

Дан код.

```
package ua.nure.your_last_name.Practice8;

import java.util.List;

import ua.nure.your_last_name.Practice8.db.DBManager;
import ua.nure.your_last_name.Practice8.db.entity.Group;
import ua.nure.your_last_name.Practice8.db.entity.User;

public class Demo {

    private static <T> void printList(List<T> list) {
        for (T element : list) {
            System.out.println(element);
        }
    }

    public static void main(String[] args) {
        // users ==> [ivanov]
        // groups ==> [teamA]

        DBManager dbManager = DBManager.getInstance();

        // Part 1
        dbManager.insertUser(User.createUser("petrov"));
        dbManager.insertUser(User.createUser("obama"));
        printList(dbManager.findAllUsers());
        // users ==> [ivanov, petrov, obama]

        System.out.println("=====");

        // Part 2
        dbManager.insertGroup(Group.createGroup("teamB"));
        dbManager.insertGroup(Group.createGroup("teamC"));
        printList(dbManager.findAllGroups());
        // groups ==> [teamA, teamB, teamC]

        System.out.println("=====");

        // Part 3
        User userPetrov = dbManager.getUser("petrov");
        User userIvanov = dbManager.getUser("ivanov");
        User userObama = dbManager.getUser("obama");

        Group teamA = dbManager.getGroup("teamA");
        Group teamB = dbManager.getGroup("teamB");
        Group teamC = dbManager.getGroup("teamC");

        // method setGroupsForUser must implement transaction!
        dbManager.setGroupsForUser(userIvanov, teamA);
        dbManager.setGroupsForUser(userPetrov, teamA, teamB);
        dbManager.setGroupsForUser(userObama, teamA, teamB, teamC);

        for (User user : dbManager.findAllUsers()) {
            printList(dbManager.getUserGroups(user));
            System.out.println("~~~~~");
        }
        // teamA
        // teamA teamB
        // teamA teamB teamC

        System.out.println("=====");
    }
}
```

```
}
```

Создать и реализовать соответствующие типы таким образом, чтобы при запуске класса Демо отработывала соответствующая функциональность.

Метод DBManager#setGroupsForUser должен реализовывать транзакцию. Грамотно реализовать логику commit/rollback транзакции.

Метод DBManager#getUserGroups возвращает объект java.util.List<Group>.