

```
}  
}
```

这里提供的是一个享元工厂，通过 `Map` 结构存放已经从库表或接口中查询到的数据并存放内存中，方便下次直接获取。这种结构在编程开发中是比较常见的，有时也为了保证分布式系统部署能获取到信息，会把数据存放到 Redis 中。

### 14.5.5 模拟 Redis 服务

```
public class RedisUtils {  
    private ScheduledExecutorService scheduledExecutorService  
        = Executors.newScheduledThreadPool(1);  
    private AtomicInteger stock = new AtomicInteger(0);  
    public RedisUtils() {  
        scheduledExecutorService.scheduleAtFixedRate(() -> {  
            // 模拟库存消耗  
            stock.addAndGet(1);  
        }, 0, 100000, TimeUnit.MICROSECONDS);  
    }  
    public int getStockUsed() {  
        return stock.get();  
    }  
}
```

这里除了模拟 Redis 的操作工具类，还提供了一个定时任务，用于模拟库存消耗，这样可以在测试时观察商品库存的变化。

### 14.5.6 活动控制类

```
public class ActivityController {  
    private RedisUtils redisUtils = new RedisUtils();  
    public Activity queryActivityInfo(Long id) {  
        Activity activity = ActivityFactory.getActivity(id);  
        // 模拟从 Redis 中获取商品库存变化信息  
        Stock stock = new Stock(1000, redisUtils.getStockUsed());  
        activity.setStock(stock);  
        return activity;  
    }  
}
```

在活动控制类中使用了享元工厂获取活动信息，查询后将商品库存信息再补充到商品