Short Questions

1. What is the historical perspective of database system applications?

2. Compare and contrast file systems with a Database Management System (DBMS).

3. What is the significance of the data model in database systems?

4. Define the levels of abstraction in a DBMS.

5. Explain data independence in databases.

6. What is the typical structure of a DBMS?

7. How does database design relate to ER diagrams?

8. Define entities and attributes in a database.

9. What is an entity set, and how is it different from an entity?

10. Explain the concept of relationships in a database.

11. Define relationship sets and provide an example.

12. What are additional features of the ER model?

13. How is conceptual design achieved using the ER model?

14. Describe the process of designing a database.

15. What is the role of ER diagrams in the design process?

16. Define cardinality in the context of relationships.

17. Explain the difference between a weak and a strong entity.

18. How does normalization contribute to database design?

19. What is the purpose of a primary key in a database table?

20. Describe the importance of foreign keys in relational databases.

21. Explain the concept of a schema in a DBMS.

22. Define data redundancy and how it is minimized in databases.

23. What is the role of indexing in a database system?

24. Differentiate between horizontal and vertical partitioning in databases.

25. How does a DBMS ensure data integrity?
26. Define and provide an example of a functional dependency.
27. What is the purpose of a transaction in a database?
28. Explain the concept of atomicity in transactions.
29. Describe the ACID properties of a transaction.
30. What is a view in a database, and how is it useful?
31. Explain the concept of data warehousing.
32. Define OLAP and OLTP in the context of databases.
33. What is a stored procedure, and how is it used in databases?
34. Describe the role of triggers in database systems.
35. What is a data dictionary, and why is it essential?
36. How do databases handle concurrent transactions?
37. Explain the concept of a data warehouse and its benefits.
38. What is the role of metadata in a database system?
39. Define the term denormalization and its purpose.
40. How does a DBMS support data security and access control?
41. Explain the concept of a composite key in a database table.
42. What is the difference between a superkey and a candidate key?
43. Describe the role of the entity-relationship model in database design.
44. Define the terms tuple and attribute in a relational database.
45. How does the normalization process contribute to data organization?
46. Explain the concept of referential integrity.
47. What is the purpose of a data warehouse in business intelligence?
48. Describe the concept of data mining in databases.
50. Define the term query optimization in the context of databases.

51. What is the relational model in database management?
52. Who is credited with the development of the relational model?
53. What are relations in the relational model?
54. How are relations represented in tabular form?
55. What is the key concept of the relational model regarding data organization?
56. What are integrity constraints in a relational database?
57. Provide examples of common integrity constraints.
58. How do integrity constraints ensure data accuracy and consistency?
59. Explain the role of primary keys in enforcing integrity constraints.
60. What is referential integrity, and why is it important?
61. How do you enforce integrity constraints in a relational database?
62. What happens when an integrity constraint is violated?
63. Describe the purpose of cascading actions in enforcing referential integrity.
64. What is a foreign key, and how does it relate to enforcing constraints?
65. How can you ensure that data meets domain constraints?
66. What is a SQL query, and how is it used to retrieve data from a relational database?
67. Explain the SELECT statement in SQL.
68. What are the basic components of a SQL query?
69. How do you specify filtering conditions in SQL queries?
70. What is the purpose of the ORDER BY clause in SQL?
71. What is logical database design, and how does it differ from physical design?
72. How do you identify entities and relationships during logical database design?
73. What is the role of normalization in logical database design?
74. Why is denormalization sometimes necessary in database design?
75. What are functional dependencies, and how are they related to normalization?

76. Explain the concepts of first, second, and third normal forms (1NF, 2NF, 3NF).

77. How does the process of normalization help reduce data redundancy?

78. What is the purpose of a view in a relational database?

79. How can you create a view in SQL?

80. What are the advantages of using views in database management?

81. How do views provide data abstraction and security?

82. What are the potential drawbacks of using views?

83. How can you delete a table in a relational database?

84. What precautions should be taken when destroying a table with valuable data?

85. How do you alter the structure of an existing table in SQL?

86. What are the differences between DROP and DELETE statements in SQL?

87. What is the role of relational algebra in database operations?

88. Explain the concept of projection in relational algebra.

89. How is the selection operation used to filter data in relational algebra?

90. Describe the difference between union and intersection operations in relational algebra.

91. What is the purpose of the join operation in relational algebra?

92. How does the Cartesian product differ from a natural join?

93. What is the closure property in the context of relational algebra?

94. How is the division operation used in relational algebra?

95. Explain the concepts of tuple and domain relational calculus.

96. What is the primary difference between tuple and domain relational calculus?

97. How can you express a query using tuple relational calculus?

98. Provide an example of a query expressed in domain relational calculus.

99. What are the advantages of using relational calculus for query formulation?

100. How do relational algebra and relational calculus relate to SQL in database query languages?

101. What is SQL, and what does it stand for?

102. How do you retrieve all records from a table named "Employees"?

103. What is the purpose of the WHERE clause in an SQL query?

104. How do you retrieve records where the "Salary" is greater than 50000 from a table named "Employees"?

105. What SQL command is used to add new records to a table?

106. How can you update existing records in an SQL table?

107. What SQL command is used to remove records from a table?

108. Explain the purpose of the UNION operator in SQL.

109. What does the INTERSECT operator do in SQL?

110. What is the purpose of the EXCEPT operator in SQL?

111. What are nested queries in SQL, and why are they used?

112. How do you use the COUNT() aggregation operator to count the number of records in a table?

113. What is the purpose of the GROUP BY clause in SQL?

114. How do you retrieve the highest value from a column named "Price" in an SQL table?

115. What does the SQL NULL value represent?

116. How do you check for NULL values in an SQL query?

117. Explain the concept of primary keys and their importance in SQL tables.

118. What are foreign keys, and how do they relate to primary keys in SQL tables?

119. What are SQL constraints, and why are they used?

120. How do you create a new table with SQL constraints for columns?

121. What are triggers in SQL, and how are they used?

122. How do you define a trigger in SQL?

123. What is the difference between an "AFTER" and "BEFORE" trigger in SQL?

124. Explain the concept of an "INSTEAD OF" trigger in SQL.

125. How do you drop or remove a trigger in SQL?