

Json转换利器Gson之实例六-注册TypeAdapter及处理Enum类型

标签: Enum enum Gson json JSON Json TypeAdapter 实例

2012-06-22 22:50

17708人阅读

评论(14)

收藏

举报

分类:

Json (5)

版权声明：
本

文为博主原创文章，未经博主允许不得转载。

枚举类型给我们的程序带来了好处,如何用Gson来实现与Json的互转呢?请看本文.

本文重点掌握如何自己写一个TypeAdapter及注册TypeAdapter和处理Enum类型.

实体类:

[java]

```
01. public enum PackageState {
02.     PLAY, UPDATE, UPDATING, DOWNLOAD, DOWNLOADING,
03. }
```

[java]

```
01. public class PackageItem {
02.     private String name;
03.     private PackageState state;
04.     private String size;
05.
06.     public String getName() {
07.         return name;
08.     }
09.
10.     public void setName(String name) {
11.         this.name = name;
12.     }
13.
14.     public PackageState getState() {
15.         return state;
16.     }
17.
18.     public void setState(PackageState state) {
19.         this.state = state;
20.     }
21.
22.     public String getSize() {
23.         return size;
24.     }
25.
26.     public void setSize(String size) {
27.         this.size = size;
28.     }
29. }
```

```
30.     @Override
31.     public String toString() {
32.         return "PackageItem [name=" + name + ", size=" + size + ", state="
33.             + state + "]";
34.     }
35. }
```

自己写一个转换器实现JsonSerializer<T>接口和jsonDeserializer<T>接口:

[java]

```
01. import java.lang.reflect.Type;
02.
03. import com.google.gson.JsonDeserializationContext;
04. import com.google.gson.JsonDeserializer;
05. import com.google.gson.JsonElement;
06. import com.google.gson.JsonParseException;
07. import com.google.gson.JsonPrimitive;
08. import com.google.gson.JsonSerializationContext;
09. import com.google.gson.JsonSerializer;
10.
11. public class EnumSerializer implements JsonSerializer<PackageState>,
12.     JsonDeserializer<PackageState> {
13.
14.     // 对象转为Json时调用,实现JsonSerializer<PackageState>接口
15.     @Override
16.     public JsonElement serialize(PackageState state, Type arg1,
17.         JsonSerializationContext arg2) {
18.         return new JsonPrimitive(state.ordinal());
19.     }
20.
21.     // json转为对象时调用,实现JsonDeserializer<PackageState>接口
22.     @Override
23.     public PackageState deserialize(JsonElement json, Type typeOfT,
24.         JsonDeserializationContext context) throws JsonParseException {
25.         if (json.getAsInt() < PackageState.values().length)
26.             return PackageState.values()[json.getAsInt()];
27.         return null;
28.     }
29.
30. }
```

测试类:

[java]

```
01. import com.google.gson.Gson;
02. import com.google.gson.GsonBuilder;
03.
04. public class GsonTest6 {
05.
06.     public static void main(String[] args) {
07.         GsonBuilder gsonBuilder = new GsonBuilder();
08.         gsonBuilder.registerTypeAdapter(PackageState.class,
09.             new EnumSerializer());
10.     }
```

```
10.         Gson gson = gsonBuilder.create();
11.         PackageItem item = new PackageItem();
12.         item.setName("item_name");
13.         item.setSize("500M");
14.         item.setState(PackageState.UPDATING);// 这个 state是枚举值
15.
16.         String s = gson.toJson(item);
17.         System.out.println(s);
18.
19.         System.out.println("-----");
20.
21.         PackageItem retItem = gson.fromJson(s, PackageItem.class);
22.         System.out.println(retItem);
23.     }
24. }
```

输出结果(结果中已经将state的对应枚举类型转为了int类型):

```
[plain]
01.     {"name":"item_name","state":2,"size":"500M"}
02.     -----
03.     PackageItem [name=item_name, size=500M, state=UPDATING]
```

Json转换利器Gson之实例一-简单对象转化和带泛型的List转化

(http://blog.csdn.net/lk_blog/article/details/7685169)

Json转换利器Gson之实例二-Gson注解和GsonBuilder

(http://blog.csdn.net/lk_blog/article/details/7685190)

Json转换利器Gson之实例三-Map处理(上)

(http://blog.csdn.net/lk_blog/article/details/7685210)

Json转换利器Gson之实例四-Map处理(下)

(http://blog.csdn.net/lk_blog/article/details/7685224)

Json转换利器Gson之实例五-实际开发中的特殊需求处理

(http://blog.csdn.net/lk_blog/article/details/7685237)

Json转换利器Gson之实例六-注册TypeAdapter及处理Enum类型

(http://blog.csdn.net/lk_blog/article/details/7685347)

实例代码下载: http://download.csdn.net/detail/lk_blog/4387822