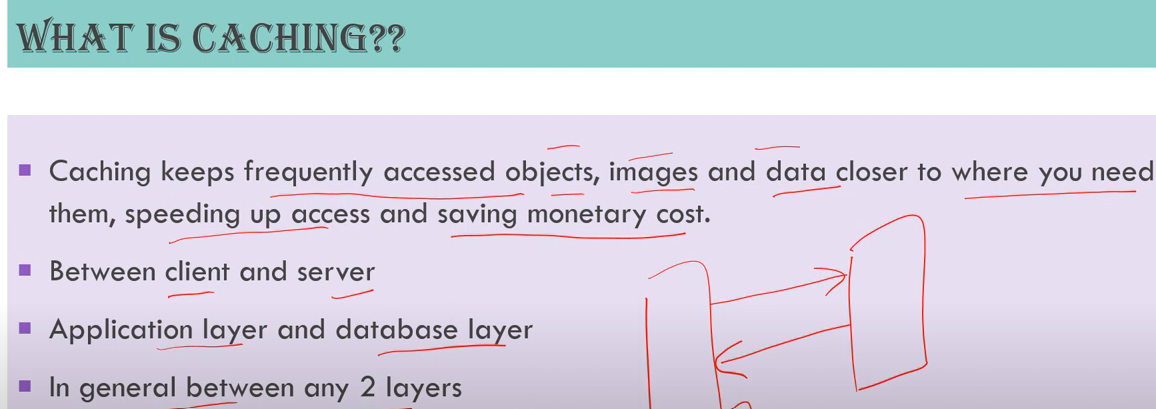
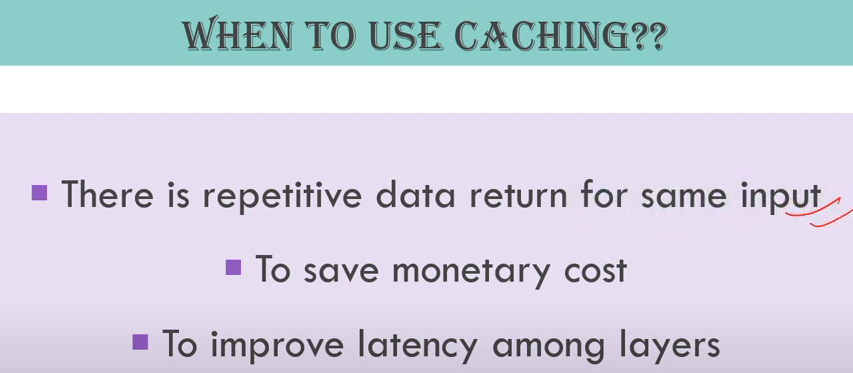
**What is Cache:**

Caching is a part of temporary memory (RAM). It lies between the application and persistence database. It stores the recently used data that reduces the number of database hits as much as possible. In other words, caching is to store data for future reference.

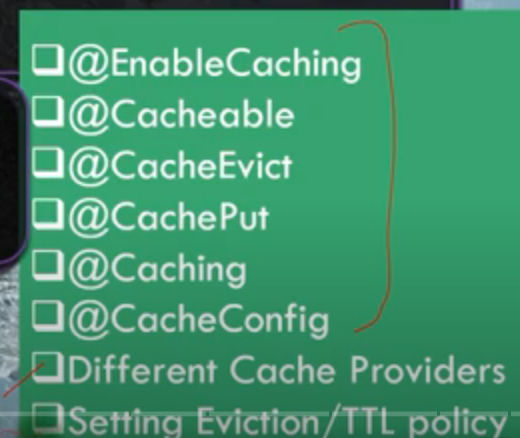
****

**When to use Cache:**

* You can use caching to significantly reduce latency and improve IOPS(Input/Output operation per second) for many read heavy application workloads such as Q&A portals, gaming, media sharing and social networking.

****

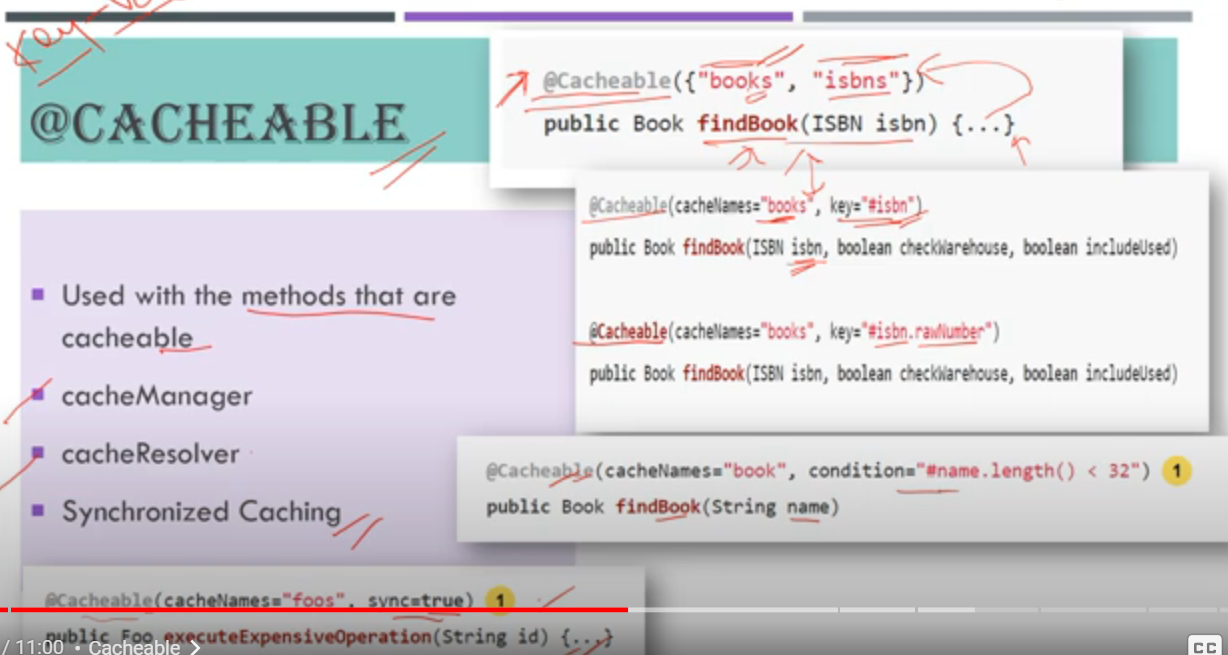
**Spring Boot caching:**

****

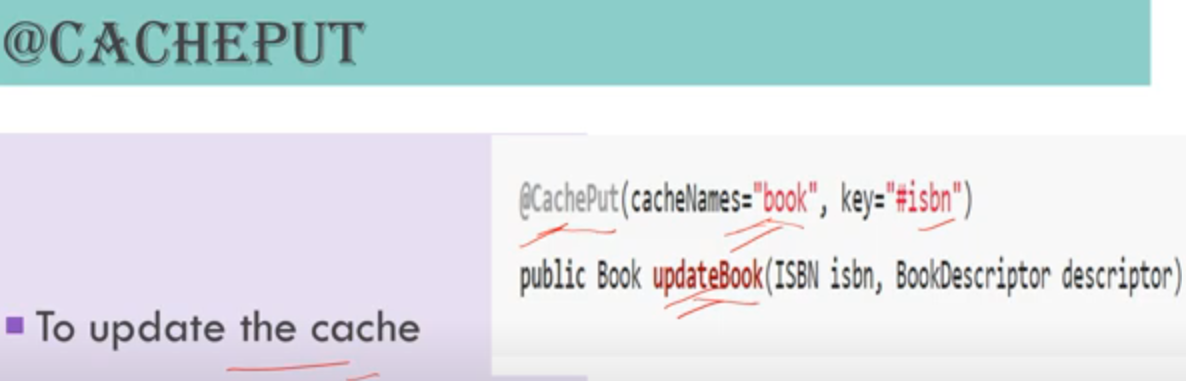
* **@Enablecaching: Enable caching in spring.**

****

* **@cacheable –** 
  + this should always be used on top of method name.
  + 1st argument will be cachename and 2nd will be key.
  + We can provide fieldname as key in cache using key=”#object.fieldname”
  + We can also given condition for key. E.g. condition=”#name.lenght()<32”
  + There might be chances of race conditions, to avoid it we can make cache as synchronized using sync keyword. E.g. sync=true

****

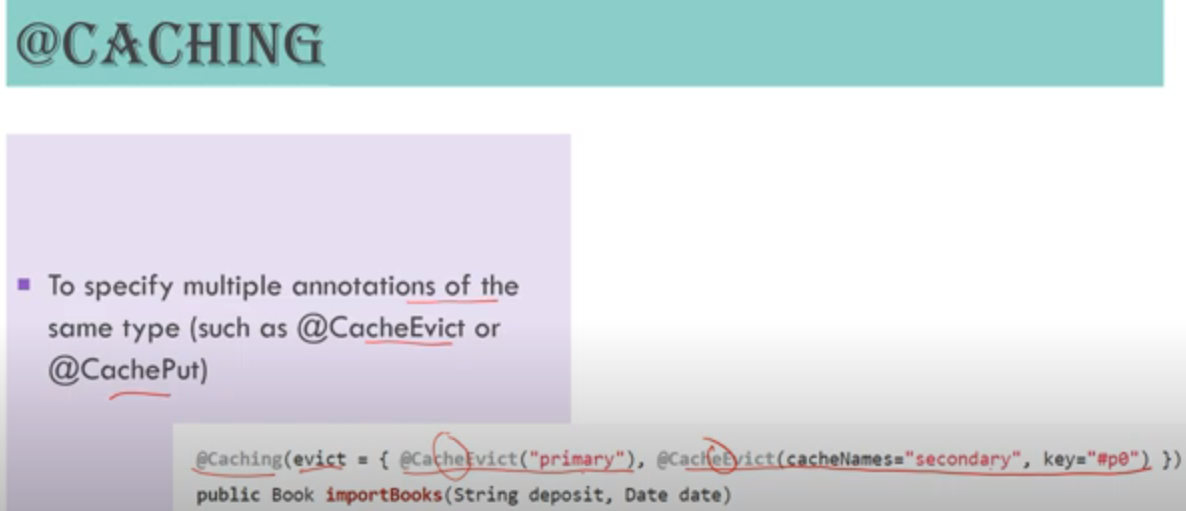
* **@cahceput:** it is always used when we are making any update operation to resource/database so that it will update cache value accordingly.

****

* **@cacheevict:** it is used when we want to delete either specific data or all data from cache(allEntries=true)

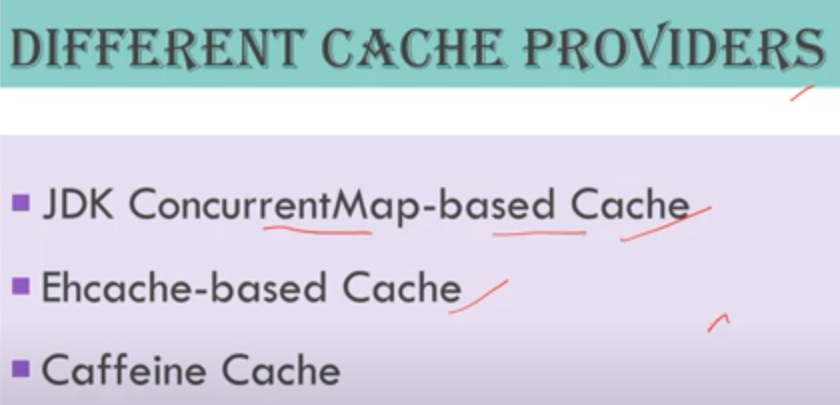
****

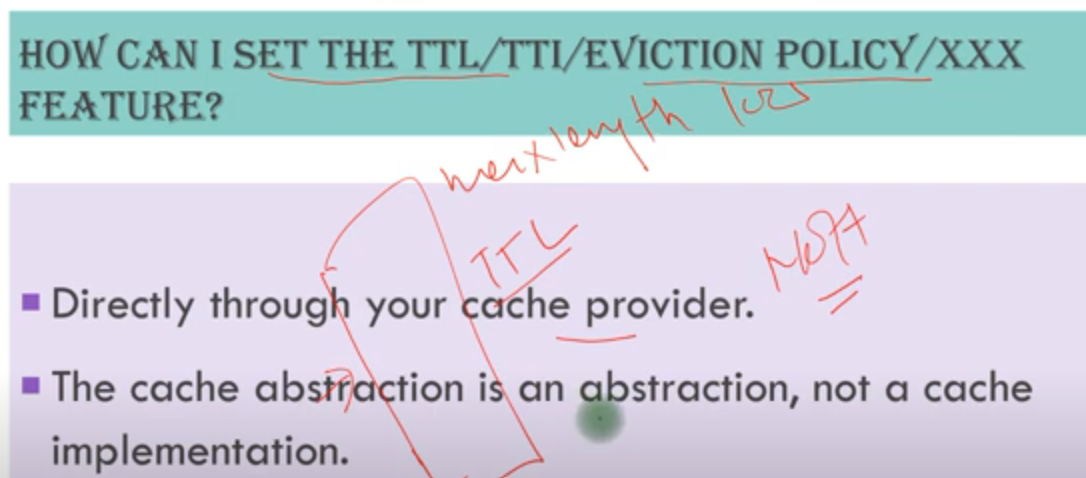
* **@caching:** Regroups multiple cache operations to be applied on a method

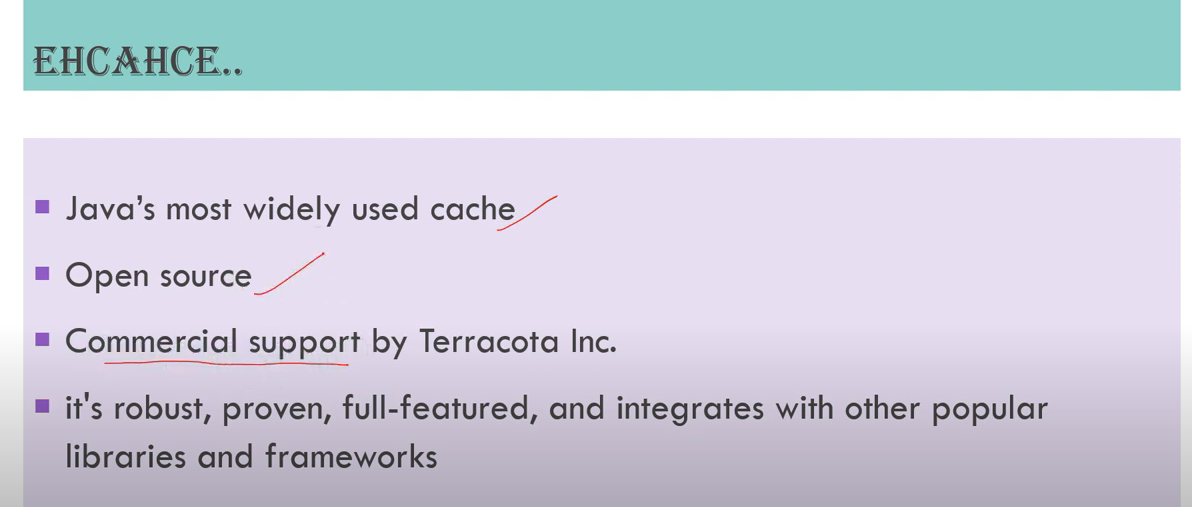
****

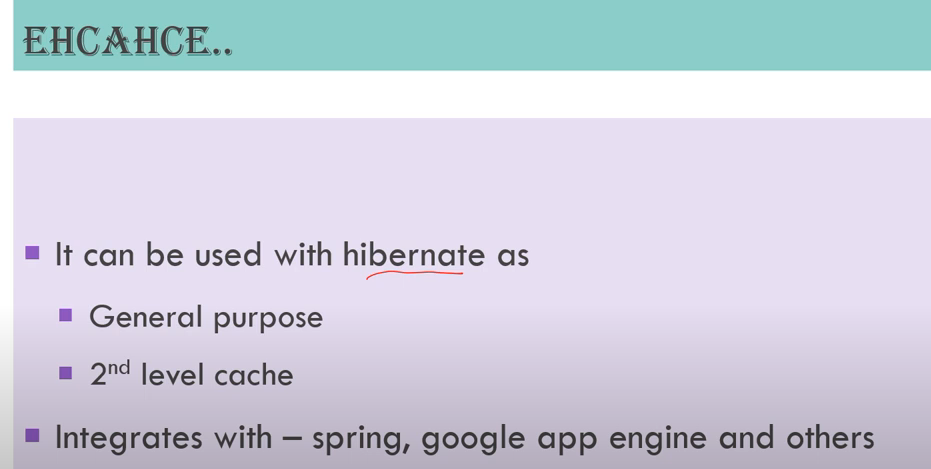
* **@cacheconfig:** When you want to apply same cache to multiple methods of a class then you can use @cacheconfig at class level.
* ****

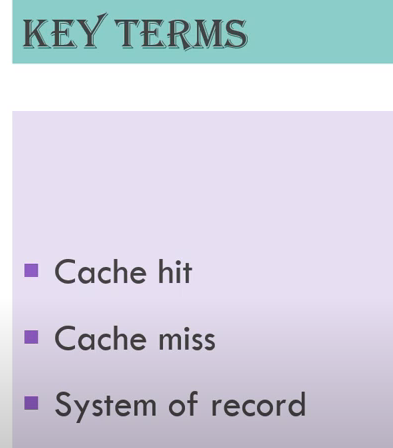
**Different Cache providers:**

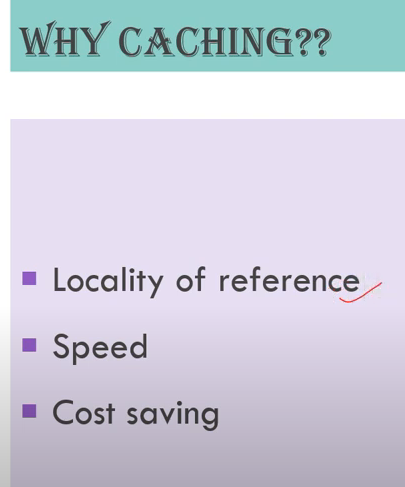
****

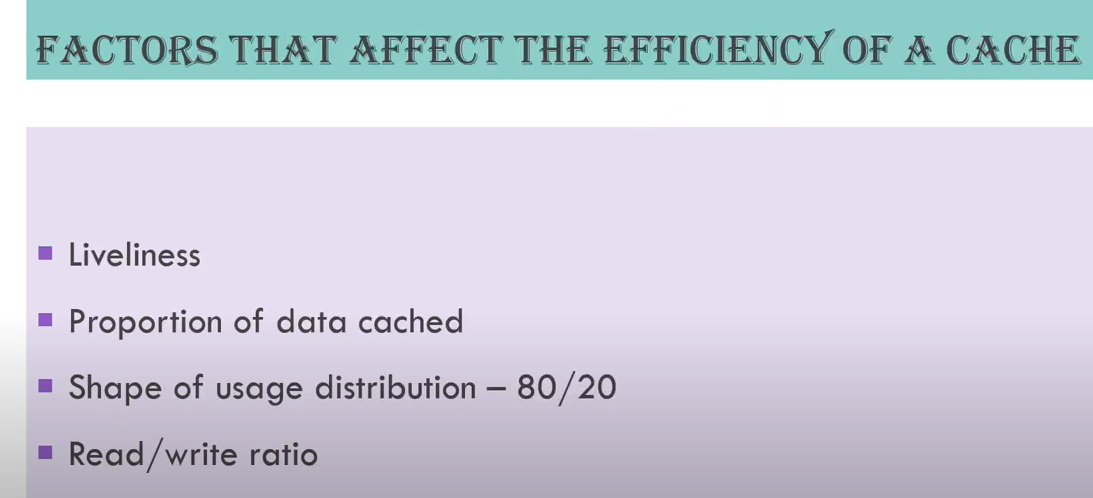
****

****

****

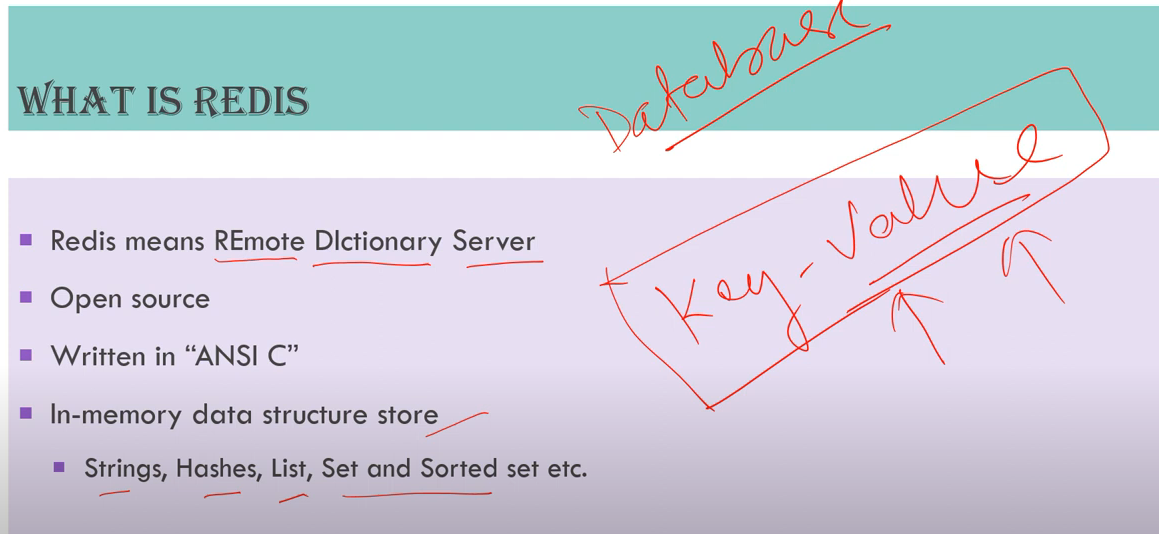
****

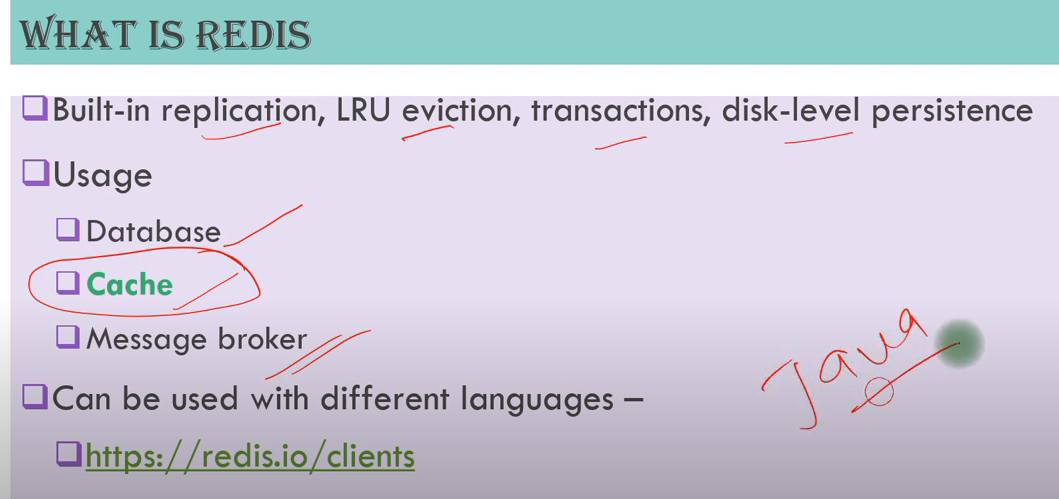
****

****

****

****

****

****