트랜잭션 - TransactionTemplate 활용

프로젝트명 : B08bTransactionTemplate

준비사항

- 기존 B08aTransactionManager를 복사해서 현재 프로젝트명으로 변경한다.
- 컨텍스트 루트 및 settings.gradle 을 변경한다.
- gradle을 리플레쉬 한다.

컨트롤러 : com.edu.springboot.MainController.java

```java
1  package com.edu.springboot;
2
3  import javax.servlet.http.HttpServletRequest;
17
18 @Controller
19 public class MainController {
20
21     @Autowired
22     ITicketService dao;
23
24     @Autowired
25     TransactionTemplate transactionTemplate;
26
27     @RequestMapping("/")
28     public String home() {
29         return "home";
30     }
31
```

```java
32     //티켓구매
33     @RequestMapping(value="/buyTicket.do", method=RequestMethod.GET)
34     public String buy1() {
35         return "buy";
36     }
37     @RequestMapping(value="/buyTicket.do", method=RequestMethod.POST)
38     public String buy2(TicketDTO ticketDTO, PayDTO payDTO, HttpServletRequest req,
39             Model model) {
40         String viewPath = "success";
41         try {
42             transactionTemplate.execute(new TransactionCallbackWithoutResult() {
43                 @Override
44                 protected void doInTransactionWithoutResult(
45                         TransactionStatus status) {
46                     //DB처리1
47                     payDTO.setAmount(ticketDTO.getT_count() * 10000);
48                     int result1 = dao.payInsert(payDTO);
49                     if(result1==1)
50                         System.out.println("transaction_pay 입력성공");
```

```java
51
52                    //비즈니스로직 처리(의도적인 에러발생)
53                    String errFlag = req.getParameter("err_flag");
54                    if(errFlag!=null) {
55                        int money = Integer.parseInt("100원");
56                    }
57
58                    //DB처리2
59                    int result2 = dao.ticketInsert(ticketDTO);
60                    if(result2==1)
61                        System.out.println("transaction_ticket 입력성공");
62
63                    model.addAttribute("ticketDTO", ticketDTO);
64                    model.addAttribute("payDTO", payDTO);
65                }
66            });
67        }
68        catch (Exception e) {
69            e.printStackTrace();
70            viewPath = "error";
71        }
72
73        return viewPath;
74    }
75 }
```