Certainly! In a nutshell, an **LRU (Least Recently Used) cache** is a data structure that efficiently manages limited memory by discarding the least recently accessed items when new data needs to be stored. It’s commonly used for caching purposes in computer science and software engineering.

Here are **five free reference links** where you can learn more about LRU caches and their implementations:

1. [**GeeksforGeeks**: This comprehensive tutorial covers LRU cache concepts, working, and various implementations using different data structures like queues, linked lists, and hash maps1](https://www.geeksforgeeks.org/lru-cache-implementation/).
2. [**Interview Cake**: Dive into the details of LRU caches with this article, which pairs a doubly linked list with a hash map to create an efficient caching system](https://www.geeksforgeeks.org/lru-cache-implementation/)[2](https://www.interviewcake.com/concept/java/lru-cache).
3. [**Educative**: Learn about LRU cache as a common caching strategy and explore how it discards the least recently used items first](https://www.geeksforgeeks.org/lru-cache-implementation/)[3](https://www.educative.io/implement-least-recently-used-cache).
4. [**The Crazy Programmer**: Discover how to implement an LRU cache using a deque (doubly ended queue) and a hash map](https://www.geeksforgeeks.org/lru-cache-implementation/)[4](https://www.thecrazyprogrammer.com/2021/02/lru-cache.html).
5. [**Baeldung**: Explore LRU cache concepts and find a Java implementation example](https://www.geeksforgeeks.org/lru-cache-implementation/)[5](https://www.baeldung.com/java-lru-cache).

Feel free to explore these resources to deepen your understanding of LRU caches! 📚👩‍💻