


```
FROM Taxa INNER JOIN (Records INNER JOIN Sites ON Records.[*Site] = Sites.[_guk]) ON Taxa.[_guk] = Records.[*Taxon]
```

```
GROUP BY Records.[_guk], Sites.[_guk], Taxa.[_guk], Taxa.Taxon, Sites.Name, Sites.OSGridRef, Int([_xo]/1000), Int([_yo]/1000), Sites.ViceCounty, Records.Quantity
```

HAVING

```
((If(Len([OSGridRef])<=5,Null,If(Len([OSGridRef])=6,[OSGridRef],If(Len([OSGridRef])=8,Left([OSGridRef],4) & Mid([OSGridRef],6,2),If(Len([OSGridRef])=10,Left([OSGridRef],4) & Mid([OSGridRef],7,2),If(Len([OSGridRef])=12,Left([OSGridRef],4) & Mid([OSGridRef],8,2),If(Len([OSGridRef])=14,Left([OSGridRef],4) & Mid([OSGridRef],9,2)))))))) Is Not Null) AND ((Sites.ViceCounty)=17) AND ((Records.Quantity)<>-7))
```

```
ORDER BY Sites.OSGridRef;
```

Note: In MS Access the above queries the Site and Records tables to generate a Select Query of Records._guk, Sites._guk, OSGridRef, 1k Square and X and Y co-ordinates. Records where taxa are recorded as 'Not present' (Quantity = -7) are ignored. **Once created you do not need to run this query as it is called by the next query.**

Mapping Records - Step 2

```
SELECT DISTINCT Count(Records.[_guk]) AS CountOf_guk, [Mapping Records - Step 1].Sites.[_guk], [Mapping Records - Step 1].Taxa.[_guk], [Mapping Records - Step 1].[1kSquare], [Mapping Records - Step 1].X, [Mapping Records - Step 1].Y, Year([Date]) AS Year, Records.Date
```

```
FROM [Mapping Records - Step 1] INNER JOIN (Records INNER JOIN Taxa ON Records.[*Taxon] = Taxa.[_guk]) ON ([Mapping Records - Step 1].Records.[_guk] = Records.[_guk]) AND ([Mapping Records - Step 1].Sites.[_guk] = Records.[*Site])
```

```
GROUP BY [Mapping Records - Step 1].Sites.[_guk], [Mapping Records - Step 1].Taxa.[_guk], [Mapping Records - Step 1].[1kSquare], [Mapping Records - Step 1].X, [Mapping Records - Step 1].Y, Records.Date;
```

Note: In MS Access the above queries the Records, and Sites tables and the Mapping Records – Step 1 query to generate a Select Query of Records._guk, Sites._guk, Taxa._gukOSGridRef, 1k Square and X and Y co-ordinates, Year(Date) and Date. **Once created you do not need to run this query as it is called by the next query.**

Mapping Records - Step 3

```
SELECT [Mapping Records - Step 2].[1kSquare], [Mapping Records - Step 2].X, [Mapping Records - Step 2].Y, Sum([Mapping Records - Step 2].CountOf_guk) AS SumOfCountOf_guk INTO [Mapping Records - Table 1]
```

```
FROM [Mapping Records - Step 2]
```

```
WHERE ((([Mapping Records - Step 2].Year)>=[Start year] And ([Mapping Records - Step 2].Year)<=[End year]))
```

```
GROUP BY [Mapping Records - Step 2].[1kSquare], [Mapping Records - Step 2].X, [Mapping Records - Step 2].Y;
```

Note: In MS Access the above queries the Mapping Records – Step 2 query and prompts you to input the Start Year and End Year to generate a Select query which includes a list of 1k Squares, X and Y co-ordinates, Sum of Records for the Year Range (Start Year to End Year inclusive)

Mapping Records - Step 4

```
SELECT DISTINCT [1kSquareList].ID, [1kSquareList].[Vice-County], [Mapping Records - Table 1].[1kSquare], [Mapping Records - Table 1].X, [Mapping Records - Table 1].Y, [1kSquareList].ONE_KM, [1kSquareList].X, [1kSquareList].Y, [Mapping Records - Table 1].SumOfCountOf_guk INTO [Mapping Records - Table 2]
```

```
FROM 1kSquareList LEFT JOIN [Mapping Records - Table 1] ON [1kSquareList].ONE_KM = [Mapping Records - Table 1].[1kSquare]
```

```
WHERE ((([Mapping Records - Table 1].[1kSquare]) Is Null) AND ((([Mapping Records - Table 1].X) Is Null) AND (([Mapping Records - Table 1].Y) Is Null));
```

Note: In MS Access the above queries the Mapping Records Table 1 – Table generated against the 1k SquareList table you downloaded and imported earlier to generate a new table of 1k Squares in your chosen Vice-County which do not have any records. This new table is Mapping Records – Table 2 – Table.

Mapping Records - Step 5

```
INSERT INTO [Mapping Records - Table 1] ( 1kSquare, X, Y )
```

```
SELECT [Mapping Records - Table 2].ONE_KM, [Mapping Records - Table 2].[1kSquareList_X], [Mapping Records - Table 2].[1kSquareList_Y]
```

```
FROM [Mapping Records - Table 2];
```

Note: In MS Access the above queries the 1k Squares without records (Totals = null) in Mapping Records Table 2 – Table are appended to the records Mapping Records Table 1 – Table.

Mapping Records - Step 6

```
UPDATE [Mapping Records - Table 1] SET [Mapping Records - Table 1].SumOfCountOf_guk = 0
```

```
WHERE ((([Mapping Records - Table 1].SumOfCountOf_guk) Is Null));
```

Note: In MS Access the above updates the null entries in Mapping Records Table 1 - Table **to zero**

Mapping Records - Step 7

```
SELECT [1kSquareList].ONE_KM, [1kSquareList].X, [1kSquareList].Y, [Mapping Records - Table 1].[1kSquare], [Mapping Records - Table 1].X, [Mapping Records - Table 1].Y, [Mapping Records - Table 1].SumOfCountOf_guk
```

```
FROM 1kSquareList RIGHT JOIN [Mapping Records - Table 1] ON [1kSquareList].ONE_KM = [Mapping Records - Table 1].[1kSquare]
```

```
WHERE ((([1kSquareList].ONE_KM) Is Null));
```

Note: In MS Access the above query generates a list of records in which the 1k Square lies outside the Vice-County. These are mostly probably due to an input error. **You should review these records in 'Mapping Records Table 1 – Table' and either correct or delete them before proceeding to the final query.**

Mapping Records - Step Final

```
TRANSFORM Sum([Mapping Records - Table 1].SumOfCountOf_guk) AS SumOfSumOfCountOf_guk
```

```
SELECT [Mapping Records - Table 1].Y
```

```
FROM [Mapping Records - Table 1]
```

```
GROUP BY [Mapping Records - Table 1].Y
```

```
ORDER BY [Mapping Records - Table 1].Y DESC
```

```
PIVOT [Mapping Records - Table 1].X;
```

Note: In MS Access the above query creates a cross-tab table, plotting the numbers of records in each 1km Square by eastings and northings (X and Y co-ordinates).

With the cursor in the top left-hand corner copy the table to the clipboard (Ctrl+C).

Open Excel and paste the table into a new blank worksheet (Ctrl-V) and save it.

You are now ready to customise the 'map' using Excel.