105 Entity Framework Interview Questions for Hiring Top Talent

Questions

1. What is Entity Framework, in simple words, and why do we use it?

2. Can you explain the difference between Code First, Database First, and Model First approaches in Entity Framework?

3. What is an entity in Entity Framework?

4. How do you connect your application to a database using Entity Framework?

5. What is a DbContext in Entity Framework, and what is its purpose?

6. How do you perform basic CRUD (Create, Read, Update, Delete) operations using Entity Framework?

7. What are migrations in Entity Framework, and why are they useful?

8. Explain what LINQ is and how it is used with Entity Framework.

9. What is lazy loading in Entity Framework, and what are its advantages and disadvantages?

10. How would you include related data in your queries using Entity Framework (e.g., loading a blog post with its comments)?

11. What is the purpose of the 'Include' method in Entity Framework?

12. What is connection string and how it is important in Entity Framework?

13. What is the difference between 'Single' , 'First' and 'FirstOrDefault'?

14. What is an 'Entity Set' and 'Entity Type'?

15. What is the use of 'SqlQuery'?

16. Can you describe the role of DbSet in Entity Framework?

17. How do you handle exceptions or errors that might occur during database operations with Entity Framework?

18. What are some advantages of using Entity Framework over traditional ADO.NET?

19. Explain what the 'SaveChanges' method does in Entity Framework.

20. How do you configure relationships between entities in Entity Framework (e.g., one-to-many, many-to-many)?

21. What is eager loading and how does it differ from lazy loading?

22. How can you improve the performance of Entity Framework queries?

23. What is Object Relational Mapping (ORM) and how does Entity Framework implement it?

24. What is shadow property in EF core?

25. What do you understand about Entity Framework interceptors?

26. Explain about the concept of 'context pooling'.

27. What are the benefits of using 'dotnet ef' command line tools?

28. What is connection resiliency?

29. What are some common configuration options you can set in the DbContext?

30. Explain the difference between 'AsNoTracking()' and 'AsNoTrackingWithIdentityResolution()'.

31. What is Entity Framework and why do we use it?

32. Can you describe the difference between Code First and Database First approaches in Entity Framework?

33. What is a DbSet in Entity Framework?

34. How do you install Entity Framework in a .NET project?

35. What is LINQ and how is it used with Entity Framework?

36. Explain the concept of Migrations in Entity Framework.

37. What is the purpose of the SaveChanges() method in Entity Framework?

38. How can you query data from a database using Entity Framework and LINQ?

39. What is the difference between Single() and FirstOrDefault() when querying with LINQ and Entity Framework?

40. How do you configure a connection string for Entity Framework?

41. What is the purpose of the DbContext class in Entity Framework?

42. Can you explain what eager loading is in Entity Framework?

43. What is lazy loading, and what are its potential drawbacks?

44. Describe how to add a new entity to the database using Entity Framework.

45. How do you update an existing entity in the database?

46. How do you delete an entity from the database?

47. What is the benefit of using an ORM like Entity Framework?

48. What are some common data annotations used with Entity Framework?

49. Explain how to handle relationships between tables (e.g., one-to-many) using Entity Framework.

50. How can you improve the performance of Entity Framework queries?

51. What is the role of primary keys and foreign keys in database relationships when using Entity Framework?

52. Describe a situation where you might need to use raw SQL queries with Entity Framework.

53. What is the purpose of the AsNoTracking() method in Entity Framework, and when would you use it?

54. How can you optimize Entity Framework queries to improve performance when dealing with large datasets? Explain the different strategies.

55. Describe the concept of connection pooling in Entity Framework and its benefits. How can you configure connection pooling?

56. What is the difference between eager loading, lazy loading, and explicit loading in Entity Framework? When would you use each approach?

57. Explain how you would handle concurrency conflicts in Entity Framework. Describe different concurrency resolution strategies.

58. What are the advantages and disadvantages of using stored procedures with Entity Framework? How do you map stored procedures to entities?

59. How can you implement soft delete functionality in Entity Framework without physically deleting data from the database?

60. Describe the use of interceptors in Entity Framework. What are some common use cases for interceptors?

61. Explain how you would implement auditing in Entity Framework to track changes made to entities.

62. What are the different ways to execute raw SQL queries in Entity Framework? When would you choose to use raw SQL?

63. How can you use Entity Framework with dependency injection? Explain the benefits of using dependency injection.

64. Describe how you would handle database migrations in a team environment using Entity Framework Core. How do you resolve merge conflicts?

65. What are the best practices for handling database contexts in an ASP.NET Core application using Entity Framework Core?

66. How can you improve the performance of Entity Framework queries using compiled queries? Explain the benefits and drawbacks.

67. Explain how you would implement a custom validation attribute in Entity Framework to enforce specific business rules.

68. Describe how you can use Entity Framework to work with different database providers, such as SQL Server, PostgreSQL, and MySQL.

69. What are the different ways to configure relationships between entities in Entity Framework? Provide examples of one-to-one, one-to-many, and many-to-many relationships.

70. How would you implement a repository pattern on top of Entity Framework? Discuss the benefits and drawbacks.

71. Explain the concept of shadow properties in Entity Framework. When might you use shadow properties?

72. What are owned entities in Entity Framework Core? How do they differ from regular entities, and what are their use cases?

73. How can you implement optimistic concurrency control in Entity Framework? Explain the purpose of the Timestamp attribute.

74. Describe the different strategies for seeding data in Entity Framework Core. What are the advantages and disadvantages of each approach?

75. How can you use Entity Framework Core to interact with a database view? Explain the steps involved.

76. Explain how you would configure global query filters in Entity Framework Core. What are some common use cases for global query filters?

77. Describe the process of using the Include method with multiple levels of related data. How does this impact performance?

78. What are value converters in Entity Framework Core? Explain how they are used and provide an example.

79. How can you implement database-generated GUID values in Entity Framework? Discuss the configuration options.

80. Explain how you would handle complex data types (e.g., JSON columns) in Entity Framework Core using value conversions or other techniques.

81. Describe how you can use Entity Framework to implement multi-tenancy in a database. What are some different approaches?

82. How would you diagnose and resolve common Entity Framework performance issues, such as slow queries or excessive database round trips?

83. Explain the difference between using AsNoTracking() and AsNoTrackingWithIdentityResolution() in Entity Framework Core. When is each appropriate?

84. How can you optimize Entity Framework queries to improve performance in a high-traffic application?

85. Describe a scenario where you would choose to use stored procedures over raw SQL queries with Entity Framework and why?

86. Explain the implications of using AsNoTracking() in Entity Framework and when it would be beneficial?

87. How do you handle concurrency conflicts in Entity Framework, and what strategies can you employ to resolve them?

88. What are the advantages and disadvantages of using Lazy Loading, Eager Loading, and Explicit Loading in Entity Framework?

89. Can you explain the role of the DbContext lifecycle and how it impacts performance and data consistency?

90. How would you implement a soft delete strategy using Entity Framework?

91. Describe a situation where you might need to drop down to ADO.NET from Entity Framework and how you would approach it.

92. How can you improve the startup time of an Entity Framework application with a large database model?

93. Explain how you can use database views with Entity Framework and what the benefits are.

94. How would you go about debugging performance issues within Entity Framework, detailing the tools and techniques used?

95. Describe a real-world project where you used Entity Framework and the specific challenges you encountered.

96. How can you ensure data integrity and consistency when working with multiple DbContext instances?

97. What are the different change tracking mechanisms in Entity Framework and how do they work?

98. How do you manage database migrations in a team environment using Entity Framework Core?

99. Explain different ways to handle relationships (one-to-one, one-to-many, many-to-many) in Entity Framework Core, detailing their configuration and usage.

100. How can you effectively use Entity Framework with asynchronous programming patterns?

101. Describe your experience with testing Entity Framework code, including mocking DbContext and DbSet objects.

102. What are the potential security vulnerabilities associated with using Entity Framework, and how can you mitigate them?

103. Explain your understanding of connection pooling in Entity Framework and its impact on performance.