그냥 controller는 처리 결과를 view(jsp, html)로 넘기는 것에 특화되어 있음.

물론 view가 아닌 응답을 할 수 있다. Response body나 responseEntity를 이용하는 등. JSON이나 XML 형식을 주고 받는 rest기반 통신이 가능하긴 하다.

하지만 좀 더 rest 특화된 controller를 사용하는 것이 좋다.

🡺 rest controller. // 해당 클래스의 예외 처리용인 rest controller advice도 있다.

**기억해야할 키워드 :** @RestController, rest controller advice, @RequestBody, @Transactional,  
@EnableTransactionManagement, ResponseEntity<T>, @RestControllerAdvice, @ExceptionHandler

**구성 및 준비 :**

Vo, DTO 준비

Vo를 채울 DB연산이 구현된 매퍼 준비

매퍼를 이용하는 서비스 준비 2개이상의 매퍼를 이용한 연산이 한번에 되어야 할 때 @Transactional어노테이션 이용. //물론 RootConfig에 @EnableTransactionManagement를 적용해 트랜잭션 AOP기능을 켜야한다.

**활용 :**

Src>main>java>org.scoula>???Controller.java  
@RestController //모든 메서드에 @ResponseBody 자동 추가. 응답헤더의 content-type을 application/json으로  
변경함. 메서드가 객체를 리턴하면 자동으로 json으로 변환. jackson라이브러리가 담당.  
@RequsetMapping(“/api/ex”)  
@RequiredArgsConstructor @Log4j2  
public class ???Controller{  
 @GetMapping( “/ex1” )  
 public BoardDTO get(Long no){  
 return service.get(no);  
 }  
  
 @PostMapping(“/ex2”)  
 public BoardDTO create(@RequestBody boardDTO newBoard){  
 //요청의 바디가 application/json인코딩인 경우, 해당DTO로 받고 싶다면 @RequestBody 앞에다가.  
 //@RequestBody말고 다른거나 없다면 요청의 body가 아니라 쿼리스트링이나   
 //폼데이터(멀티파트 포함(에서 값을 추출함.  
 return service.create(newBoard);  
 }  
  
 // 예를 들어 응답코드를 수동으로 다른걸 내보내고 싶다. 커스터마이징 할 수 있는 방법  
 //ResponseEntity<json이될 객체type>활용. 상태코드와 헤더 설정하기 좋다. Body도 설정 가능.  
 @GetMapping(“/ex3”)  
 public ResponseEntity<BoardDTO> get2(Long no){  
 if(…){ return ResponseEntity.ok( ).build( ); } //정상 응답 보내고 싶을때  
 else if(…){ return ResponseEntity.ok(???DTO); }  
 else if(…){ return ResponseEntity.ok( ).headers(headers).body(???DTO); }  
 else if(…){ return ResponseEntity.status(상태코드).build( ); } //다른 상태 응답 보내고 싶을때  
 else if(…){ return ResponseEntity.status(상태코드).body(???DTO); }  
 else if(…){ return ResponseEntity.status(상태코드).headers(headers).body(???DTO); }  
 }  
}

에러처리 : RestController에서의 처리과정에서 발생한 예외는 따로 RestControllerAdvice가 핸들링 해야한다.

@RestControllerAdvice //AOP 기능을 이용. @ControllerAdvice도 같다  
public class ApiExceptionAdvice{  
 @ExceptionHandler( { 처리할예외클래스1.class, 처리할예외클래스2.class, … } )  
 protected ResponeEntity<??? > handleIllegalArgumentException(???Exception e){  
 …  
 }  
 //rest기반 통신시 에러 중에서 없는 url로 겟 요청하면 json이 아닌 에러페이지가 응답으로 오는데  
 //에러인데, 응답 상태코드가 200으로 정상으로 도착한다.  
 //좋지 않다. 해당 요청의 응답으로 상태코드가 알맞으며 json형식으로 응답이 구성되게끔 하자.  
 @ExceptionHandler( { NoSuchElementException.class, … } ) //404 : 없는 것을 get할 때 에러코드  
 protected ResponeEntity<String> handleIllegalArgumentException(NoSuchElementException e){  
 return ResponseEntity  
 .status(HttpStatus.NOT\_FOUND)  
 .header(“Content-Type”, “text/plain;charseet=UTF-8”)  
 .body(“해당 자원이 없습니다”);  
 }  
 //번외 405 : 지정한 url은 맞는데 지정한 메소드로 된건 없다!  
  
 @ExceptionHandler(Exception.class)//500에러 : 서버 내부 예외  
 protected ResponseEntity<String> handleExcption(Exception e){  
 return ResponseEntity  
 .status(HttpStatus.INTERNAL\_SERVER\_ERROR)  
 .header(“Content-Type”, “text/plain;charseet=UTF-8”)  
 .body(e.getMessage( ));  
 }  
}