

```
1 /*
2  * DAO sınıfını oluşturduk.
3  * Kullanıcı ayrıntılarını depolamak için bir usersMap kullandık (sadece DB etkileşimini önlemek için)
4  * diğer tüm yöntemler, kullanıcı ayrıntılarını okumak, eklemek, güncellemek ve silmek için bu userMap'i kullanır.
5  */
6
7 package com.kb.dao;
8
9 import java.util.ArrayList;
10
11 //Just to avoid DB calls in this example, Assume below data is interacting with DB
12 public class UserDao {
13     static HashMap<String, User> usersMap = new HashMap<String, User>();
14
15     public UserDao() {
16     }
17
18     public List<User> getAllUsers() {
19
20         List<User> userList = new ArrayList<User>(usersMap.values());
21         return userList;
22     }
23
24     public User getUserForId(String id) {
25         User user = usersMap.get(id);
26         return user;
27     }
28
29     public User createUser(User user) {
30         usersMap.put(user.getId(), user);
31         return usersMap.get(user.getId());
32     }
33
34     public User updateUser(User user) {
35         User existingUser = usersMap.get(user.getId());
36         if (existingUser != null) {
37             existingUser.setName(user.getName());
38             existingUser.setEmail(user.getEmail());
39             existingUser.setCountry(user.getCountry());
40         }
41     }
42 }
```