

XLWings Handout

What is xlwings?

- `xlwings` provides a Python interface to and from Excel.
- Python capabilities overlap with VBA, but Python is faster, and has many built-in numerics packages that provide functionality not easily accessibly with VBA.
- `xlwings` aims to provide these capabilities with minimal effort.

Importing xlwings module

```
import xlwings as xw
```

Manipulating a Workbook

- Open a workbook

```
wb = xw.books.open('data.xlsx')
```

- Save and Close a workbook

```
wb.save()  
wb.close()
```

- Select a sheet from a workbook

```
sheet = wb.sheets('Sheet1')
```

- Create a new sheet

```
wb.sheets.add('New Sheet', after='Sheet1')
```

Moving data between Excel and Python

- Assign a Python variable from an Excel cell

```
ft_per_m = sheet.range('B4').value
```

- Assign a numpy array from a range of Excel cells

```
v = sheet.range('F2:F42').options(np.array).value
```

- Assign an Excel cell from a Python variable

```
sheet.range('B6').value = lbm_per_kg
```

- Assign a range of Excel cells from a numpy array

```
# need np.newaxis because t is a 1D array ,  
# but all ranges in Excel are 2D  
sheet.range('J2:J42').value = t[:,np.newaxis]
```

- Add a formula to a cell

```
sheet.range('H2').formula = "=sum(E2:E42)"
```

Put a matplotlib plot in Excel

- Example:

```
fig = plt.figure()  
  
plt.rc("font", size=14)  
plt.plot(t,T**1.5)  
plt.xlabel('t (s)')  
plt.ylabel('T (K)')  
  
sheet.pictures.add(fig, name="T_vs_t", update=True,\  
                  left=700, top=150)  
  
# keeps the plot from showing in python console  
plt.close()
```

Other Features

- Run Python script as a [macro](#) from within Excel.
- Create a [user defined function](#) for Excel from a Python script.