

## Multiple Choice Questions and Answers

1. Which Python library provides a declarative API for data visualization?

- a) seaborn
- b) plotly
- c) altair
- d) matplotlib

Answer: c) Altair

2. How do you change the mark or plot type in Altair?

- a) mark()
- b) type()
- c) change\_mark()
- d) set\_type()

Answer: a) mark()

3. Which function is used for global configuration in Altair?

- a) configure()
- b) global\_config()
- c) set\_config()
- d) config()

Answer: a) configure()

4. What do encoding arguments define in Altair?

- a) data structure
- b) visual encoding
- c) plot elements
- d) data types

Answer: b) Visual encoding

5. How do you save Altair charts as files?

- a) save\_chart()

- b) export()
- c) save()
- d) to\_file()

Answer: c) save()

6. Which Altair datatype represents temporal data?

- a) time
- b) temporal
- c) datetime
- d) temporaldata

Answer: c) DateTime

7. What method is used to create titles in Altair?

- a) set\_title()
- b) add\_title()
- c) title()
- d) set\_chart\_title()

Answer: c) title()

8. Which property is used to specify tooltips in Altair?

- a) tooltip()
- b) set\_tooltip()
- c) tooltips()
- d) set\_tooltips()

Answer: d) set\_tooltips()

9. How do you make Altair plots interactive?

- a) set\_interactive()
- b) make\_interactive()
- c) interactive()
- d) enable\_interactive()

Answer: b) make\_interactive()

10. In Altair, what is used to control properties like color, shape, and size of marks?

- a) encoding arguments
- b) configurations
- c) visual settings
- d) mark properties

Answer: a) Encoding arguments

11. What is the purpose of the `mark_point()` method in Altair?

- a) to create points
- b) to create lines
- c) to create bars
- d) to create areas

Answer: a) To create points

12. Which method is used to configure the appearance of a legend in Altair?

- a) `configure_legend()`
- b) `set_legend()`
- c) `legend()`
- d) `configure()`

Answer: b) `set_legend()`

13. What is the default output format for Altair charts?

- a) html
- b) svg
- c) png
- d) json

Answer: a) HTML

14. How do you specify the width and height of an Altair chart?

- a) `set_size()`
- b) `configure_size()`

- c) size()
- d) set\_dimensions()

Answer: c) size()

15. Which method is used to add gridlines to an Altair plot?

- a) add\_gridlines()
- b) grid()
- c) set\_grid()
- d) configure\_grid()

Answer: a) add\_gridlines()

16. How do you set the background color of an Altair chart?

- a) set\_background\_color()
- b) background\_color()
- c) set\_color()
- d) configure\_background()

Answer: a) set\_background\_color()

17. Which Altair method is used to customize axis properties?

- a) customize\_axis()
- b) set\_axis()
- c) axis()
- d) configure\_axis()

Answer: d) configure\_axis()

18. In Altair, how do you specify the scale type for an axis?

- a) set\_scale()
- b) scale()
- c) configure\_scale()
- d) set\_axis\_scale()

Answer: b) scale()

19. What does the transform\_fold() method do in Altair?

- a) unfolds data
- b) folds data
- c) filters data
- d) transforms data

Answer: a) Unfolds data

20. Which method is used to facet a plot in Altair?

- a) facet()
- b) set\_facet()
- c) make\_facet()
- d) configure\_facet()

Answer: a) facet()

21. How do you customize the appearance of facet labels in Altair?

- a) customize\_facet\_labels()
- b) set\_facet\_labels()
- c) facet\_labels()
- d) configure\_facet\_labels()

Answer: b) set\_facet\_labels()

22. What is the purpose of the repeat() method in Altair?

- a) to repeat charts
- b) to repeat data
- c) to repeat encoding
- d) to repeat facets

Answer: c) To repeat encoding

23. Which Altair method is used to create conditional visualizations?

- a) create\_conditionals()
- b) conditional()
- c) make\_conditional()
- d) condition()

Answer: b) conditional()

24. How do you configure the appearance of a tooltip in Altair?

- a) `configure_tooltip()`
- b) `set_tooltip_config()`
- c) `tooltip_config()`
- d) `tooltip_configure()`

Answer: a) `configure_tooltip()`

25. Which method is used to create layered visualizations in Altair?

- a) `layer()`
- b) `create_layer()`
- c) `make_layer()`
- d) `add_layer()`

Answer: a) `layer()`

26. Which library is used for data visualization in Python?

- a) `matplotlib`
- b) `seaborn`
- c) `plotly`
- d) `pandas plotting`

Answer: c) `Plotly`

27. What is the advantage of Plotly over Matplotlib?

- a) interactivity
- b) speed
- c) simplicity
- d) customization

Answer: a) Interactivity

28. What file format can be used to save Plotly visualizations offline?

- a) `png`
- b) `jpeg`

- c) html
- d) csv

Answer: c) HTML

29. Which method is used for adding and updating traces in Plotly?

- a) add\_trace()
- b) update\_trace()
- c) append\_trace()
- d) insert\_trace()

Answer: a) add\_trace()

30. Which module of Plotly is used for creating subplots?

- a) plotly.subplots
- b) plotly.subplots2
- c) plotly.subfigs
- d) plotly.subplot

Answer: a) plotly.subplots

31. What is the purpose of drop-down menus in Plotly?

- a) filter data
- b) add annotations
- c) change colors
- d) rotate plots

Answer: a) Filter data

32. Which feature of Plotly allows for dash interactivity?

- a) dropdown
- b) hover
- c) click
- d) all of the above

Answer: d) All of the above

33. In Plotly, what object is used to define the layout of the plot?

- a) figure
- b) layout
- c) axes
- d) graph

Answer: b) Layout

34. Which method is used to update the layout of a Plotly plot?

- a) `update_layout()`
- b) `edit_layout()`
- c) `modify_layout()`
- d) `change_layout()`

Answer: a) `update_layout()`

35. Which module of Plotly is used for easy and rapid plotting?

- a) plotly express
- b) plotly quick
- c) plotly rapid
- d) plotly ez

Answer: a) Plotly Express

36. What data structure does Plotly Express accept for plotting?

- a) pandas dataframe
- b) numpy array
- c) lists
- d) tuples

Answer: a) Pandas DataFrame

37. Which method is used for creating line plots in Plotly Express?

- a) `px.line()`
- b) `px.scatter()`
- c) `px.bar()`
- d) `px.histogram()`

Answer: a) px.line()

38. What type of plot is created using px.scatter() in Plotly Express?

- a) scatter plot
- b) line plot
- c) bar plot
- d) histogram

Answer: a) Scatter plot

39. Which method is used for creating bar plots in Plotly Express?

- a) px.bar()
- b) px.scatter()
- c) px.line()
- d) px.histogram()

Answer: a) px.bar()

40. Which method is used for creating histograms in Plotly Express?

- a) px.histogram()
- b) px.scatter()
- c) px.line()
- d) px.bar()

Answer: a) px.histogram()

41. What is the purpose of online and offline plotting in Plotly?

- a) online for sharing
- b) offline for speed
- c) both a and b
- d) none of the above

Answer: c) Both A and B

42. What is the structure of a Plotly plot?

- a) json
- b) xml

- c) csv
- d) yaml

Answer: a) JSON

43. Which module in Plotly is used for generating graphs directly from data?

- a) graph objects
- b) graph structures
- c) graph views
- d) graph models

Answer: a) Graph Objects

44. In Plotly, what do graph objects represent?

- a) traces
- b) annotations
- c) shapes
- d) all of the above

Answer: a) Traces

45. What are the two main components of Plotly Express?

- a) figure and axes
- b) plot and layout
- c) data and layout
- d) figure and data

Answer: c) Data and Layout

46. Which module in Plotly is used for updating plots dynamically?

- a) dash
- b) streamlit
- c) bokeh
- d) flask

Answer: a) Dash

47. What is the primary use of Plotly's dropdown menus?

- a) interactivity
- b) aesthetics
- c) animation
- d) navigation

Answer: a) Interactivity

48. Which Plotly module is used for creating complex dashboards?

- a) dash
- b) streamlit
- c) bokeh
- d) flask

Answer: a) Dash

49. Which function is used to add dropdown menus to a Plotly Dash app?

- a) dcc.dropdown()
- b) html.dropdown()
- c) dash.dropdown()
- d) py.dropdown()

Answer: a) dcc.Dropdown()

50. What is the primary purpose of the dcc.Graph() function in Plotly Dash?

- a) display plots
- b) handle callbacks
- c) define layout
- d) import data

Answer: a) Display plots

51. Which method is used to add interactivity to a Plotly Dash app?

- a) callbacks
- b) triggers
- c) bindings
- d) hooks

Answer: a) Callbacks

52. What is the primary purpose of the 'children' property in Plotly Dash components?

- a) define layout
- b) display content
- c) handle callbacks
- d) import data

Answer: b) Display content

53. Which function is used to create interactive components in Plotly Dash?

- a) dcc.component()
- b) html.component()
- c) dash.component()
- d) py.component()

Answer: a) dcc.Component()

54. What is the purpose of Plotly Dash's 'State' object?

- a) store data temporarily
- b) store data permanently
- c) handle callbacks
- d) display content

Answer: a) Store data temporarily

55. Which component in Plotly Dash is used for user input?

- a) dcc.input()
- b) html.input()
- c) dash.input()
- d) py.input()

Answer: a) dcc.Input()

56. What is the purpose of the 'Output' object in Plotly Dash?

- a) define layout

- b) display content
- c) handle callbacks
- d) import data

Answer: b) Display content

57. Which object in Plotly Dash is used to link components together?

- a) callbacks
- b) triggers
- c) bindings
- d) state

Answer: a) Callbacks

58. Which method is used to create a dropdown menu in Plotly Dash?

- a) dcc.dropdown()
- b) html.dropdown()
- c) dash.dropdown()
- d) py.dropdown()

Answer: a) dcc.Dropdown()

59. What is the primary purpose of the 'Input' object in Plotly Dash?

- a) define layout
- b) handle user input
- c) handle callbacks
- d) import data

Answer: b) Handle user input

60. Which module is used for handling interactivity in Plotly Dash?

- a) dash.dependencies
- b) dash.interactivity
- c) dash.interact
- d) dash.interactive

Answer: a) dash.dependencies

61. What is the primary use of Plotly Dash's 'Layout' object?

- a) define layout
- b) display content
- c) handle callbacks
- d) import data

Answer: a) Define layout

62. Which function is used to create callback functions in Plotly Dash?

- a) `@app.callback()`
- b) `@dash.callback()`
- c) `@plotly.callback()`
- d) `@dash_app.callback()`

Answer: a) `@app.callback()`

63. What is the primary purpose of the 'Graph' object in Plotly Dash?

- a) define layout
- b) display plots
- c) handle callbacks
- d) import data

Answer: b) Display plots

64. Which object is used to specify the callback inputs in Plotly Dash?

- a) input
- b) output
- c) state
- d) event

Answer: a) Input

65. What is the primary purpose of the 'dcc' module in Plotly Dash?

- a) user interface components
- b) data visualization
- c) state management

d) layout management

Answer: a) User interface components

66. Which function is used to create layout components in Plotly Dash?

- a) dcc.layout()
- b) html.layout()
- c) dash.layout()
- d) py.layout()

Answer: c) dash.Layout()

67. What is the primary purpose of the 'dash\_html\_components' module in Plotly Dash?

- a) handle callbacks
- b) layout components
- c) state management
- d) data visualization

Answer: b) Layout components

68. Which object is used to specify the callback outputs in Plotly Dash?

- a) output
- b) input
- c) state
- d) event

Answer: a) Output

69. What is the primary purpose of the 'dash\_core\_components' module in Plotly Dash?

- a) user interface components
- b) data visualization
- c) state management
- d) layout management

Answer: a) User interface components

70. Which function is used to create interactive components in Plotly Dash?

- a) dcc.component()
- b) html.component()
- c) dash.component()
- d) py.component()

Answer: a) dcc.Component()

71. What is the primary use of Plotly Dash's 'State' object?

- a) store data temporarily
- b) store data permanently
- c) handle callbacks
- d) display content

Answer: a) Store data temporarily

72. Which component in Plotly Dash is used for user input?

- a) dcc.input()
- b) html.input()
- c) dash.input()
- d) py.input()

Answer: a) dcc.Input()

73. What is the purpose of the 'Output' object in Plotly Dash?

- a) define layout
- b) display content
- c) handle callbacks
- d) import data

Answer: b) Display content

74. Which object in Plotly Dash is used to link components together?

- a) callbacks
- b) triggers
- c) bindings

d) state

Answer: a) Callbacks

75. Which method is used to create a dropdown menu in Plotly Dash?

- a) dcc.dropdown()
- b) html.dropdown()
- c) dash.dropdown()
- d) py.dropdown()

Answer: a) dcc.Dropdown()

76. What is the concept behind CGPlot2/Plotnine?

- a) the grammar of graphics
- b) data visualization
- c) statistical plots
- d) plotting techniques

Answer: a) The Grammar of Graphics

77. Which library emphasizes the Grammar of Graphics principles?

- a) matplotlib
- b) seaborn
- c) plotnine
- d) pandas

Answer: c) Plotnine

78. How can you create plots using CGPlot2/Plotnine?

- a) with high-level commands
- b) through arrays
- c) using html/css
- d) via sql queries

Answer: a) With high-level commands

79. Which aspect of plots can be changed using CGPlot2/Plotnine?

- a) geometries

- b) database queries
- c) file formats
- d) web frameworks

Answer: a) Geometries

80. What is a key feature of CGPlot2/Plotnine regarding stats in visualization?

- a) statistical functions
- b) data sorting
- c) data transformation
- d) dimension reduction

Answer: a) Statistical functions

81. In CGPlot2/Plotnine, what does faceting refer to?

- a) grouping data
- b) data cleaning
- c) data filtering
- d) data transformation

Answer: a) Grouping data

82. What does coordinates manipulation entail in CGPlot2/Plotnine?

- a) adjusting plot dimensions
- b) plot scaling
- c) axis manipulation
- d) plot rotation

Answer: c) Axis manipulation

83. How can annotations be incorporated in CGPlot2/Plotnine?

- a) overlaying text
- b) changing colors
- c) adjusting sizes
- d) rotating plots

Answer: a) Overlaying text

84. What is scaling in the context of CGPlot2/Plotnine?

- a) adjusting data ranges
- b) plot resizing
- c) axis formatting
- d) data transformation

Answer: a) Adjusting data ranges

85. What role do themes play in CGPlot2/Plotnine?

- a) visual style presets
- b) data analysis tools
- c) geometric shapes
- d) data sources

Answer: a) Visual style presets

86. Which component of CGPlot2/Plotnine handles the visual representation of data categories?

- a) legends
- b) axes
- c) titles
- d) annotations

Answer: a) Legends

87. What do palettes represent in CGPlot2/Plotnine?

- a) color schemes
- b) geometric shapes
- c) statistical tests
- d) data distributions

Answer: a) Color schemes

88. How does CGPlot2/Plotnine facilitate the creation of visualizations?

- a) through a layered approach
- b) using sql queries

- c) using html/css
- d) through file formats

Answer: a) Through a layered approach

89. In CGPlot2/Plotnine, what is the primary focus when changing geoms?

- a) altering plot structures
- b) modifying colors
- c) adjusting sizes
- d) transforming data

Answer: a) Altering plot structures

90. Which library shares similarities with CGPlot2/Plotnine regarding the Grammar of Graphics?

- a) seaborn
- b) numpy
- c) tensorflow
- d) keras

Answer: a) Seaborn

91. How can stats be integrated into CGPlot2/Plotnine visualizations?

- a) statistical functions
- b) plot formatting
- c) data filtering
- d) html/css customization

Answer: a) Statistical functions

92. What is the significance of faceting in CGPlot2/Plotnine?

- a) grouping data subsets
- b) data cleaning
- c) data sorting
- d) data transformation

Answer: a) Grouping data subsets

93. Which feature of coordinates manipulation aids in data clarity in CGPlot2/Plotnine?

- a) adjusting axis positions
- b) plot scaling
- c) plot rotation
- d) changing colors

Answer: a) Adjusting axis positions

94. How can annotations improve the interpretability of CGPlot2/Plotnine visualizations?

- a) providing additional context
- b) adjusting colors
- c) changing sizes
- d) grouping data

Answer: a) Providing additional context

95. Why is scaling important in CGPlot2/Plotnine?

- a) ensures data visibility
- b) enhances aesthetics
- c) improves speed
- d) increases plot size

Answer: a) Ensures data visibility

96. What distinguishes themes in CGPlot2/Plotnine?

- a) visual style presets
- b) statistical models
- c) data sources
- d) plot resolutions

Answer: a) Visual style presets

97. How do legends aid in understanding visualizations in CGPlot2/Plotnine?

- a) represent data categories

- b) adjust plot sizes
- c) modify axis limits
- d) change color schemes

Answer: a) Represent data categories

98. What is the primary function of palettes in CGPlot2/Plotnine?

- a) define color schemes
- b) generate statistical plots
- c) facilitate data transformation
- d) change plot structures

Answer: a) Define color schemes

99. How does CGPlot2/Plotnine approach visualization creation?

- a) layered approach
- b) sql queries
- c) html/css styling
- d) file import/export

Answer: a) Layered approach

100. What is emphasized when changing geoms in CGPlot2/Plotnine?

- a) plot structure alteration
- b) data manipulation
- c) color modification
- d) statistical analysis

Answer: a) Plot structure alteration

101. In addition to CGPlot2/Plotnine, which library also follows the Grammar of Graphics principles?

- a) seaborn
- b) TensorFlow
- c) keras
- d) PyTorch

Answer: a) Seaborn

102. What role do stats play in CGPlot2/Plotnine visualizations?

- a) provide analytical insights
- b) adjust plot colors
- c) modify plot sizes
- d) filter data

Answer: a) Provide analytical insights

103. How does faceting contribute to understanding data in CGPlot2/Plotnine?

- a) groups data subsets
- b) rearranges data
- c) cleans data
- d) sorts data

Answer: a) Groups data subsets

104. What does adjusting axis positions accomplish in CGPlot2/Plotnine?

- a) enhances data clarity
- b) adjusts plot sizes
- c) changes color schemes
- d) modifies legend styles

Answer: a) Enhances data clarity

105. How do annotations enhance the interpretability of CGPlot2/Plotnine visualizations?

- a) provide additional information
- b) change plot sizes
- c) adjust plot colors
- d) modify legend styles

Answer: a) Provide additional information

106. Why is scaling crucial in CGPlot2/Plotnine visualizations?

- a) ensures data fits the plot

- b) enhances aesthetics
- c) modifies data types
- d) increases plot resolution

Answer: a) Ensures data fits the plot

107. What do themes offer in CGPlot2/Plotnine?

- a) visual style presets
- b) statistical models
- c) data transformations
- d) plot resizing

Answer: a) Visual style presets

108. How do legends contribute to understanding CGPlot2/Plotnine visualizations?

- a) represent data categories
- b) modify plot colors
- c) adjust plot sizes
- d) change axis limits

Answer: a) Represent data categories

109. What is the primary function of palettes in CGPlot2/Plotnine?

- a) define color schemes
- b) generate statistical plots
- c) facilitate data transformation
- d) change plot structures

Answer: a) Define color schemes

110. How does CGPlot2/Plotnine approach visualization creation?

- a) layered approach
- b) sql queries
- c) html/css styling
- d) file import/export

Answer: a) Layered approach

111. What is emphasized when changing geoms in CGPlot2/Plotnine?

- a) plot structure alteration
- b) data manipulation
- c) color modification
- d) statistical analysis

Answer: a) Plot structure alteration

112. In addition to CGPlot2/Plotnine, which library also follows the Grammar of Graphics principles?

- a) seaborn
- b) TensorFlow
- c) keras
- d) PyTorch

Answer: a) Seaborn

113. What role do stats play in CGPlot2/Plotnine visualizations?

- a) provide analytical insights
- b) adjust plot colors
- c) modify plot sizes
- d) filter data

Answer: a) Provide analytical insights

114. How does faceting contribute to understanding data in CGPlot2/Plotnine?

- a) groups data subsets
- b) rearranges data
- c) cleans data
- d) sorts data

Answer: a) Groups data subsets

115. What does adjusting axis positions accomplish in CGPlot2/Plotnine?

- a) enhances data clarity

- b) adjusts plot sizes
- c) changes color schemes
- d) modifies legend styles

Answer: a) Enhances data clarity

116. How do annotations enhance the interpretability of CGPlot2/Plotnine visualizations?

- a) provide additional information
- b) change plot sizes
- c) adjust plot colors
- d) modify legend styles

Answer: a) Provide additional information

117. Why is scaling crucial in CGPlot2/Plotnine visualizations?

- a) ensures data fits the plot
- b) enhances aesthetics
- c) modifies data types
- d) increases plot resolution

Answer: a) Ensures data fits the plot

118. What do themes offer in CGPlot2/Plotnine?

- a) visual style presets
- b) statistical models
- c) data transformations
- d) plot resizing

Answer: a) Visual style presets

119. How do legends contribute to understanding CGPlot2/Plotnine visualizations?

- a) represent data categories
- b) modify plot colors
- c) adjust plot sizes

d) change axis limits

Answer: a) Represent data categories

120. What is the primary function of palettes in CGPlot2/Plotnine?

- a) define color schemes
- b) generate statistical plots
- c) facilitate data transformation
- d) change plot structures

Answer: a) Define color schemes

121. How does CGPlot2/Plotnine approach visualization creation?

- a) layered approach
- b) sql queries
- c) html/css styling
- d) file import/export

Answer: a) Layered approach

122. What is emphasized when changing geoms in CGPlot2/Plotnine?

- a) plot structure alteration
- b) data manipulation
- c) color modification
- d) statistical analysis

Answer: a) Plot structure alteration

123. In addition to CGPlot2/Plotnine, which library also follows the Grammar of Graphics principles?

- a) seaborn
- b) TensorFlow
- c) keras
- d) PyTorch

Answer: a) Seaborn

124. What role do stats play in CGPlot2/Plotnine visualizations?

- a) provide analytical insights
- b) adjust plot colors
- c) modify plot sizes
- d) filter data

Answer: a) Provide analytical insights

125. How does faceting contribute to understanding data in CGPlot2/Plotnine?

- a) groups data subsets
- b) rearranges data
- c) cleans data
- d) sorts data

Answer: a) Groups data subsets

