**GTSC2143 Machine Learning for Business**

**Tutorial 2**

**Please write down your answers in this document and submit it at iSpace by the end of this tutorial.**

### Pandas Series Operations

1. Creating and Manipulating Series

1. Create a Series with daily temperatures:

* Index: [“Mon”, “Tue”, “Wed”, “Thu”, “Fri”]
* Values: [22, 25, 19, 21, 24]

1. Access Wednesday's temperature and get the first 3 days
2. Find days with temperature above 22 degrees

2. Series Operations and Statistics

1. Convert temperatures from Celsius to Fahrenheit (F = C \* 9/5 + 32)
2. Calculate the mean temperature and find the hottest day

### Pandas DataFrame

1. DataFrame Creation

1. Create a DataFrame using this student data:

student\_data = {

'StudentID': [2001, 2002, 2003, 2004, 2005, 2006],

'Name': ['Alice Wong', 'Bob Chen', 'Carol Lee', 'David Kim', 'Emma Liu', 'Frank Tan'],

'Age': [20, 21, 19, 22, 20, 21],

'Major': ['Computer Science', 'Mathematics', 'Physics', 'Computer Science', 'Mathematics', 'Physics'],

'Math\_Score': [85, 92, 78, 96, 88, 91],

'English\_Score': [78, 85, 92, 82, 91, 86]

}

1. Create the DataFrame and display its shape and first 3 rows
2. Display summary statistics for score columns
3. Show all column names and data types

### DataFrame Operations (Using Student Dataset)

1. Data Selection and Filtering

1. Select only “Name” and “Math\_Score” columns
2. Filter students with Math\_Score > 85
3. Filter Computer Science students
4. Find students whose English\_Score is higher than Math\_Score

2. Data Modification

1. Add a “Total\_Score” column (Math\_Score + English\_Score)
2. Create a “Grade\_Level” column:

* “High” if Total\_Score >= 170
* “Medium” if Total\_Score 150-169
* “Low” if Total\_Score < 150

1. Sort DataFrame by Total\_Score in descending order

3. Grouping and Analysis

1. Group by “Major” and calculate mean Math\_Score for each major
2. Group by “Grade\_Level” and count students in each category

### Data Merging

1. Create Course Dataset

1. Create a second DataFrame using this data:

course\_data = {

'StudentID': [2001, 2002, 2003, 2004, 2005, 2008],

'Course': ['Python', 'Statistics', 'Physics', 'ML', 'Calculus', 'Chemistry'],

'Credits': [3, 4, 3, 4, 4, 3]

}

2. Merge Operations

1. Perform an inner merge on “StudentID”
2. Perform a left merge to keep all students
3. Find students who are enrolled in courses
4. Calculate total credits per student from the merged data

- End of Tutorial 2 -