

## Core Data, Sqlite and other Persistence stores

No.	Question	Priority
1	<p>What is persistence Storage?</p> <ol style="list-style-type: none"> <li>1. What are different persistence storage available in iOS?</li> <li>2. Different type of GCD?</li> </ol>	
2	<p>What is Sqlite?</p> <ol style="list-style-type: none"> <li>1. How to do Sqlite implementation in your app.</li> <li>2. If Sqlite is thread-safe?</li> </ol>	
3	<p>What is CoreData?</p> <ol style="list-style-type: none"> <li>1. Difference between Sqlite and CoreData. Which is better?</li> <li>2. What are the benefits of using CoreData?</li> <li>3. Which is faster Sqlite or CoreData?</li> <li>4. Why CoreData is faster than Sqlite?</li> <li>5. Can Sqlite and CoreData exists in same project.</li> <li>6. What is persistentStoreCoordinator in CoreData?</li> <li>7. What is NSPredicate?</li> <li>8. What is NSManagedObject.? How NSManagedObject is different from NSObject?</li> <li>9. What is NSManagedObjectContext in CoreData?</li> <li>10. Have you worked with CoreData disk storage?</li> <li>11. What is transient property in CoreData?</li> <li>12. Which are the various data types supported by CoreData Entity?</li> <li>12. Which datatype is for storing image?</li> <li>14. Is CoreData is threadSafe?</li> <li>15. How to work with relation in CoreData?. Why we need relations.</li> <li>16. What is Faulting in CoreData?</li> <li>17. Can CoreData be multi-threaded?</li> <li>18. How to work in multi-threaded environment in CoreData</li> <li>19. Can we add and delete same time in CoreData?</li> <li>20. How to fetch 1000 rows from CoreData for displaying in tableview</li> <li>21. How to delete 1000 rows from CoreData Entity?</li> <li>22. Can I store Array in CoreData?</li> <li>23. How to delete a row in core data after 24 hours?</li> <li>24. If you create model for each entity?</li> <li>25. If CoreData is Synchronous ?</li> <li>26. Can two thread simultaneously work on CoreData?</li> <li>27. In CoreData , How to fetch 5 highly paid workers. Even if there is only 2 records in the entity?</li> <li>28. How to track down local data. And how to free space if it is getting full by a lot of chat messages?</li> <li>29. How to migrate CoreData?</li> <li>30. Why do we migrate database when there are some changes. Can we work without migration. What will happen if forget to migrate. Which one will get crash existing user or new users? And what is the reason for the crash</li> <li>31. How you decide when to use CoreData and when not to use CoreData?</li> <li>32. How to merge private queue into mainQueue in CoreData?</li> <li>33. If Json data to be stored in the CoreData entity. What is the best way if there are calculations?</li> <li>34. How you decide when to use CoreData and when not to use CoreData?</li> <li>35. How to create Primary Key in CoreData?</li> <li>36. How to merge existing database to CoreData?</li> <li>37. What will be the parent class of Entity named class Student?</li> <li>38. What is Composite property in CoreData?</li> </ol>	
4	<p>What is Plist?</p> <ol style="list-style-type: none"> <li>1. What types of variable can be stored in plist?</li> <li>2. Different type of GCD?</li> </ol>	

What is UserDefaults?

- 5
  - 1. How to delete all value of UserDefaults?
  - 2. How to secure UserDefaults data that it can not be read by any?

What is keyChain?

- 1. How to store credentials in keyChain?
  - 2. How to fetch data from keyChain?
  - 3. How to delete value form keychain
- 6
  - 4. What is the benefit of keyChain?
  - 5. If I delete the app. Will the data in keychain will also be deleted or not
  - 6. Other secure way to store credentials except keychain?

7 Difference between Graph and relational Database?

8 How to save secure private data?

9 Where to store secure data like Username and password?

10 What are joins? What are self joins?