**Python:**

1. How can you match a string that starts with an uppercase letter followed by any number of lowercase letters using regular expressions in Python?

Solution- pattern = r’^[A-Z][a-z]\*$’

1. When would you choose to use append() over extend() and vice versa?

Solution- Use **append()** when you want to add a single element to the end of a list.

* Use **extend()** when you want to add multiple elements to the end of a list, such as elements from another list or any iterable.

1. What is the purpose of the Python garbage collector, and how does it reclaim memory from objects that are no longer referenced?

Solution- The Python garbage collector is responsible for automatically reclaiming memory from objects.

1. How can you convert a string to an integer in Python using built-in functions or methods?

Solution- **Using the int() function:**

**Using the str.isdigit() method:**

1. Are there any available programs or libraries to aid with static analysis or problem finding in Python?

Solution- **Pylint**: Pylint is a widely used tool for analyzing Python code for errors, enforcing coding standards, and providing code quality feedback. It can detect errors, potential bugs, unused variables, and more.

**flake8**: Flake8 is a Python linting tool that combines several static analysis tools, including PyFlakes, pycodestyle (formerly known as pep8), and McCabe. It checks Python code against style conventions and detects various issues.

**mypy**: Mypy is a static type checker for Python that can detect type errors and provide type annotations for Python code. It helps catch common programming errors and improves code readability and maintainability.

**Bandit**: Bandit is a security-focused static analysis tool for Python that identifies security vulnerabilities and potential security issues in Python code. It can detect issues such as hardcoded passwords, insecure use of subprocess, and more.

**PyLint-Django**: PyLint-Django is a plugin for Pylint that adds support for Django-specific code analysis. It provides checks and warnings tailored for Django projects, helping ensure adherence to Django best practices and conventions.

1. Create a programme to convert between celsius and fahrenheit values.

Solution-

def celsius\_to\_fahrenheit(celsius):

fahrenheit = (celsius \* 9/5) + 32

return fahrenheit

def fahrenheit\_to\_celsius(fahrenheit):

celsius = (fahrenheit - 32) \* 5/9

return celsius

def main():

choice = input("Choose conversion:\n1. Celsius to Fahrenheit\n2. Fahrenheit to Celsius\nEnter choice (1/2): ")

if choice == '1':

celsius = float(input("Enter temperature in Celsius: "))

fahrenheit = celsius\_to\_fahrenheit(celsius)

print("Temperature in Fahrenheit:", fahrenheit)

elif choice == '2':

fahrenheit = float(input("Enter temperature in Fahrenheit: "))

celsius = fahrenheit\_to\_celsius(fahrenheit)

print("Temperature in Celsius:", celsius)

else:

print("Invalid choice")

rev= main()

**EDA:**

1. Which statistical measure provides information about the spread or variability of a dataset?

Solution- The statistical measure that provides information about the spread or variability of a dataset is the **standard deviation.** The standard deviation is a measure of the dispersion or spread of a set of data points. It quantifies how much the individual data points in a dataset differ from the mean (average) of the dataset. A higher standard deviation indicates that the data points are more spread out from the mean, while a lower standard deviation indicates that the data points are closer to the mean.

1. Which data visualisation technique is used to display the relationship between two numerical variables?

Solution- One of the most common data visualization techniques used to display the relationship between two numerical variables is a **scatter plot**.

1. What is the formula for calculating Kurtosis and which python function is used to get the kurtosis value?

Solution- In Python, you can use the **kurtosis()** function from the **scipy.stats** module to calculate the kurtosis value of a datase

1. What is the formula for calculating skewness ?
2. Why is mean influenced by outliers, but why not median?

Solution- the mean is sensitive to outliers because it takes into account the magnitude of each value, whereas the median is resistant to outliers because it only depends on the order of the values

**Data preprocessing:**

1. How can you check for stationarity in a given time series data using statistical tests or visual inspection?
2. Can you provide examples of issues such as seasonality, trends, or irregularities that need to be addressed during preprocessing?
3. How do you handle missing values in time series data and What are some techniques for imputing missing values in time series data?
4. Explain the impact of outliers on time series analysis and forecasting.
5. What is the significance of smoothing techniques in time series data preprocessing and name some smoothing techniques used?
6. Discuss the concept of window size or lag in moving average smoothing and its impact on the level of smoothing.
7. What are the key components or parameters involved in exponential smoothing?
8. How does weighted moving average smoothing differ from simple moving average smoothing?
9. Explain the process of decomposing a time series into its trend, seasonal, and residual components using STL decomposition.
10. Discuss the metrics or criteria used to assess the performance of different smoothing techniques.