

1. Implement LRU cache on the basis of LinkedHashMap

a. Create the next interface

```
public interface LruCache {

    /**
     * @return the value of the key if the key exists,
     * otherwise return -1
     */
    int get(int key);

    /**
     * Update the value of the key if the key exists.
     * Otherwise, add the key-value pair to the cache.
     * If the number of keys exceeds the capacity from this operation,
     * evict the least recently used key.
     */
    void put(int key, int value);

    /**
     * Set the max capacity of the cache
     */
    void setCapacity(int capacity);

}
```

b. Implement LruCache interface in class DefaultLruCache

c. Commit with changes to check only files that were changed (solution) -

[https://github.com/AndriiPiatakha/learnit\\_java\\_core/commit/826cf08b3a121763b363b2777a0e32ca16d43e54](https://github.com/AndriiPiatakha/learnit_java_core/commit/826cf08b3a121763b363b2777a0e32ca16d43e54)