Here's a detailed explanation of your student performance analysis code, along with potential viva questions and answers:

### \*\*Code Explanation\*\*

1. \*\*Loading Data\*\*

```python

df = pd.read\_csv(r"C:\Users\sumit\OneDrive\Desktop\Student\_performance.csv")

```

- Loads student performance dataset from CSV

- Likely contains columns like math\_score, reading\_score, writing\_score etc.

2. \*\*Initial Inspection\*\*

```python

df.isnull().sum() # Checks for missing values

df.dtypes # Shows data types of columns

```

- Helps identify data quality issues

- Confirms numeric type for math\_score

3. \*\*Outlier Detection (IQR Method)\*\*

```python

q1 = df.math\_score.quantile(0.25) # 25th percentile

q3 = df.math\_score.quantile(0.75) # 75th percentile

IQR = q3 - q1 # Interquartile Range

lower\_limit = q1 - 1.5\*IQR

upper\_limit = q3 + 1.5\*IQR

```

- Calculates boundaries for "normal" scores

- Common method to identify statistical outliers

4. \*\*Outlier Filtering\*\*

```python

df[(df.math\_score < lower\_limit) | (df.math\_score > upper\_limit)] # Shows outliers

df[(df.math\_score > lower\_limit) & (df.math\_score < upper\_limit)] # Filters outliers

```

- First line displays outlier records

- Second line keeps only non-outlier records

5. \*\*Z-Score Calculation\*\*

```python

df['zscore'] = (df.math\_score - df.math\_score.mean())/df.math\_score.std()

```

- Measures how many standard deviations each score is from mean

- Helps identify extremely high/low scores

6. \*\*Visualization\*\*

```python

sns.boxplot(df["math\_score"])

```

- Creates boxplot showing:

- Median (middle line)

- IQR (box)

- Whiskers (1.5\*IQR)

- Outliers (dots beyond whiskers)

---

### \*\*Viva Questions & Answers\*\*

#### \*\*Basic Concepts\*\*

\*\*Q1: What is IQR? Why use 1.5 for outlier detection?\*\*

A1:

- IQR = Q3 - Q1 (middle 50% data range)

- 1.5 is a common threshold that identifies ~0.7% of normally distributed data as outliers

- Can adjust to 3 for stricter detection (captures 99.7%)

\*\*Q2: How do z-scores help in outlier detection?\*\*

A2:

- Scores beyond ±3 are extreme outliers (99.7% within ±3)

- Example: z-score of -4 means 4 standard deviations below mean

\*\*Q3: Why check for outliers in student performance data?\*\*

A3:

- May indicate data entry errors

- Could reveal special cases (gifted/struggling students)

- Affects statistical analyses and ML model performance

#### \*\*Technical Questions\*\*

\*\*Q4: What does `quantile(0.25)` calculate?\*\*

A4: The value below which 25% of scores fall (first quartile)

\*\*Q5: How would you handle the outliers?\*\*

A5: Options include:

1. Remove them (if errors)

2. Cap at limits (winsorization)

3. Keep with explanation (if valid)

4. Transform data (log scaling)

\*\*Q6: What does the boxplot show that numbers don't?\*\*

A6:

- Visual distribution shape

- Immediate outlier identification

- Comparison of median vs mean

- Data symmetry/skewness

#### \*\*Advanced Questions\*\*

\*\*Q7: When would you prefer z-score over IQR?\*\*

A7:

- When data is normally distributed

- When you need precise outlier thresholds (e.g., ±2.5σ)

- For comparing across different scaled variables

\*\*Q8: How would this analysis differ for writing scores?\*\*

A8: Would need to:

1. Check different distribution shape

2. Adjust limits if non-normal

3. Consider subject-specific outlier thresholds

\*\*Q9: What statistical assumptions does IQR method make?\*\*

A9:

- No strong normality assumption

- Only assumes ordinal data

- Robust to moderate skewness

---

### \*\*Practical Extensions\*\*

1. \*\*Comparative Analysis\*\*

```python

sns.boxplot(x='test\_preparation\_course', y='math\_score', data=df)

```

- Compare scores between students who did/didn't take prep course

2. \*\*Multivariate Outliers\*\*

```python

from sklearn.ensemble import IsolationForest

clf = IsolationForest().fit(df[['math\_score','reading\_score']])

df['outlier\_flag'] = clf.predict(df[['math\_score','reading\_score']])

```

- Detects outliers considering multiple subjects simultaneously

3. \*\*Outlier Treatment\*\*

```python

# Winsorization example

df['math\_score'] = np.where(df['math\_score'] > upper\_limit, upper\_limit,

np.where(df['math\_score'] < lower\_limit, lower\_limit, df['math\_score']))

```

---

### \*\*Key Takeaways\*\*

1. Always visualize distributions before analysis

2. IQR method is robust for non-normal data

3. Outlier handling depends on context

4. Z-scores standardize comparisons

5. Boxplots provide quick diagnostic overview

Would you like me to demonstrate any specific aspect of this analysis in more detail?