

### Imports

```
import xlwings
import Pandas as pd
```

### Workbook and Sheet Functions

```
wb = xw.Book('TSLA.xlsx') #Connect to a workbook
```

```
wb.sheets #Show sheets in the workbook
```

```
wb.sheet[0] #Access a sheet by integer
```

```
wb.sheet['sheet_name'] #Access a sheet by name
```

```
wb.sheets.add('new') #Add a sheet
```

```
wb.__delitem__('Sheet Name to Delete')
#Delete a sheet
```

### Read and Write into an Excel file

```
wb.range('A1').value #Return the value from cell
```

```
wb.range('A1:C3').value #Return values from a range
```

```
wb.range('A1').expand('right') #Returns all values in a row
```

```
wb.range('A1').expand('down') #Returns all values in a column
```

```
wb.range('A1').value = 'New Value' #assign a new value to a cell horizontally
```

```
wb.range('A1').options(transpose=True).value = [1,2,3] #Fill values to a range of cells up vertically
```

```
wb.range('A1').value is none #Return boolean
```

```
wb.range('A1').expand('right').options(pd.DataFrame).value #Create a dataframe from a row
```

```
wb.range('A1:C3').options(pd.DataFrame).value #Create a dataframe from a range
```

```
wb.range('A1').expand().options(pd.DataFrame).value #Create a dataframe for all data
```

### Alternate to Range

```
wb['A1'].value
```

### Read and Write into an Excel file (cont)

```
wb['B4'].name = 'Cell reference name' #
Creates a named reference
```

### Plotting

```
import matplotlib as plt #import plotting library
```

```
fig = df['Adj Close'].plot
```

```
plt.xlabel('Date')
```

```
plt.ylabel('$ price')
```

```
plt.title('TSLA Price')
```

```
plt.show()
```

```
tsla_fig = fig.get_figure() #Save the figure
```

```
xw.sheets.active.pictures.add(tsla_fig,
name='TSLA',update=True) # copy the figure to Excel
```

### Values

```
tsla_sht.range('A1').value = '$12' #This is a text value
```

```
price = tsla_sht.range('A1').value
```

```
price = float(price) #convert text into float
```

```
price + 2.5 #You can now perform computation in the float
```

### Saving Workbooks

```
wb.save('tsal_new.xlsx') #Save as a new Excel file
```

```
app = xw.apps.activate
```

```
app.quit() #quit the active app
```

### Saving Workbooks

```
wb.save('tsal_new.xlsx') #Save as a new Excel file
```

```
app = xw.apps.activate
```

```
app.quit() #quit the active app
```

### Add Style

#### Store RGB Values

```
white = (255,255,255) #RGB for white
```

```
blue = (31,73,125) #RGB for Blue
```

```
yellow = (155,192,0) #RGB for yellow
```

#### Store Cell Range

```
rowHeader = tsla.range('B1:D1')
```

```
colHeader = tsla.range('A2:A4')
```

```
dataCells = tsla.range('B2:D4')
```

#### Set Cell Color

##### On Single Cell

```
rowHeader.Color = blue
```

```
colHeader.Color = blue
```

##### On a Range

```
dataCells.Color = yellow'
```

#### Set Font Color

```
rowHeader.api.Font.Color = white
```

```
colHeader.api.Font.Color = white
```

#### Set Font Bold

```
rowHeader.api.Font.Bold = True
```

```
colHeader.api.Font.Bold = True
```

#### Set Cell Border on a Cell

```
B6 = tsla.range('B6')
```

```
C6 = tsla.range('C6')
```

```
D6 = tsla.range('D6')
```

```
E6 = tsla.range('E6')
```

```
B6.api.Borders(1).Weight=3 #Left border
```

```
C6.api.Borders(2).Weight=3 #Right border
```

```
B6.api.Borders(3).Weight=3 #Top border
```

```
B6.api.Borders(4).Weight=3 #Bottom border
```

#### Set Border on a Range

```
dataCells.api.Borders(1).Weight=3
```

```
dataCells.api.Borders(2).Weight=3
```

```
dataCells.api.Borders(3).Weight=3
```

```
dataCells.api.Borders(4).Weight=3
```

#### Center Text

```
center = xw.constants.HAlign.xlHAlign
```

```
dataCells.api.HorizontalAlignment=
```



By Ianh

[cheatography.com/ianh/](https://cheatography.com/ianh/)

Not published yet.

Last updated 29th September, 2023.

Page 1 of 1.

Sponsored by [ApolloPad.com](https://apollopadd.com)

Everyone has a novel in them. Finish

Yours!

<https://apollopadd.com>