Below is one possible specification that groups the instructions into two main API functions. One function handles both visualization and reporting tasks, while the other focuses solely on visualization. You can adjust or further break down these functions based on your project needs.

**Title of the function**: Function 1: Generate Visualization and Report  
**Function Name**: generate\_visualization\_report  
**ID**: 31\_9, 34\_9, 38\_8, 12\_8, 39\_8  
**Purpose**:  
Creates a set of visualizations (e.g., charts, plots, heat maps, wafer maps) to illustrate semiconductor chip test data analytics—such as yield trends, failure contributions, and test site impacts—and compiles a detailed report summarizing the findings. This function is designed to support workflows where the output includes both visual insight and a comprehensive written summary of the analysis.

**Signature**:

def generate\_visualization\_report(data: pd.DataFrame, viz\_options: dict, report\_options: dict) -> None:

"""

Generates visualizations and compiles a detailed report for semiconductor chip test data analytics.

Parameters:

data (pd.DataFrame): The input dataset containing chip test data.

viz\_options (dict): A dictionary of visualization parameters. This can include:

- 'chart\_types': List of chart types to generate (e.g., 'bar', 'line', 'heatmap', 'wafer\_map').

- 'attributes': Key metrics to visualize (e.g., 'yield\_trends', 'failure\_contributions', 'test\_site\_impacts').

- Other options like color schemes, filters, or grouping criteria.

report\_options (dict): A dictionary of report configuration options, such as:

- 'format': The output format (e.g., PDF, HTML).

- 'sections': List of sections to include (e.g., summary, detailed analysis).

- 'detail\_level': The level of detail required (e.g., 'detailed', 'summary').

Returns:

None

"""

**Used For**:

* **Workflow 31, Instruction 9**: Monitor yield trends, flag lots below target yield, identify responsible test bins/programs, and evaluate if failure rates increase at specific test sites.
* **Workflow 34, Instruction 9**: Create charts and summary reports for yield performance variations due to handler or load board changes, highlighting dominant failure modes when yield drops.
* **Workflow 38, Instruction 8**: Visualize yield improvements and shifts in test bin distributions for lots with less than 93% yield and compile a detailed report of the findings.
* **Workflow 12, Instruction 8**: Generate annotated wafer maps and charts to correlate spatial yield patterns, failure clusters, and test program versions, and compile a comprehensive report to assess potential probing misalignment issues.
* **Workflow 39, Instruction 8**: Create time series and bar charts to track yield trends and the recurring impact of low-performing test bins, followed by a summary report.

**Title of the function**: Function 2: Generate Visualization  
**Function Name**: generate\_visualization  
**ID**: 53\_6, 2\_8, 7\_6, 16\_9, 17\_7, 18\_6, 22\_6, 25\_7, 27\_7, 32\_7  
**Purpose**:  
Produces a variety of visual representations (such as scatter plots, box plots, heat maps, and wafer maps) to reveal trends, patterns, and correlations in semiconductor chip test data. This function is meant for workflows where the output is primarily graphical, assisting in analyses like detecting checkerboard patterns, yield variations, and root-cause relationships without an attached report.

**Signature**:

def generate\_visualization(data: pd.DataFrame, viz\_options: dict) -> None:

"""

Generates visualizations for semiconductor chip test data analytics.

Parameters:

data (pd.DataFrame): The input dataset containing chip test data.

viz\_options (dict): A dictionary of visualization parameters. This can include:

- 'chart\_types': List of chart types to generate (e.g., 'scatter', 'bar', 'box', 'heatmap', 'wafer\_map').

- 'attributes': Metrics or patterns to highlight (e.g., 'yield\_variation', 'checkerboard\_patterns', 'measurement\_variability', 'equipment\_settings').

- Other options such as filters, grouping criteria, and display settings.

Returns:

None

"""

**Used For**:

* **Workflow 53, Instruction 6**: Create visual representations to correlate repeating checkerboard pattern occurrences with test program parameters and calibration intervals.
* **Workflow 2, Instruction 8**: Visualize yield dips using bar charts and scatter plots to show contributions from different test programs and parametric test families.
* **Workflow 7, Instruction 6**: Generate comparative visual plots (e.g., box plots or bar charts) to illustrate wafer-level yield variation by test house.
* **Workflow 16, Instruction 9**: Create time-series plots overlaying yield trends, fab process change events, and key test measurement shifts for visual inspection.
* **Workflow 17, Instruction 7**: Produce visualizations (heat maps, scatter plots, or bar charts) to highlight correlations between abnormal grid/checkerboard patterns and equipment settings.
* **Workflow 18, Instruction 6**: Create scatter plots or heatmaps to illustrate correlations between edge-ring fail patterns and specific test handlers or probers.
* **Workflow 22, Instruction 6**: Generate time-series plots or line charts to inspect the downward trend in yield for lots tested on a specific load board.
* **Workflow 25, Instruction 7**: Create visualizations (wafer maps, bar charts, trend plots) to correlate failing die locations with probe touch order and compare early versus late probing cycles.
* **Workflow 27, Instruction 7**: Generate annotated wafer maps, correlation plots, and frequency charts to reveal recurring spatial patterns that correlate with low yield.
* **Workflow 32, Instruction 7**: Create scatter and box plots to analyze wafer-to-wafer measurement variability and its impact on yield for root cause analysis.

This specification outlines two main functions with clear parameters to handle the diverse set of visualization and reporting requirements in semiconductor chip test data analytics.