



Comprehensive Interview Questions List: SQL, GitHub/Git & Programming Basics

Based on extensive research, here's a complete compilation of **303 interview questions** covering **SQL**, **GitHub/Git**, and **Programming Fundamentals** for tech interviews.

SQL Interview Questions (87 Questions)

Basic SQL Questions (23 Questions)

Database Fundamentals

1. What is SQL? [\[1\]](#) [\[2\]](#) [\[3\]](#)
2. What is a database? [\[3\]](#) [\[4\]](#) [\[5\]](#)
3. What are the main types of SQL commands? [\[4\]](#) [\[6\]](#) [\[7\]](#)
4. What is DDL, DML, DCL, and TCL? [\[2\]](#) [\[1\]](#) [\[4\]](#)
5. What is a primary key? [\[2\]](#) [\[3\]](#) [\[4\]](#)
6. What is a foreign key? [\[3\]](#) [\[4\]](#) [\[2\]](#)
7. What is the difference between SQL and NoSQL databases? [\[8\]](#) [\[5\]](#) [\[4\]](#)
8. What are the different data types in SQL? [\[9\]](#) [\[1\]](#) [\[3\]](#)
9. What is a table and a field in SQL? [\[5\]](#) [\[1\]](#) [\[4\]](#)
10. What are constraints in SQL? [\[6\]](#) [\[1\]](#) [\[4\]](#)
11. What is normalization? [\[1\]](#) [\[8\]](#) [\[4\]](#)
12. What is a NULL value? [\[8\]](#) [\[4\]](#) [\[2\]](#)
13. What is the difference between CHAR and VARCHAR? [\[10\]](#) [\[6\]](#) [\[1\]](#)
14. What is the SELECT statement? [\[4\]](#) [\[1\]](#) [\[3\]](#)
15. What is the WHERE clause? [\[11\]](#) [\[1\]](#) [\[4\]](#)
16. What is the ORDER BY clause? [\[11\]](#) [\[6\]](#) [\[1\]](#)
17. What is the DISTINCT keyword? [\[1\]](#) [\[2\]](#) [\[4\]](#)
18. What are aliases in SQL? [\[11\]](#) [\[4\]](#) [\[1\]](#)
19. What is the BETWEEN operator? [\[4\]](#) [\[1\]](#) [\[11\]](#)
20. What is the LIKE operator? [\[1\]](#) [\[11\]](#) [\[4\]](#)
21. What is the IN operator? [\[5\]](#) [\[11\]](#) [\[4\]](#)

22. What are wildcard characters in SQL? [\[11\]](#) [\[4\]](#) [\[1\]](#)

23. What is the difference between COUNT(*) and COUNT(column_name)? [\[6\]](#) [\[4\]](#) [\[1\]](#)

Intermediate SQL Questions (25 Questions)

Joins and Advanced Queries

24. What are the different types of joins in SQL? [\[6\]](#) [\[4\]](#) [\[11\]](#)

25. Explain INNER JOIN with example [\[4\]](#) [\[6\]](#) [\[11\]](#)

26. What is LEFT JOIN (LEFT OUTER JOIN)? [\[6\]](#) [\[11\]](#) [\[4\]](#)

27. What is RIGHT JOIN (RIGHT OUTER JOIN)? [\[11\]](#) [\[4\]](#) [\[6\]](#)

28. What is FULL OUTER JOIN? [\[8\]](#) [\[4\]](#) [\[11\]](#)

29. What is CROSS JOIN? [\[8\]](#) [\[4\]](#) [\[11\]](#)

30. What is SELF JOIN? [\[1\]](#) [\[4\]](#) [\[11\]](#)

31. What are aggregate functions? [\[4\]](#) [\[6\]](#) [\[1\]](#)

32. Explain GROUP BY clause [\[6\]](#) [\[1\]](#) [\[4\]](#)

33. What is the HAVING clause? [\[11\]](#) [\[4\]](#) [\[6\]](#)

34. What is the difference between WHERE and HAVING? [\[4\]](#) [\[6\]](#) [\[11\]](#)

35. What are subqueries? [\[1\]](#) [\[11\]](#) [\[4\]](#)

36. What is a correlated subquery? [\[8\]](#) [\[11\]](#) [\[4\]](#)

37. What is the difference between correlated and nested subqueries? [\[8\]](#) [\[11\]](#) [\[4\]](#)

38. What is UNION and UNION ALL? [\[2\]](#) [\[8\]](#) [\[4\]](#)

39. What is the difference between UNION and UNION ALL? [\[2\]](#) [\[8\]](#) [\[4\]](#)

40. What are indexes? [\[1\]](#) [\[8\]](#) [\[4\]](#)

41. What is the difference between clustered and non-clustered indexes? [\[6\]](#) [\[8\]](#) [\[4\]](#)

42. What are views? [\[2\]](#) [\[8\]](#) [\[4\]](#)

43. What is a stored procedure? [\[2\]](#) [\[8\]](#) [\[4\]](#)

44. What is a function in SQL? [\[5\]](#) [\[8\]](#) [\[4\]](#)

45. What is the difference between stored procedure and function? [\[5\]](#) [\[8\]](#) [\[4\]](#)

46. What are triggers? [\[8\]](#) [\[2\]](#) [\[4\]](#)

47. What is a cursor? [\[2\]](#) [\[4\]](#) [\[8\]](#)

48. What is a CTE (Common Table Expression)? [\[11\]](#) [\[4\]](#) [\[8\]](#)

Advanced SQL Questions (24 Questions)

Performance and Complex Queries

49. What are window functions? [\[4\]](#) [\[8\]](#) [\[11\]](#)

50. Explain ROW_NUMBER(), RANK(), and DENSE_RANK() [\[12\]](#) [\[8\]](#) [\[4\]](#)

51. What is PARTITION BY? [\[12\]](#) [\[8\]](#) [\[4\]](#)

52. What are materialized views? [\[8\]](#) [\[11\]](#) [\[4\]](#)

53. What is database partitioning? [\[11\]](#) [\[4\]](#) [\[8\]](#)

54. What is a transaction? [\[5\]](#) [\[4\]](#) [\[8\]](#)

55. What are ACID properties? [\[5\]](#) [\[4\]](#) [\[8\]](#)

56. What is deadlock? [\[5\]](#) [\[4\]](#) [\[8\]](#)

57. How do you handle deadlocks? [\[4\]](#) [\[5\]](#) [\[8\]](#)

58. What is query optimization? [\[5\]](#) [\[8\]](#) [\[4\]](#)

59. How do you optimize slow queries? [\[8\]](#) [\[4\]](#) [\[5\]](#)

- 60. What is SQL injection? [\[2\]](#) [\[4\]](#) [\[8\]](#)
- 61. How do you prevent SQL injection? [\[2\]](#) [\[4\]](#) [\[8\]](#)
- 62. What is the execution plan? [\[4\]](#) [\[5\]](#) [\[8\]](#)
- 63. What are temporary tables? [\[5\]](#) [\[8\]](#) [\[4\]](#)
- 64. What is the difference between DELETE, TRUNCATE, and DROP? [\[13\]](#) [\[1\]](#) [\[4\]](#)
- 65. What is a database backup? [\[8\]](#) [\[4\]](#) [\[5\]](#)
- 66. What are different types of backups? [\[4\]](#) [\[5\]](#) [\[8\]](#)
- 67. What is database replication? [\[5\]](#) [\[8\]](#) [\[4\]](#)
- 68. What is sharding? [\[8\]](#) [\[4\]](#) [\[5\]](#)
- 69. What are recursive queries? [\[2\]](#) [\[4\]](#) [\[8\]](#)
- 70. What is the MERGE statement? [\[11\]](#) [\[4\]](#) [\[8\]](#)
- 71. What are regular expressions in SQL? [\[11\]](#) [\[4\]](#) [\[8\]](#)
- 72. What are temporal tables? [\[11\]](#) [\[4\]](#) [\[8\]](#)

Practical SQL Questions (15 Questions)

Real-World Query Problems

- 73. Write a query to find duplicate records [\[1\]](#) [\[4\]](#) [\[11\]](#)
- 74. Write a query to find the second highest salary [\[1\]](#) [\[4\]](#) [\[11\]](#)
- 75. Write a query to find employees with salary greater than their manager [\[1\]](#) [\[4\]](#) [\[11\]](#)
- 76. Write a query to find top 5 customers by total order amount [\[4\]](#) [\[8\]](#) [\[11\]](#)
- 77. Write a query to calculate running total [\[12\]](#) [\[8\]](#) [\[4\]](#)
- 78. Write a query to find employees who joined in the last 6 months [\[1\]](#) [\[11\]](#) [\[4\]](#)
- 79. Write a query to list departments with more than 10 employees [\[1\]](#) [\[11\]](#) [\[4\]](#)
- 80. Write a query to find common records between two tables [\[11\]](#) [\[4\]](#) [\[1\]](#)
- 81. Write a query to remove duplicate records [\[4\]](#) [\[1\]](#) [\[11\]](#)
- 82. Write a query to calculate average salary by department [\[1\]](#) [\[11\]](#) [\[4\]](#)
- 83. Write a query to find the difference in days between two dates [\[8\]](#) [\[11\]](#) [\[4\]](#)
- 84. Write a query to pivot data [\[12\]](#) [\[8\]](#) [\[4\]](#)
- 85. Write a query to unpivot data [\[12\]](#) [\[8\]](#) [\[4\]](#)
- 86. Write a query to find nth highest salary [\[11\]](#) [\[4\]](#) [\[1\]](#)
- 87. Write a query to find employees without managers [\[4\]](#) [\[1\]](#) [\[11\]](#)

GitHub/Git Interview Questions (95 Questions)

Basic Git Questions (20 Questions)

Version Control Fundamentals

- 88. What is Git? [\[14\]](#) [\[15\]](#) [\[16\]](#)
- 89. What is version control? [\[17\]](#) [\[7\]](#) [\[14\]](#)
- 90. What is a repository in Git? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 91. What is the difference between Git and GitHub? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 92. What is the difference between Git and SVN? [\[14\]](#) [\[17\]](#) [\[7\]](#)
- 93. How do you initialize a Git repository? [\[18\]](#) [\[15\]](#) [\[7\]](#)
- 94. What is a commit in Git? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 95. How do you clone a repository? [\[7\]](#) [\[18\]](#) [\[15\]](#)

- 96. What is the staging area? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 97. What is the working directory? [\[14\]](#) [\[15\]](#) [\[18\]](#)
- 98. What is HEAD in Git? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 99. What are the three states of files in Git? [\[19\]](#) [\[18\]](#) [\[15\]](#)
- 100. What is git config command? [\[7\]](#) [\[18\]](#) [\[15\]](#)
- 101. How do you check the status of files? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 102. How do you add files to staging area? [\[14\]](#) [\[15\]](#) [\[18\]](#)
- 103. How do you commit changes? [\[15\]](#) [\[18\]](#) [\[14\]](#)
- 104. How do you view commit history? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 105. What is .gitignore file? [\[14\]](#) [\[15\]](#) [\[18\]](#)
- 106. How do you create a .gitignore file? [\[15\]](#) [\[18\]](#) [\[14\]](#)
- 107. What is git diff command? [\[18\]](#) [\[14\]](#) [\[15\]](#)

Intermediate Git Questions (20 Questions)

Branching and Collaboration

- 108. What is branching in Git? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 109. How do you create a new branch? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 110. How do you switch between branches? [\[14\]](#) [\[15\]](#) [\[7\]](#)
- 111. What is merging in Git? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 112. What is the difference between merge and rebase? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 113. What is a merge conflict? [\[14\]](#) [\[15\]](#) [\[7\]](#)
- 114. How do you resolve merge conflicts? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 115. What is git pull? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 116. What is git fetch? [\[14\]](#) [\[15\]](#) [\[7\]](#)
- 117. What is the difference between git pull and git fetch? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 118. What is git push? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 119. What is a remote repository? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 120. How do you add a remote repository? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 121. What is forking? [\[14\]](#) [\[15\]](#) [\[18\]](#)
- 122. What is a pull request? [\[20\]](#) [\[18\]](#) [\[14\]](#)
- 123. What is cherry-picking? [\[7\]](#) [\[15\]](#) [\[14\]](#)
- 124. What is git stash? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 125. How do you unstage files? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 126. How do you undo the last commit? [\[7\]](#) [\[15\]](#) [\[18\]](#)
- 127. What is git reset? [\[14\]](#) [\[15\]](#) [\[7\]](#)

Advanced Git Questions (20 Questions)

Complex Git Operations

- 128. What is git rebase? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 129. When should you use rebase vs merge? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 130. What is interactive rebase? [\[14\]](#) [\[15\]](#) [\[7\]](#)
- 131. What is a detached HEAD? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 132. How do you fix a detached HEAD? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 133. What is git reflog? [\[14\]](#) [\[15\]](#) [\[7\]](#)
- 134. What are Git hooks? [\[21\]](#) [\[16\]](#) [\[14\]](#)

- 135. What is git bisect? [\[18\]](#) [\[15\]](#) [\[14\]](#)
- 136. How do you revert a commit? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 137. What is the difference between git reset and git revert? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 138. What are Git objects? [\[14\]](#) [\[15\]](#) [\[7\]](#)
- 139. What is the Git object model? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 140. How does Git store data? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 141. What is the .git directory? [\[14\]](#) [\[15\]](#) [\[7\]](#)
- 142. What is git gc? [\[16\]](#) [\[15\]](#) [\[14\]](#)
- 143. What are tags in Git? [\[15\]](#) [\[7\]](#) [\[14\]](#)
- 144. How do you create and delete tags? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 145. What is git archive? [\[16\]](#) [\[14\]](#) [\[15\]](#)
- 146. What is git submodule? [\[16\]](#) [\[14\]](#) [\[15\]](#)
- 147. What are Git workflows? [\[21\]](#) [\[14\]](#) [\[7\]](#)

GitHub-Specific Questions (20 Questions)

GitHub Platform Features

- 148. What is GitHub? [\[20\]](#) [\[21\]](#) [\[14\]](#)
- 149. What are GitHub Actions? [\[17\]](#) [\[20\]](#) [\[21\]](#)
- 150. What is a workflow in GitHub Actions? [\[20\]](#) [\[21\]](#) [\[17\]](#)
- 151. What are runners in GitHub Actions? [\[21\]](#) [\[17\]](#) [\[20\]](#)
- 152. What are artifacts in GitHub Actions? [\[17\]](#) [\[20\]](#) [\[21\]](#)
- 153. How do you manage secrets in GitHub Actions? [\[20\]](#) [\[21\]](#) [\[17\]](#)
- 154. What is GitHub Pages? [\[22\]](#) [\[21\]](#) [\[14\]](#)
- 155. What are GitHub Issues? [\[22\]](#) [\[21\]](#) [\[14\]](#)
- 156. What is GitHub Projects? [\[22\]](#) [\[21\]](#) [\[14\]](#)
- 157. What are GitHub Releases? [\[21\]](#) [\[22\]](#) [\[14\]](#)
- 158. What is a GitHub Wiki? [\[22\]](#) [\[21\]](#) [\[14\]](#)
- 159. What are GitHub templates? [\[17\]](#) [\[21\]](#) [\[22\]](#)
- 160. What is branch protection in GitHub? [\[21\]](#) [\[22\]](#) [\[17\]](#)
- 161. What are GitHub Apps? [\[22\]](#) [\[17\]](#) [\[21\]](#)
- 162. What is GitHub API? [\[17\]](#) [\[21\]](#) [\[22\]](#)
- 163. What are reusable workflows? [\[20\]](#) [\[21\]](#) [\[17\]](#)
- 164. What is the difference between GitHub-hosted and self-hosted runners? [\[20\]](#) [\[21\]](#) [\[17\]](#)
- 165. How do you define workflow triggers? [\[21\]](#) [\[17\]](#) [\[20\]](#)
- 166. What are jobs and steps in GitHub Actions? [\[17\]](#) [\[20\]](#) [\[21\]](#)
- 167. How do you use GitHub for CI/CD? [\[20\]](#) [\[21\]](#) [\[17\]](#)

Practical Git Scenarios (15 Questions)

Real-World Problem Solving

- 168. How do you undo the last commit but keep changes? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 169. How do you completely remove a file from Git history? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 170. How do you rename a branch? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 171. How do you delete a branch locally and remotely? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 172. How do you squash commits? [\[14\]](#) [\[15\]](#) [\[7\]](#)
- 173. How do you move commits from one branch to another? [\[15\]](#) [\[18\]](#) [\[14\]](#)

- 174. How do you recover deleted files? [\[18\]](#) [\[14\]](#) [\[15\]](#)
- 175. How do you find which commit introduced a bug? [\[14\]](#) [\[15\]](#) [\[18\]](#)
- 176. How do you change the commit message of the last commit? [\[15\]](#) [\[18\]](#) [\[14\]](#)
- 177. How do you split a commit into multiple commits? [\[7\]](#) [\[14\]](#) [\[15\]](#)
- 178. How do you handle large files in Git? [\[16\]](#) [\[21\]](#) [\[14\]](#)
- 179. How do you set up Git aliases? [\[16\]](#) [\[14\]](#) [\[15\]](#)
- 180. How do you configure Git for multiple users? [\[16\]](#) [\[14\]](#) [\[15\]](#)
- 181. How do you backup a Git repository? [\[16\]](#) [\[14\]](#) [\[15\]](#)
- 182. How do you migrate from SVN to Git? [\[17\]](#) [\[16\]](#) [\[14\]](#)

Programming Basics Questions (121 Questions)

Fundamental Concepts (22 Questions)

Programming Foundation

- 183. What is computer programming? [\[23\]](#) [\[24\]](#) [\[9\]](#)
- 184. What is a programming language? [\[23\]](#) [\[9\]](#) [\[10\]](#)
- 185. What is the difference between high-level and low-level languages? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 186. What is an algorithm? [\[25\]](#) [\[24\]](#) [\[23\]](#)
- 187. What are the characteristics of a good algorithm? [\[26\]](#) [\[25\]](#) [\[23\]](#)
- 188. What is a flowchart? [\[25\]](#) [\[26\]](#) [\[23\]](#)
- 189. What is pseudocode? [\[26\]](#) [\[23\]](#) [\[25\]](#)
- 190. What is a compiler? [\[10\]](#) [\[23\]](#) [\[9\]](#)
- 191. What is an interpreter? [\[23\]](#) [\[9\]](#) [\[10\]](#)
- 192. What is the difference between compiler and interpreter? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 193. What is machine code? [\[10\]](#) [\[23\]](#) [\[9\]](#)
- 194. What is assembly language? [\[23\]](#) [\[9\]](#) [\[10\]](#)
- 195. What is source code? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 196. What is object code? [\[10\]](#) [\[23\]](#) [\[9\]](#)
- 197. What is debugging? [\[26\]](#) [\[23\]](#) [\[9\]](#)
- 198. What is testing? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 199. What is the difference between syntax error and logic error? [\[27\]](#) [\[23\]](#) [\[9\]](#)
- 200. What is runtime error? [\[27\]](#) [\[23\]](#) [\[9\]](#)
- 201. What is an IDE? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 202. What is a framework? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 203. What is a library? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 204. What is an API? [\[28\]](#) [\[23\]](#) [\[24\]](#)

Data Structures (20 Questions)

Data Organization Concepts

- 205. What is a data structure? [\[26\]](#) [\[23\]](#) [\[24\]](#)
- 206. What is an array? [\[23\]](#) [\[24\]](#) [\[26\]](#)
- 207. What is a linked list? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 208. What is the difference between array and linked list? [\[26\]](#) [\[23\]](#) [\[24\]](#)
- 209. What is a stack? [\[23\]](#) [\[24\]](#) [\[26\]](#)

- 210. What is LIFO? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 211. What is a queue? [\[26\]](#) [\[23\]](#) [\[24\]](#)
- 212. What is FIFO? [\[23\]](#) [\[24\]](#) [\[26\]](#)
- 213. What is a tree? [\[28\]](#) [\[24\]](#) [\[23\]](#)
- 214. What is a binary tree? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 215. What is a binary search tree? [\[26\]](#) [\[23\]](#) [\[24\]](#)
- 216. What is a graph? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 217. What is a hash table? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 218. What is the difference between linear and non-linear data structures? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 219. What is a multi-dimensional array? [\[10\]](#) [\[23\]](#) [\[24\]](#)
- 220. What is a string? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 221. What are the advantages of linked lists over arrays? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 222. What is a circular linked list? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 223. What is a doubly linked list? [\[23\]](#) [\[24\]](#) [\[28\]](#)
- 224. What is recursion? [\[25\]](#) [\[24\]](#) [\[23\]](#)

Programming Constructs (22 Questions)

Language Building Blocks

- 225. What is a variable? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 226. What is a constant? [\[10\]](#) [\[9\]](#) [\[23\]](#)
- 227. What are data types? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 228. What is type casting? [\[10\]](#) [\[9\]](#) [\[23\]](#)
- 229. What is an operator? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 230. What are arithmetic operators? [\[10\]](#) [\[23\]](#) [\[9\]](#)
- 231. What are logical operators? [\[23\]](#) [\[9\]](#) [\[10\]](#)
- 232. What are relational operators? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 233. What is a conditional statement? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 234. What is an if statement? [\[26\]](#) [\[23\]](#) [\[9\]](#)
- 235. What is a switch statement? [\[10\]](#) [\[23\]](#) [\[9\]](#)
- 236. What is a loop? [\[24\]](#) [\[23\]](#) [\[9\]](#)
- 237. What is a for loop? [\[23\]](#) [\[9\]](#) [\[10\]](#)
- 238. What is a while loop? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 239. What is a do-while loop? [\[10\]](#) [\[23\]](#) [\[9\]](#)
- 240. What is the difference between while and do-while loop? [\[23\]](#) [\[9\]](#) [\[10\]](#)
- 241. What is a function? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 242. What is a method? [\[26\]](#) [\[24\]](#) [\[23\]](#)
- 243. What is the difference between function and method? [\[24\]](#) [\[10\]](#) [\[23\]](#)
- 244. What are parameters and arguments? [\[10\]](#) [\[23\]](#) [\[9\]](#)
- 245. What is function overloading? [\[23\]](#) [\[9\]](#) [\[10\]](#)
- 246. What is recursion? [\[25\]](#) [\[24\]](#) [\[23\]](#)

Advanced Concepts (22 Questions)

Programming Paradigms

- 247. What is object-oriented programming? [\[26\]](#) [\[24\]](#) [\[23\]](#)
- 248. What is procedural programming? [\[24\]](#) [\[26\]](#) [\[23\]](#)
- 249. What is functional programming? [\[26\]](#) [\[24\]](#) [\[23\]](#)
- 250. What is a class? [\[24\]](#) [\[10\]](#) [\[23\]](#)
- 251. What is an object? [\[10\]](#) [\[23\]](#) [\[24\]](#)
- 252. What is inheritance? [\[23\]](#) [\[24\]](#) [\[10\]](#)
- 253. What is polymorphism? [\[24\]](#) [\[10\]](#) [\[23\]](#)
- 254. What is encapsulation? [\[10\]](#) [\[23\]](#) [\[24\]](#)
- 255. What is abstraction? [\[23\]](#) [\[24\]](#) [\[10\]](#)
- 256. What is a pointer? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 257. What is memory management? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 258. What is garbage collection? [\[24\]](#) [\[10\]](#) [\[23\]](#)
- 259. What is exception handling? [\[9\]](#) [\[10\]](#) [\[23\]](#)
- 260. What is multithreading? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 261. What is concurrency? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 262. What is a design pattern? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 263. What is modularity? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 264. What is coupling? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 265. What is cohesion? [\[28\]](#) [\[23\]](#) [\[24\]](#)
- 266. What is the software development lifecycle? [\[26\]](#) [\[23\]](#) [\[24\]](#)
- 267. What is version control? [\[17\]](#) [\[23\]](#) [\[24\]](#)
- 268. What is code review? [\[26\]](#) [\[23\]](#) [\[24\]](#)

Algorithms & Searching/Sorting (20 Questions)

Problem-Solving Techniques

- 269. What is binary search? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 270. What is linear search? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 271. What is sorting? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 272. What is bubble sort? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 273. What is selection sort? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 274. What is insertion sort? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 275. What is merge sort? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 276. What is quick sort? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 277. What is heap sort? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 278. What is the time complexity of different sorting algorithms? [\[25\]](#) [\[23\]](#) [\[24\]](#)
- 279. What is space complexity? [\[23\]](#) [\[25\]](#) [\[24\]](#)
- 280. What is Big O notation? [\[25\]](#) [\[24\]](#) [\[23\]](#)
- 281. What is the difference between best case and worst case? [\[24\]](#) [\[23\]](#) [\[25\]](#)
- 282. What is the Fibonacci sequence? [\[23\]](#) [\[25\]](#) [\[24\]](#)
- 283. How do you implement Fibonacci using recursion? [\[25\]](#) [\[24\]](#) [\[23\]](#)
- 284. What is dynamic programming? [\[24\]](#) [\[23\]](#) [\[25\]](#)
- 285. What is greedy algorithm? [\[23\]](#) [\[25\]](#) [\[24\]](#)
- 286. What is divide and conquer? [\[25\]](#) [\[24\]](#) [\[23\]](#)

287. What is breadth-first search?[\[24\]](#) [\[23\]](#) [\[25\]](#)

288. What is depth-first search?[\[23\]](#) [\[25\]](#) [\[24\]](#)

Practical Programming (15 Questions)

Coding Problem Solutions

289. How do you reverse a string?[\[27\]](#) [\[24\]](#) [\[23\]](#)

290. How do you check if a string is a palindrome?[\[26\]](#) [\[24\]](#) [\[23\]](#)

291. How do you find the largest element in an array?[\[26\]](#) [\[24\]](#) [\[23\]](#)

292. How do you find duplicates in an array?[\[26\]](#) [\[24\]](#) [\[23\]](#)

293. How do you remove duplicates from an array?[\[26\]](#) [\[24\]](#) [\[23\]](#)

294. How do you count vowels in a string?[\[24\]](#) [\[26\]](#) [\[23\]](#)

295. How do you check if two strings are anagrams?[\[26\]](#) [\[23\]](#) [\[24\]](#)

296. How do you implement a calculator?[\[27\]](#) [\[23\]](#) [\[26\]](#)

297. How do you find the factorial of a number?[\[25\]](#) [\[23\]](#) [\[24\]](#)

298. How do you check if a number is prime?[\[23\]](#) [\[25\]](#) [\[24\]](#)

299. How do you generate prime numbers?[\[25\]](#) [\[24\]](#) [\[23\]](#)

300. How do you find the GCD of two numbers?[\[24\]](#) [\[23\]](#) [\[25\]](#)

301. How do you find the LCM of two numbers?[\[23\]](#) [\[25\]](#) [\[24\]](#)

302. How do you implement a simple sorting algorithm?[\[25\]](#) [\[24\]](#) [\[23\]](#)

303. How do you implement binary search?[\[24\]](#) [\[23\]](#) [\[25\]](#)

Interview Preparation Tips

For SQL Interviews:

- Practice **writing queries** on real databases[\[12\]](#) [\[1\]](#) [\[4\]](#)
- Understand **database design** and normalization[\[5\]](#) [\[8\]](#) [\[4\]](#)
- Know **performance optimization** techniques[\[5\]](#) [\[8\]](#) [\[4\]](#)
- Practice **complex scenarios** with multiple tables[\[1\]](#) [\[11\]](#) [\[4\]](#)

For Git/GitHub Interviews:

- Demonstrate **hands-on experience** with common workflows[\[21\]](#) [\[18\]](#) [\[14\]](#)
- Understand **branching strategies** and collaboration[\[21\]](#) [\[18\]](#) [\[14\]](#)
- Know **troubleshooting** and problem-solving scenarios[\[18\]](#) [\[14\]](#) [\[15\]](#)
- Practice **GitHub Actions** and CI/CD concepts[\[20\]](#) [\[21\]](#) [\[17\]](#)

For Programming Basics:

- Master **fundamental concepts** before advanced topics[\[26\]](#) [\[23\]](#) [\[24\]](#)
- Practice **coding problems** regularly[\[26\]](#) [\[23\]](#) [\[25\]](#)
- Understand **different programming paradigms**[\[23\]](#) [\[24\]](#) [\[26\]](#)
- Know **data structures and algorithms**[\[25\]](#) [\[24\]](#) [\[23\]](#)

This comprehensive collection of **303 questions** covers all essential topics for technical interviews in software development, database management, and version control systems.



1. <https://learninglabbb.com/sql-interview-questions-for-freshers/>
2. <https://www.interviewbit.com/sql-interview-questions/>
3. <https://www.geeksforgeeks.org/sql/sql-interview-questions/>
4. <https://www.simplilearn.com/top-sql-interview-questions-and-answers-article>
5. <https://www.ccbp.in/blog/articles/sql-interview-questions-for-data-analyst>
6. <https://www.youtube.com/watch?v=oX5Y26O5dBE>
7. <https://www.simplilearn.com/tutorials/git-tutorial/git-interview-questions>
8. <https://upesonline.ac.in/blog/advanced-sql-interview-questions>
9. <https://www.geeksforgeeks.org/c/c-interview-questions/>
10. <https://www.interviewbit.com/c-interview-questions/>
11. <https://roadmap.sh/questions/sql-queries>
12. <https://www.youtube.com/watch?v=nlbsOKG36qk>
13. <https://www.semanticscholar.org/paper/27791a955c431e9b3e39da95260eabb05e2fe3ea>
14. <https://www.geeksforgeeks.org/git/git-interview-questions-and-answers/>
15. <https://www.interviewbit.com/git-interview-questions/>
16. <https://www.vinsys.com/blog/git-interview-questions>
17. <https://razorops.com/blog/top-50-version-control-systems-interview-question-and-answers/>
18. <https://www.linkedin.com/pulse/interview-questions-answers-git-automation-testers-kushal-parikh-oqgpf>
19. <https://web.talenttitan.com/candidates/interview-questions/version-control-interview-questions>
20. <https://razorops.com/blog/top-50-github-actions-interview-question-and-answers/>
21. <https://github.com/NotHarshhaa/DevOps-Interview-Questions>
22. <https://github.com/DopplerHQ/awesome-interview-questions>
23. <https://www.testgorilla.com/blog/programming-skills-interview-questions/>
24. <https://www.simplilearn.com/coding-interview-questions-article>
25. <https://www.geeksforgeeks.org/interview-prep/practice-for-cracking-any-coding-interview/>
26. <https://pigmancareers.uky.edu/blog/2024/08/04/45-common-coding-interview-questions/>
27. <https://www.youtube.com/watch?v=xqMIZ4f6-7A>
28. <https://www.adaface.com/blog/computer-science-fundamentals-interview-questions/>