assign.M1.Assignment.1.covid.data

0.Problem summary

For this problem

- **location:** the country for which the information is provided
- **date:** th

Objectives:

1. **Import and manipulate a .csv file**
2. **Assess your Python Programming Skills**

=> Other assignments are more challenging. Use this to assess your skills.
=> Attempt to solve the problems without searching for online assistance.
=> Prepare questions for class discussion to help source additional tools.

Codebook and data files

a. = article (news, journal)	c. = cheatsheet
	code. = .py or .ipynb
		g = graphic

howTo. = explanandum	py.M. exercise or assignment python file	r = reading

File Name	**Purpose\Description**
https://github.com/cosc-526/cosc.526.home.page/blob/main/code_notebook_cosc_526.ipynb **save your own copy!**	Course Codebook in Jupyter Notebook name = code.notebook.cosc.526.ipynb
	Course github of source data
	i) Kaggle data homepage ii) grab an api key from this page if using that method to import data

Note.1: The codebook is formatted differently, and below highlights expected outcomes.

Note.2: The instructions below are an overview with additional details in the Notebook.

Note.3: Perform your work in your Notebook and generate outcomes for each code block. Export the Notebook as a .pdf for submission. If you have issues generating a .pdf, ensure to submit a .ipynb file at the very minimum.

Mean Imputation with Pandas:

- Use the fillna() method in Pandas to replace missing values with the mean of the column: df.fillna(df.mean(), inplace=True).
- This method replaces missing values with the mean of the corresponding column, providing a simple imputation strategy.

K-Nearest Neighbors (KNN) Imputation with scikit-learn:

- Utilize the KNNImputer class from scikit-learn to impute missing values based on the values of the nearest neighbors: imputer = KNNImputer(n_neighbors=5); imputed_data = imputer.fit_transform(data).
- This approach imputes missing values by considering the values of the k nearest neighbors in the feature space.

Logistic Regression with scikit-learn:

- Import the necessary modules: from sklearn.linear_model import LogisticRegression.
- Create an instance of the LogisticRegression class: logreg = LogisticRegression().

Problem 0 - Import, inspect, and view descriptive statistics

Import data and view descriptive statistics with the pandas library.

- grab data from **Github** URL, .csv. or kaggle api

Problem.1 - Description =>

The United S

Task.0 - Expected outcome:

Problem.2 - Description =>

Which v

Task.0 - Expected outcome:

Problem.3 - Description =>

Which cou

Task.0 - Expected outcome:

Problem.4 - Description =>

Task.0 - Expected outcome:

Problem.5 - Description =>

Determine e

Task.0 - Expected outcome:

Problem.6 - Description =>

Determine

Task.0 - Expected outcome:

Additional resources

- https://github.com/cosc-526/cosc.526.home.page
- Jupyter Community Forum

Additional resources

- need help? Jupyter Community Forum

10. Additional resources

- Anaconda for windows
- Install scientific packages.
- Anaconda installation documentation.
- Jupyter Notebook documentation (including get started guides).
- Jupyter Discourse Forum.
    - Search here for tips, tricks, and solutions.
- Python Package Index (pypi)