|  |  |
| --- | --- |
|  | Chapter 12 Practice Questions – Working With Excel Spreadsheets |
| **Q1** | **What does openpyxl.load\_workbook() function return?** |
| A | This function takes in the filename and returns a value of the workbook data type. This Workbook object represents the Excel file to be manipulated by other functions within the openpyxl module |
| **Q2** | **What does the get\_sheet\_names() workbook method return** |
| A | Assigns the sheet to an object by getting it via its name. This object can then be manipulated e.g. cycle through rows  NOTE: get\_sheet\_names() is now deprecated. Use wb[sheetname] instead |
| **Q3** | **How would you retrieve the Worksheet object for a sheet named ‘Sheet1’?** |
| A | Assign it to an object variable:  Sheet = wb[Sheet1] |
| **Q4** | **How would you retrieve the Worksheet object for the workbook’s active sheet?** |
| A | By using the get\_active\_sheet() method.  NOTE: get\_active\_sheet() method is now deprecated. Use wb.active |
| **Q5** | **How would you retrieve the value in the cell C5?** |
| A | Sheet[‘C5’].value  Or sheet.cell(r=5, column=3).value |
| **Q6** | **How would you set the value in the cell C5 to “Hello”?** |
| A | Sheet[‘C5’] = ‘Hello’ | sheet[‘C5’].value = ‘Hello’  Or sheet.cell(row=5, column=3).value = ‘Hello’ |
| **Q7** | **How would you retrieve the cell’s row and column as integers?** |
| A | To convert from letters to numbers call the openpyxl.cell.column\_index\_from\_string() function  Example:  Column\_index\_from\_string(‘A’) returns 1 |
| **Q8** | **What do the get\_highest\_column() and get\_highest\_row() sheet methods return, and what is the data type of these return values** |
| A | The get\_highest\_row() and get\_highest\_column() both the highest column and highest row in a sheet.  NOTE: The methods are replaced with updated method:  sheet.max\_row  sheet.max\_column |
| **Q9** | **If you needed to get the integer index for column ‘M’, what function would you need to call?** |
| A | First, call the: from openpyxl.cell.column\_index\_from\_string() function  Then, use the method: column\_index\_from\_string(‘M’) |
| **Q10** | **If you needed to get the string name for column 14, what function would you need to call?** |
| A | First, call the: from openpyxl.cell.column\_index\_from\_string() function  Then, use the method: get\_column\_letter(14) |
| **Q11** | **How can you retrieve a tuple of all the cell objects from A1 to F1?** |
| A | A1:F1 is one row  For example:  import openpyxl  wb = openpyxl.load\_workbook(‘example.xlsx’)  sheet = wb[‘Sheet1’]  tuple(sheet[‘A1’:’F1’])  This will contain one tuple for one row in an overall tuple |
| **Q12** | **How would you save the workbook to the filename example.xlsx?** |
| A | wb.save(‘example.xlsx’) |
| **Q13** | **How do you set a formula in a cell?** |
| A | sheet[‘A1’] = ‘SUM(A2:C2)’  As an example.  Formulas, which begin with an equal sign, can configure cells to contain values calculated from other cells |
| **Q14** | **If you want to retrieve the result of a cell’s formula instead of the cell’s formula itself, what must you do first?** |
| A | If you want to see the result of the calculation for the formular instead of the literal formula, you must pass True for the data\_only keyword argument to load\_workbook()  Example:  wbDataOnly = openpyxl.load\_workbook(‘writeFormula.xlsx’, data\_only=True)  sheet[‘A3’].value  i.e. cell A3 has a formula in it |
| **Q15** | **How would you set the height of row 5 to 100?** |
| A | A row height can be set by using the method below:  sheet.row\_dimensions[1].height = 100 |
| **Q16** | **How would you hide column C?** |
| A | Set column C width to 0  sheet.column\_dimensions[‘C’].hidden = True |
| **Q17** | **Name a few features that OpenPyXL 2.1.4 does not load from a spreadsheet file** |
| A | Unfortunately, in the current version of OpenPyXL (2.1.4), the load\_workbook() function does not load charts in Excel files. If the Excel file has charts, the loaded Workbook object will not include them. If you load a Workbook object and immediately save it to the same .xlsx filename, you will effectively remove the charts from it.  Also, it doesn’t load freeze panes, print titles, images |
| **Q18** | **What is a freeze pane?** |
| A | It is the freezing usually of rows or columns that contain headings in order to be able to scroll large data sets whilst the headings remain static |
| **Q19** | **What five functions and methods do you have to call to create a bar chart?** |
| A | 1. Create a Reference object from a rectangular selection of cells (plot data) 2. Create a Series object by passing in the Reference object 3. Create a Chart object 4. Append the Series object to the Chart object 5. Add the Chart object to the Worksheet object 6. openpyxl.charts.Reference() 7. openpyxl.charts.series() 8. openpyxl.charts.Barchart() 9. chartObj.append(seriesObj) 10. add\_chart() |