ASIAN INSTITUTE OF MANAGEMENT

Data Mining and Wrangling

Working with Different Data Formats

Session 3 and 4

BSDSBA 2028

28 January 2026

ASIAN INSTITUTE OF MANAGEMENT

Session 3 and 4 – Working with Different Data Formats

Gameplan

Part 1 – CSV and Excel Files

11:00 AM to 11:10 AM	Class Administrative Matters
11:10 AM to 11:30 AM	Lecture Discussion
11:30 AM to 12:00 NN	Hands-On Coding
12:00 NN to 12:30 PM	In-Class Activity

AIM

Class Administrative Matters

Deliverables

ICA00 – Quick Diagnostics

ICA01 – Who are you? (due 28 Jan 2026 EOD)

R01 - Course Introduction and Review (due 28 Jan 2026 EOD)

Resources

Lecture Materials - https://github.com/aim-bsdsba2028/dmw-2301-lectures

Supplementary Materials - https://github.com/aim-bsdsba2028/dmw-2301-supplementary

Discussion Board

Guides

Git and GitHub Primer

GitHub Classroom

Python Environments

Class Administrative Matters

Course Schedule and Topics

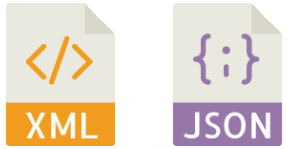
Schedule	Topic	`pandas` concepts
Session 3 and 4	Working with different data types	Indexing and Selection; Essential Functionalities
Session 5 and 6	Regular expressions	Data Preparation and Exploration
Session 7 and 8	Working with databases	Group operations; Data Restructuring
Session 9 and 10	Web scraping	Time-series Functionalities

Data Types for Data Mining

Structured

Semi-structured

Unstructured



AIM

Icons made by Smashicons from www.flaticon.com

Comma-Separated Value (CSV) Format

What is a CSV file?

A text file format used to store tabular data consisting of values separated by commas.

CSV File Syntax:

<header1>,<header2>,<header3>,<header4>

<value1>,<value2>,<value3>,<value4>

<value1>,<value2>,<value3>,<value4>

<value1>,<value2>,<value3>,<value4>

<value1>,<value2>,<value3>,<value4>

<value1>,<value2>,<value3>,<value4>

<value1>,<value2>,<value3>,<value4>

AIM

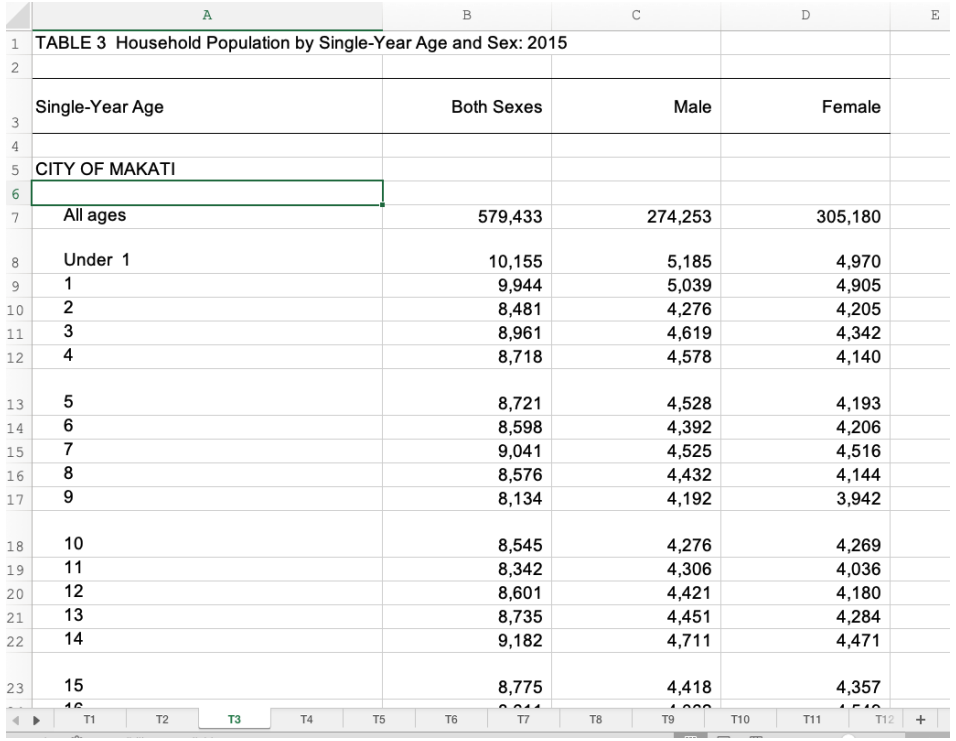
Excel Binary Format (XLS)

What is a XLS file?

A spreadsheet file format used by Microsoft Excel versions 1997 to 2003.

XLS are widely used for storing, analyzing, and sharing spreadsheet data.

Example of an Excel File:



	A	B	C	D	E
1	TABLE 3 Household Population by Single-Year Age and Sex: 2015				
2					
3	Single-Year Age	Both Sexes	Male	Female	
4					
5	CITY OF MAKATI				
6					
7	All ages	579,433	274,253	305,180	
8	Under 1	10,155	5,185	4,970	
9	1	9,944	5,039	4,905	
10	2	8,481	4,276	4,205	
11	3	8,961	4,619	4,342	
12	4	8,718	4,578	4,140	
13	5	8,721	4,528	4,193	
14	6	8,598	4,392	4,206	
15	7	9,041	4,525	4,516	
16	8	8,576	4,432	4,144	
17	9	8,134	4,192	3,942	
18	10	8,545	4,276	4,269	
19	11	8,342	4,306	4,036	
20	12	8,601	4,421	4,180	
21	13	8,735	4,451	4,284	
22	14	9,182	4,711	4,471	
23	15	8,775	4,418	4,357	

T1 T2 T3 T4 T5 T6 T7 T8 T9 T10 T11 T12 +

Loading CSV or XLS into a DataFrame

Data Loading Functions

Indexing

Selecting columns as the DataFrame, and whether to provide headers or indexes

Type inference and data conversion

Define specific functions for data conversions and list of missing value markers

Date and time parsing

Combining capabilities such as combining date and time information from multiple columns

Iterating

Support for reading and iterating over large chunk of files

Data cleaning

Functions for skipping rows, footer, comments, etc.

Text and binary data loading in **pandas**

Function	Description
`read_csv`	Load delimited data from a file, URL, or file-like object; use comma as default delimiter
`read_fwf`	Read data in fixed-width column format (i.e., no delimiters)
`read_clipboard`	Variation of `read_csv` that reads data from the clipboard; useful for converting tables from web pages
`read_excel`	Read tabular data from an Excel XLS or XLSX file
`read_hdf`	Read HDF5 files written by pandas
`read_html`	Read all tables found in the given HTML document
`read_json`	Read data from a JSON (JavaScript Object Notation) string representation, file, URL, or file-like object
`read_feather`	Read the Feather binary file format
`read_orc`	Read the Apache ORC binary file format
`read_parquet`	Read the Apache Parquet binary file format
`read_pickle`	Read an object stored by pandas using the Python pickle format
`read_sas`	Read a SAS dataset stored in one of the SAS system's custom storage formats
`read_spss`	Read a data file created by SPSS
`read_sql`	Read the results of a SQL query (using SQLAlchemy)
`read_sql_table`	Read a whole SQL table (using SQLAlchemy); equivalent to using a query that selects everything in that table using `read_sql`
`read_stata`	Read a dataset from Stata file format
`read_xml`	Read a table of data from an XML file

Wes, McKinney. *Python for data analysis 3rd Edition*. O'Reilly Publishing, 2022.

Session 3 and 4 – Working with Different Data Formats

Gameplan

Part 2 – JSON Files

1:30 PM to 2:00 PM	Lecture + Hand-On Discussion
2:00 PM to 2:45 PM	In-Class Activity
2:45 PM to 3:00 PM	In-Class Activity Discussion

JavaScript Object Notation

What is a JSON file?

A text file with a .json extension that stores data in a human-readable key-value pairs.

An example of a hierarchical data format.

This text file is usually the standard format for APIs, configuration settings.


```
{
  "firstName": "John",
  "lastName": "Smith",
  "isAlive": true,
  "age": 27,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021-3100"
  },
  "phoneNumbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "office",
      "number": "646 555-4567"
    }
  ],
  "children": [],
  "spouse": null
}
```


AIM

JavaScript Object Notation

Reading and Writing of JSON

- `load` – Read JSON from file object
- `dump` – Write JSON into file object
- `loads` – Read JSON from string
- `dumps` – Output JSON as string

JavaScript Object Notation

Reading and Writing of JSON (pandas)

- `pd.read_json` – Read JSON from object, path or string
- `to_json` – Write JSON into file object, path or string

JavaScript Object Notation

Reading and Writing of JSON (using pandas)

`pd.json_normalize` – convert dict or list of dicts into a table

```
pd.json_normalize([
    {"id": 1, "name": {"first": "Coleen", "last": "Volk"}},
    {"name": {"given": "Mark", "family": "Regner"}},
    {"id": 2, "name": "Faye Raker"},
])
```

	id	name.first	name.last	name.given	name.family	name
0	1.0	Coleen	Volk	NaN	NaN	NaN
1	NaN	NaN	NaN	Mark	Regner	NaN
2	2.0	NaN	NaN	NaN	NaN	Faye Raker

Reminders

Resources

Supplementary Materials - https://github.com/aim-bsdsba2028/dmw-2301-supplementary

Deliverables

ICA00 – Quick Diagnostics

ICA01 – Who are you? (due 28 Jan 2026 EOD)

R01 - Course Introduction and Review (due 28 Jan 2026 EOD)

R02 – Data Formats (due 28 Jan 2026 EOD)

ICA02 – Data Formats (due 04 Feb 2026 EOD)

E1 – Data Formats (due 06 Feb 2026 EOD)