



# Reading Excel files

Jeffrey Leek  
Johns Hopkins Bloomberg School of Public Health

# Excel files

*Still probably the most widely used format for sharing data*

The image shows a browser window at the top with the URL [office.microsoft.com/en-us/excel/](http://office.microsoft.com/en-us/excel/). Below the browser is a search bar and a promotional banner for Excel. The banner features the Excel logo and the text "Analyze. What's new in Excel? >". To the right of the banner is a laptop displaying an Excel spreadsheet titled "EMPLOYEE TRAVEL EXPENSE TRENDS". The spreadsheet shows a bar chart with data for various months and categories. Below the banner are three buttons: "Discover", "Visualize", and "Share".

Buy with Office Try 1 month FREE

Excel

Analyze.

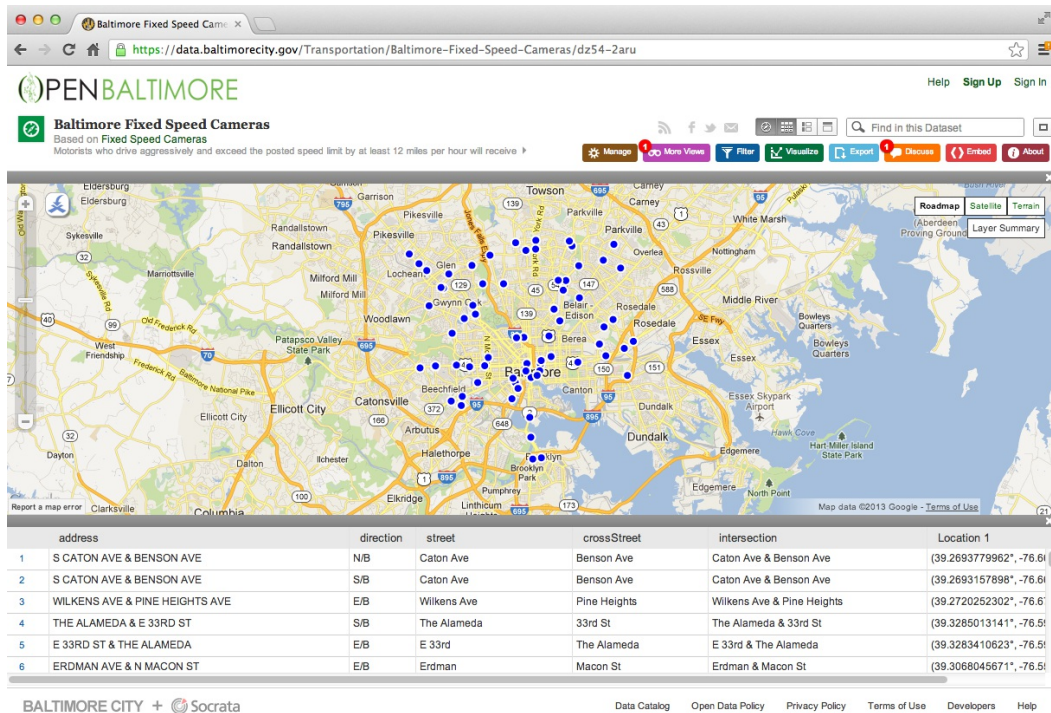
What's new in Excel? >

Discover Visualize Share

Discover and reveal the insights hidden in your data

<http://office.microsoft.com/en-us/excel/>

# Example - Baltimore camera data



<https://data.baltimorecity.gov/Transportation/Baltimore-Fixed-Speed-Cameras/dz54-2aru>

# Download the file to load

```
if(!file.exists("data")){dir.create("data")}  
fileUrl <- "https://data.baltimorecity.gov/api/views/dz54-2aru/rows.xlsx?accessType=DOWNLOAD"  
download.file(fileUrl,destfile="./data/cameras.xlsx",method="curl")  
dateDownloaded <- date()
```

# read.xlsx(), read.xlsx2() {xlsx package}

```
library(xlsx)
cameraData <- read.xlsx("./data/cameras.xlsx", sheetIndex=1, header=TRUE)
head(cameraData)
```

	address	direction	street	crossStreet	intersection
1	S CATON AVE & BENSON AVE	N/B	Caton Ave	Benson Ave	Caton Ave & Benson Ave
2	S CATON AVE & BENSON AVE	S/B	Caton Ave	Benson Ave	Caton Ave & Benson Ave
3	WILKENS AVE & PINE HEIGHTS AVE	E/B	Wilkens Ave	Pine Heights	Wilkens Ave & Pine Heights
4	THE ALAMEDA & E 33RD ST	S/B	The Alameda	33rd St	The Alameda & 33rd St
5	E 33RD ST & THE ALAMEDA	E/B	E 33rd	The Alameda	E 33rd & The Alameda
6					
1	(39.2693779962, -76.6688185297)				
2	(39.2693157898, -76.6689698176)				
3	(39.2720252302, -76.676960806)				
4	(39.3285013141, -76.5953545714)				
5	(39.3283410623, -76.5953594625)				
6	(39.3068045671, -76.5593167803)				

# Reading specific rows and columns

```
colIndex <- 2:3
rowIndex <- 1:4
cameraDataSubset <- read.xlsx("./data/cameras.xlsx", sheetIndex=1,
                              colIndex=colIndex, rowIndex=rowIndex)
cameraDataSubset
```

	direction	street
1	N/B	Caton Ave
2	S/B	Caton Ave
3	E/B	Wilkins Ave

# Further notes

- The *write.xlsx* function will write out an Excel file with similar arguments.
- *read.xlsx2* is much faster than *read.xlsx* but for reading subsets of rows may be slightly unstable.
- The [XLConnect](#) package has more options for writing and manipulating Excel files
- The [XLConnect vignette](#) is a good place to start for that package
- In general it is advised to store your data in either a database or in comma separated files (.csv) or tab separated files (.tab/.txt) as they are easier to distribute.