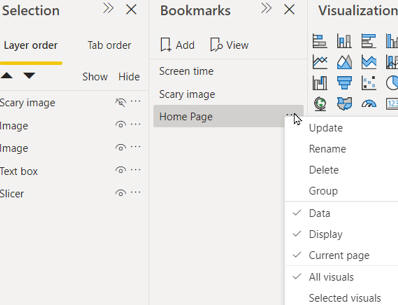
*If you left Data ticked then your synced slicer won't have any affect on the bookmark!*

Clicking **NO** will present a disapproving image! Create a bookmark that brings up the **Scary Image** (found in the above folder) on the **Home page** when a user clicks **NO**:



*Choose****Selected visuals****and only highlight the****Scary Image****icon when creating the bookmark.*

Attach the new bookmark to the **NO** button. Create a final bookmark for the **Home Page** where the **Scary image** is hidden.



*This time choose All visuals as we want to save the state of the entire page. Link this bookmark to****Scary Image****.*

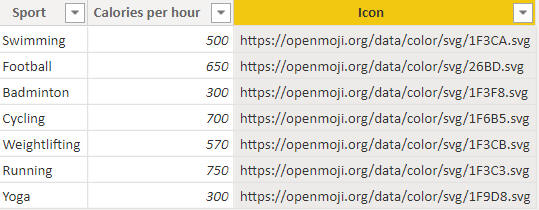
Now when you click **YES** we see the hidden page filtered by the **Importance** slicer. Clicking **NO** gives a blood curdling image:



*Thankfully clicking on the image will return use to the****Home page****and to (relative) safety.*

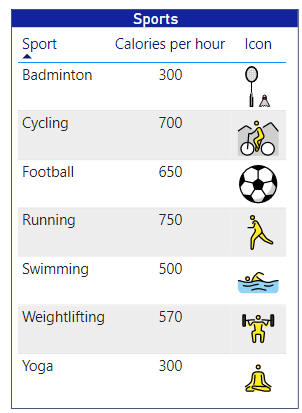
Create a back button on the **Screen** time page to come full circle. Optionally save this file as **Game of Screen time**, then close it down.

**5)** Load the table of data from **Sport Images** workbook in the above folder, and choose to show the icon as an image:



*Choose to show the column shown selected above as an Image URL.*

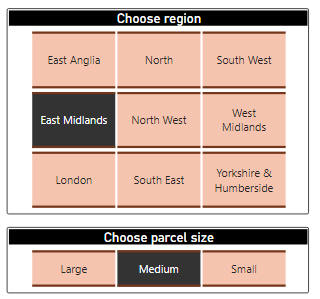
Create and format a visual to list out the sports, to look something like this:



*Your formatting doesn't have to look the same as this!*

Save this report as Sporting icons, then close it down.

**6)** Create a new Power BI report, and load data from **Parcels** workbook in the above folder.  Create two slicers to allow a user to choose a region and parcel size:



*Your slicers don't have to look anything like this, or even be formatted!*

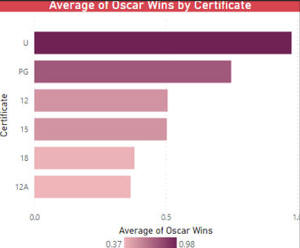
Now add a visual showing the total quantity of parcels by weight for the region/parcel size you have selected:



*Providing you choose the correct aggregation options for each field, this is what you should see for the****East Midlands****region for****Medium****parcel weights.*

Save this report as **Weighty problem**, then close it down.

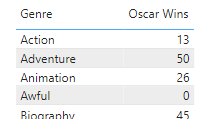
**7)** Create a new report, and load the **Movies**worksheet from the Excel workbook of films in the above folder.  From this, create the following visual:



*This bar chart shows the average number of Oscar wins by certificate (filter the data to remove****Unknown****certificates). The higher the average Oscar wins, the darker the bar's colour should appear. You can use gradient conditional formatting to achieve this - Power BI should then automatically create a legend as shown.*

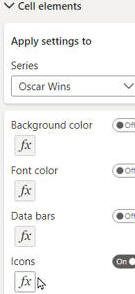
Save your report as **U can win Oscars**, then close it down.

**8)** Load the two worksheets from **Movies** workbook in the above folder, and create a table showing the total number of Oscars won by genre:



*Here the genres are listed in alphabetical order.*

Now apply icons to the Oscar Wins column:

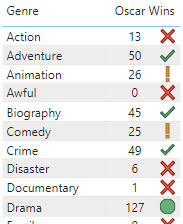


*This shows one way to apply icons to this column.*

Apply icons as follows:

|  |  |
| --- | --- |
| Total number of Oscars | Display |
| Up to 20 | A red cross |
| 20 to 40 | A yellow exclamation mark |
| 40 to 60 | A green tick |
| More than 60 | A green circle |

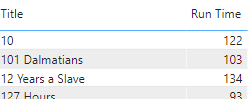
Here's what your final table should look like:



*Your final table should look similar to this (with the icons to the right of the text).*

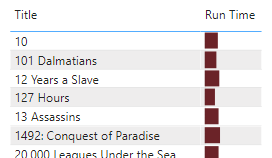
Save your report as I-Con, then close it down.

**9)** Create a new report, and load the tables from **Movies** workbook in the above folder.  Use this data to create a table showing the run time in minutes for each film:



*The start of the table, with the films in alphabetical order.*

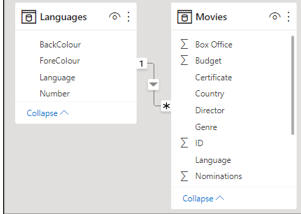
Tell Power BI to show the run time as a bar (starting from 0 minutes):



*Here we've chosen to show just the bar, and not the underlying number.*

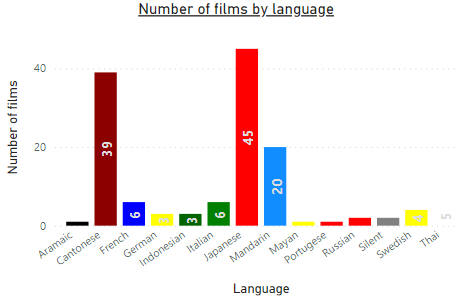
Save this report as **Barcode**, then close it down.

**10)** Create a new Power BI report, and load data from both worksheets in the **Movies** workbook in the above folder:



*Power BI will automatically link the tables by the****Language****column. Conveniently, the****Languages****table contains the foreground and background colour you should use for formatting each language.*

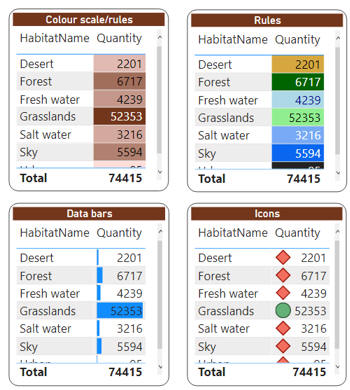
 Create a column chart showing the number of films by language, applying the foreground and background colours from the table of languages:



*To avoid****English****swamping your data (as it so often does), filter your chart to remove it. The above chart uses the****ForeColour****field in the****Language****table to set the data label text colour, and the****BackColour****field in the same table to set the column value back colour.*

Save this chart as **Inappropriate colours**, then close it down.

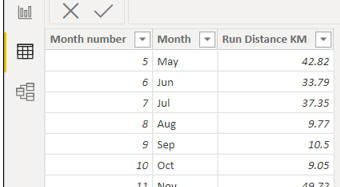
**11)** Create a new report, and load data from **Habitatty stuff** workbook in the above folder.  Use this to see how many of the following effects you can create!



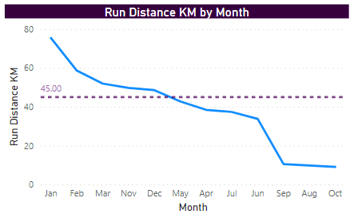
*A smorgasbord of conditional formatting effects!*

 When you've finished (or had enough), save your report as **Smorgasbord**, then close it down.

**12)** Create a new Power BI report, and load the running data from **Total Distance Run 2020** CSV file in the above folder:



Create a line chart showing distance run by month:



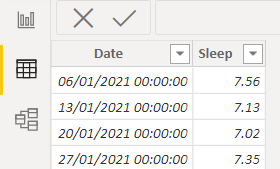
*See the notes below for how to do this!*

 Here are some extra things you'll need to do:

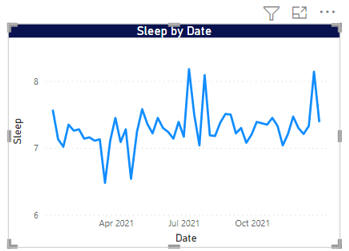
1. You'll need to sort the month name by the month number column to get the months to appear in the correct chronological order.
2. You will need to add a Y axis constant line to show the initial target of 45 km per month.

Save this report as **Oh COVID**, then close it down.

**13)** Create a new Power BI report, and load data from **Sleep** CSV file in the above folder:

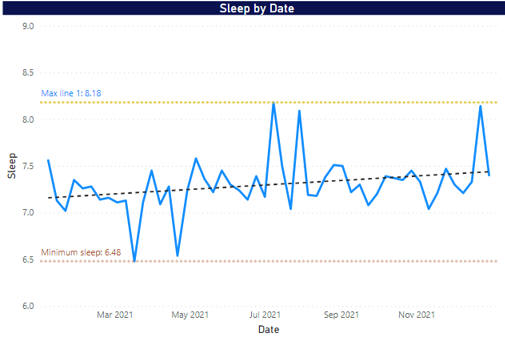


Show the hours slept by week (you'll need to remove the date hierarchy to do this):



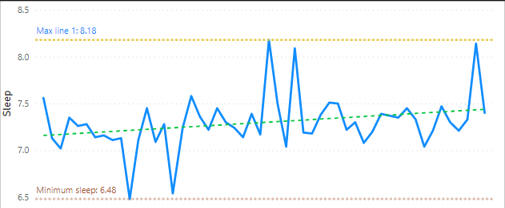
*REM These figures seem reasonable (this is a joke for old programmers only).*

Add a minimum and maximum line, making sure you show the label for each:



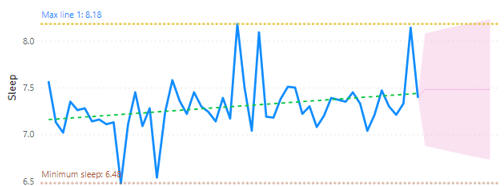
*You may want to change your scaling, so that the minimum isn't on the bottom axis.*

Now add a trend line showing what this person can expect:



*The trend is up: this person is gradually getting more sleep!*

If you still have the energy, add a two-month forecast period:



*There are only 12 months of data, so the forecast is ... uncertain.*

Save this report as **Greeting the sleep monster**, then close it down.