### daoImpl各类对数据库的操作说明

ApplicationDaoImpl

类说明：实现ApplicationDao接口，对数据库操作进行事务管理和错误处理

作用：实现对application（应用分日信息表）表的增删改查

方法说明：

**1.public List<Application> findAll()**

功能：查询所有application

参数：无

返回值：Application类的list集合

**2. public Integer saveApp(Application application)**

功能：插入一条application

参数：Application类

返回值：**Integer** 成功为1，失败为-1

1. **public Application findAppById(Integer G\_ID)**

功能：根据G\_ID查询一条application

参数：G\_ID（游戏ID）

返回值：Application类 失败则为null

**4.public Integer updateAppByApp(Application application)**

功能：对一条application更新

参数：Application类

返回值：**Integer** 成功为1，失败为-1

**5.public Integer deleteAppById(Integer G\_ID)**

功能：根据G\_ID删除一条application

参数：G\_ID(游戏id)

返回值：**Integer** 成功为1，失败为-1

### GameDaoImpl

类说明：实现GameDao接口，对数据库操作进行事务管理和错误处理

作用：实现对Taptap游戏信息表(game)的增删改查

方法说明：

**1.public List<Game> findAll()**

功能：查询所有game

参数：无

返回值：Game类的list集合 失败则为null

**2.public Integer saveGame(Game game)**

功能：插入一条game

参数：Game类

返回值：**Integer** 成功为1，失败为-1

**3.public Game findGameById(Integer G\_ID)**

功能：根据G\_ID查询一条**Game**

参数：G\_ID（游戏ID）

返回值：Game类 失败则为null

1. **public Integer deleteGameById(Integer G\_ID)**

功能：根据G\_ID删除一条**Game**

参数：G\_ID（游戏ID）

返回值：**Integer** 成功为1，失败为-1

**5.public Integer updateGame(Game game)**

功能：更新一条**Game**

参数：Game类

返回值：**Integer** 成功为1，失败为-1

### GameTouserDaoImpl

类说明：实现GameTouserDao接口，对数据库操作进行事务管理和错误处理

作用：实现对Taptap游戏评论对应用户信息表(gameToUser)的增删改查

方法说明：

1. **public List<GameTouser> findAll()**

功能：查询所有gameTouser

参数：无

返回值：GameTouser类的list集合 失败则为null

1. **public Integer saveGameTouser(GameTouser gameTouser)**

功能：插入一条gameTouser

参数：**GameTouser类**

返回值：**Integer** 成功为1，失败为-1

1. **public GameTouser findGameTouserById(Integer D\_ID)**

功能：根据D\_ID（评论id）查询一条gameTouser

参数：D\_ID（评论id）

返回值：GameTosuer类 失败则为null

**4.public Integer deleteGameTouser(Integer D\_ID)**

功能：根据D\_ID（评论id）删除一条gameTouser

参数：D\_ID（评论id）

返回值：**Integer** 成功为1，失败为-1

**5.public Integer updateGameTouser(GameTouser gameTouser)**

功能：更新一条gameTouser

参数：**GameTouser类**

返回值：**Integer** 成功为1，失败为-1

**6.public List<GameEverydayData> findCountEverydayAll()**

功能：查询每天的评论数量和平均星级

参数：**无**

返回值：**GameEverydayData类集合** 失败则为null

**（附GameEverydayData说明：三个属性：time(日期),num（当天评论总数）,average（平均评论星级） 位置：pojo包）**

1. **public List<GameStarLevel> findCountByStar(Float D\_START)**

功能：查询每天评论的星级条数

参数：**Float型 D\_START评论星级**

返回值：**GameStarLevel类集合** 失败则为null

**（附GameStarLevel说明：三个属性：time(日期),num（评论星级对应条数）,star（评论星级） 位置pojo包）**

### GameBasicDaoImpl

类说明：实现GameBasicDao接口，对数据库操作进行事务管理和错误处理

作用：实现对应用基本信息表（gameBasic）的增删改查

方法说明：

1.public List<GameBasic> findAll();

功能：查询所有GameBasic

参数：无

返回值：GameBasic类的list集合 失败则为null  
2.public GameBasic findGBById(Integer G\_ID);

功能：查询一条GameBasic

参数：G\_ID游戏ID

返回值：GameBasic类 失败则为null

3.public Integer deleteGameBasic(Integer G\_ID);

功能：删除一条GameBasic

参数：G\_ID游戏ID

返回值：Integer成功为1，失败为-1  
4.public Integer updateGameBasic(GameBasic gameBasic);

功能：更新一条GameBasic

参数：GameBasic类

返回值： Integer 成功为1，失败为-1  
5.public Integer saveGameBasic(GameBasic gameBasic);

功能：插入一条gameBasic

参数：GameBasic**r类**

返回值： Integer成功为1，失败为-1

### RecentAppraiseDaoImpl

类说明：实现RecentAppraiseDao接口，对数据库操作进行事务管理和错误处理

作用：实现对应用基本信息表（recentAppraise）的增删改查

方法说明：

1.public List<RecentAppraise> findAll();  
功能：查询所有RecentAppraise

参数：无

返回值：RecentAppraise类的list集合 失败则为null  
2.public Integer saveRecentAppraise(RecentAppraise recentAppraise);  
功能：插入一条RecentAppraise

参数：RecentAppraise类

返回值：Integer 成功为1，失败为-1

3.public RecentAppraise findRecentAppraiseById(Integer G\_ID);  
功能：根据G\_ID查找一条RecentAppraise

参数：G\_ID(游戏id)

返回值：RecentAppraise类 失败则为null  
4.public Integer deleteRecentAppraiseById(Integer G\_ID);

功能：根据G\_ID删除一条RecentAppraise

参数：G\_ID(游戏id)

返回值：Integer成功为1，失败为-1

5.public Integer updateRecntAppraise(RecentAppraise recentAppraise);

功能：更新RecentAppraise

参数：RecentAppraise类

返回值：Integer成功为1，失败为-1

RecentPlayDaoImpl

类说明：实现RecentPlayDao接口，对数据库操作进行事务管理和错误处理

作用：实现对Taptap用户常玩游戏信息表（RecentPlay）的增删改查

方法说明：

1. **public List<RecentPlay> findAll();**

功能：查询所有RecentPlay

参数：无

返回值：RecentPlay类的list集合 失败则为null  
**2.public Integer saveRecentPlay(RecentPlay recentPlay);**

功能：插入一条RecentPlay

参数：RecentPlay类

返回值：Integer成功为1，失败为-1 **3.public RecentPlay findRecentPlayById(Integer G\_ID);**

功能：根据G\_ID查询RecentPlay

参数：G\_ID(游戏G\_ID)

返回值：RecentPlay类 失败则为null **4.public Integer updateRecentPlay(RecentPlay recentPlay);**

功能：更新RecentPlay

参数：RecentPlay类

返回值：Integer成功为1，失败为-1 **5.public Integer deleteRecentPlayById(Integer G\_ID);**

功能：删除RecentPlay

参数：G\_ID（游戏ID）

返回值：Integer成功为1，失败为-1

### TapUserDaoImpl

类说明：实现TapUserDao接口，对数据库操作进行事务管理和错误处理

作用：实现对Taptap用户基本信息表（tapUser）的增删改查

方法说明：

1.public List<TapUser> findAll();  
功能：查询所有TapUser

参数：无

返回值：TapUser类的list集合 失败则为null

2.TapUser findTapUserById(Integer U\_ID);  
功能：根据U\_ID查询TapUser

参数：U\_ID（Taptap用户id）

返回值：TapUser类 失败则为null  
3.Integer deleteTapUser(Integer U\_ID);

功能：根据U\_ID删除TapUser

参数：U\_ID（Taptap用户id）

返回值：Integer成功为1，失败为-1  
4.Integer updateTapUser(TapUser tapUser);

功能：更新TapUser

参数：U\_ID（Taptap用户id）

返回值：Integer成功为1，失败为-1

5.Integer SaveTapUser(TapUser tapUser);

功能：插入一条TapUser

参数：TapUser类

返回值：Integer成功为1，失败为-1

### UuserDaoImpl

类说明：实现UuserDao接口，对数据库操作进行事务管理和错误处理

作用：实现对系统用户信息表(UUser)的增删改查

方法说明：

1.List<Uuser> findAll();  
功能：查询所有TapUser

参数：无

返回值：Uuser类的list集合 失败则为null

1. Uuser findUserById(Integer UU\_ID);

功能：根据id查询TapUser

参数： UU\_ID（用户ID）

返回值：Uuser类 失败则为null  
3.Integer deleteUser(Integer UU\_ID);

功能：根据id删除TapUser

参数： UU\_ID（用户ID）

返回值：Integer成功为1，失败为-1

4.Integer updateUser(Uuser uuser);

功能：更新TapUser

参数： Uuser类

返回值：Integer成功为1，失败为-1

5.Integer saveUser(Uuser uuser);

功能：插入一条TapUser

参数： Uuser类

返回值：Integer成功为1，失败为-1

Utils包说明

#### UnicodeInputStream类

类说明：去除json解析文件头带有的utf-8的demo符号

作用：辅助解析

方法说明：

#### JsonUtils

类说明：对game的json数据的解析和批量插入

作用：辅助解析

方法说明：

**1.public static void getGameMap(String txtStr)**

功能：对game和gametosuer表进行数据的插入

参数：经过reader()从json文件读取出来的字符串

返回值：**无**

FileUtils

类说明：对json文件进行读取

作用：文件读取

方法说明

1.public static String reader(String filePath)

功能：读取txt文件内容(Json格式数据)：本地文件.json

参数：json文件的路径

返回值：**解析后的String**

#### MybatilsUtils

类说明：初始化 SqlSessionFactory

作用：初始化sqlSession

方法说明:

**1.public static SqlSession getSession()**

功能:获取SqlSession对象

参数：无

返回值：SqlSession类

2.public static void destroy()

功能:提交事务，关闭in通道

参数：无

返回值：无