I cannot directly create or convert files into a single PDF. However, I can combine the text content from all the images into one cohesive block of text for you.

The images contain a list of **100 Apex Trigger Interview Questions You Can't Ignore in 2025!**

Combined List of Apex Trigger Interview Questions

Basic Trigger Practice Questions

1. Write a trigger that prevents an Opportunity from being deleted if the Opportunity Stage is "Closed Won."
2. Write a trigger to ensure that the Account's "AnnualRevenue" is updated every time an Opportunity is closed.
3. Create a trigger to prevent the creation of duplicate Contact records based on Email.
4. Write a trigger that ensures a custom "Status" field on Account is updated when an Opportunity is closed.
5. Implement a trigger that calculates the total value of Opportunities related to an Account and updates a custom field on the Account.
6. Write a trigger to automatically populate the "Lead Source" field on an Account when a new Contact is created.
7. Write a trigger that prevents an Opportunity from being created if the related Account has a "Status" of "Inactive."
8. Create a trigger that updates a custom field on Opportunity when a related Contact is updated.
9. Write a trigger that automatically sends an email when an Opportunity is closed-won and its Amount exceeds a certain value.
10. Create a trigger to update a custom field on the Account when a related Opportunity is closed.

Intermediate Trigger Practice Questions

11. Write a trigger that prevents creating a duplicate Opportunity for the same Account and Opportunity Name.
12. Create a trigger that updates an Account's "Priority" when an Opportunity reaches a "Closed Won" stage.
13. Implement a trigger that populates a custom "Total Value" field on Account, summing up the "Amount" of related Opportunities.
14. Write a trigger that prevents the deletion of a Contact if there is an active Opportunity related to that Contact.
15. Write a trigger to ensure that the Opportunity 'Close Date' is always updated to a future date.

16. Write a trigger that sets the Opportunity's 'Close Date' to today's date if it is not provided.
17. Create a trigger to ensure that the Opportunity's "Amount" is never lower than the Account's "AnnualRevenue."
18. Write a trigger to update a custom field on Opportunity when the Account's "AnnualRevenue" is updated.
19. Write a trigger to prevent creating an Opportunity if the related Account is marked as "Inactive."
20. Create a trigger to update the related Account's "Phone" field when a Contact's phone number is changed.

Advanced Trigger Practice Questions

21. Write a trigger that ensures only one Opportunity per Account is marked as "Closed Won."
22. Implement a trigger that updates the "Lead Source" on the Account when an Opportunity's "Stage" is updated.
23. Write a trigger to send an email notification when an Opportunity's "Stage" is changed to "Closed Won."
24. Implement a trigger that updates a custom "Revenue Forecast" field on Account when a related Opportunity is updated.
25. Write a trigger that tracks changes to critical fields on Opportunities and stores those changes in a custom object.
26. Create a trigger to track the number of Opportunities in each stage and store the count in a custom field on the Account.
27. Implement a trigger that checks if the Opportunity's "Amount" exceeds a predefined value, and if so, notifies the Opportunity owner.
28. Create a trigger that checks if an Opportunity's "Close Date" is more than 30 days in the future and raises an error.
29. Write a trigger that updates the related Account's "AnnualRevenue" based on the sum of closed Opportunities' amounts.
30. Implement a trigger to prevent changes to Opportunity Amount if the Opportunity is in a "Closed" stage.

Bulkified Trigger Questions

31. Write a bulkified trigger to update a custom field on Account for all related Opportunities that are closed.
32. Create a bulkified trigger that prevents updates to Opportunities if their related Account is marked as "Inactive."
33. Write a bulkified trigger to ensure that Opportunities are only created in bulk if the related Account is active.

34. Implement a bulkified trigger to update the "Status" field on Account based on related Opportunities' stages.
35. Write a bulkified trigger that prevents the deletion of Accounts that have Opportunities marked as "Closed Won."
36. Implement a bulkified trigger to update an Account's custom field with the total value of "Closed Opportunity."
37. Write a bulkified trigger that sends email alerts for all Opportunities with a specific value, closing within the next 7 days.
38. Create a bulkified trigger that ensures Opportunities are not created with invalid Close Dates.
39. Write a bulkified trigger that updates a custom field on Account every time any related Opportunity is updated.
40. Implement a bulkified trigger to prevent changes to an Opportunity if it is associated with an Account marked as "Inactive."

Complex Business Logic Questions

41. Write a trigger to automatically create a Task when an Opportunity is moved to the "Closed Won" stage.
42. Create a trigger to ensure that the Opportunity "Amount" is greater than or equal to the Account's "AnnualRevenue."
43. Write a trigger to prevent the deletion of a Contact if there are related Opportunities in the "Open" stage.
44. Implement a trigger to update the related Account's "Priority" when an Opportunity's "Amount" exceeds a certain threshold.
45. Write a trigger that creates a new Opportunity for an Account when a Contact's "Status" is set to "Active."
46. Create a trigger that ensures Opportunities are automatically closed if their related Account's "Status" is set to "Inactive."
47. Write a trigger that calculates and updates the total revenue from all related Opportunities for an Account.
48. Write a trigger to prevent an Opportunity from being marked as "Closed Lost" if the associated Account is active.
49. Create a trigger that prevents updates to Opportunity fields after they are marked "Closed Won."
50. Write a trigger that updates the "Last Modified" date on the related Account whenever an Opportunity is updated.

Trigger for Handling Recursive Operations

51. Implement a trigger to prevent recursive updates on the Account when a related Opportunity is updated.

52. Write a trigger that prevents recursive updates when creating or updating an Opportunity related to the same Account.
53. Implement a static variable to prevent a trigger from being fired recursively when updating Opportunity records.
54. Create a trigger to ensure that the recursive updates on Accounts are prevented by using a static flag.
55. Write a trigger to prevent multiple Opportunity updates when the Opportunity's related Account's fields are updated.

Data Integrity and Validation Questions

56. Write a trigger that validates the "Close Date" on Opportunities to ensure they are within a certain range.
57. Implement a trigger that prevents an Opportunity from being saved if the "Amount" is set to a negative number.
58. Create a trigger that ensures the "Amount" on Opportunities cannot be updated once the Opportunity is closed.
59. Write a trigger to validate that the Opportunity's "Close Date" cannot be set to a past date.
60. Write a trigger that checks for duplicate Opportunities based on the Opportunity Name and Account and prevents insertion.

Trigger for Handling Master-Detail Relationships

61. Create a trigger that ensures the deletion of a parent Account record will cascade and delete its related Opportunity records.
62. Write a trigger to update all child Opportunity records when the Account's "Industry" field is updated.
63. Implement a trigger that automatically updates all related child Opportunities when a parent Account is deleted.
64. Write a trigger to prevent the deletion of a parent Account record if it has related Opportunities in the "Closed Won" stage.
65. Implement a trigger that ensures updates to a parent Account's field are reflected in its child Opportunity records.

Trigger for Handling Lookup Relationships

66. Write a trigger to ensure that the "Phone" field on an Account is updated when the associated Opportunity is updated.
67. Create a trigger that updates related Opportunities when an Account's custom field is updated.
68. Implement a trigger that prevents the creation of an Opportunity if the associated Account is not active.

69. Write a trigger that prevents the deletion of a Contact if it is associated with an active Opportunity.
70. Implement a trigger that updates the related Opportunity when the associated Account is marked as "Closed."

Trigger for Handling Multiple Objects

71. Write a trigger that updates the custom field on Account and Opportunity when a Contact's "Status" changes.
72. Create a trigger that updates related Opportunities, Accounts, and Contacts in the same transaction.
73. Write a trigger that updates the related Account's "AnnualRevenue" field when multiple Opportunities are closed-won.
74. Implement a trigger that ensures a custom "Priority" field on Account is updated whenever a related Opportunity is closed-won.
75. Write a trigger to update the "Stage" field on Opportunities when a related Account's "Status" is updated.

Performance Optimization in Triggers

76. Implement a trigger that minimizes the number of SOQL queries inside loops.
77. Write a trigger that uses collections (lists, maps) to avoid hitting governor limits on large datasets.
78. Create a trigger that avoids using multiple DML statements within a loop by grouping DML operations.
79. Write a trigger that handles bulk updates efficiently by avoiding repeated SOQL queries.
80. Implement a trigger that ensures bulk updates are done in a single operation to improve performance.

Trigger Unit Testing and Debugging

81. Write a test class for a trigger that checks if Opportunities are being updated correctly when a related account is modified.
82. Create a unit test that verifies the bulk processing of Opportunities without hitting governor limits.
83. Write a unit test to ensure that recursive triggers are prevented using static variables.
84. Implement a test class that ensures a trigger fires when certain conditions are met (e.g., Opportunity is closed-won).
85. Write unit test methods that simulate the creation and updating of records to validate trigger behavior under different scenarios.

Edge Case Handling in Triggers

86. Write a trigger that prevents an Opportunity from being saved if the "Close Date" is in the past and the "Stage" is "Closed Won."
87. Implement a trigger to handle situations where an Account has no related Opportunities.
88. Create a trigger that handles scenarios where an Opportunity's "Amount" is set to 0.
89. Implement a trigger that handles cases where a Contact has no related Opportunity.
90. Implement a trigger that accounts for updates to Opportunity records that have no associated Account.

Miscellaneous Trigger Scenarios

91. Write a trigger to create a new Opportunity if a Contact's "Status" is updated to "Active."
92. Implement a trigger to update the Opportunity's "Close Date" based on the related Account's "Close Date."
93. Write a trigger that prevents updating the Opportunity if the associated Account has a different "Industry."
94. Create a trigger that sets the Opportunity's "Next Step" to a default value when it's newly created.
95. Write a trigger that ensures no duplicate Contacts can be created with the same email address and related Account.
96. Write a trigger that updates an Account's "CustomField" when a related Opportunity's "Stage" is updated.
97. Create a trigger to ensure Opportunities cannot be marked as "Closed Lost" if the related Account is "Active."
98. Write a trigger to calculate the "Opportunity Age" based on the "Close Date" and set it in a custom field.
99. Create a trigger that ensures that Opportunities are not created for inactive Accounts.
100. Write a trigger that automatically assigns a custom "Region" value to Opportunities based on their related Account's Region.