spring_boot_annotation

- description : spring boot annotation
- author :
- email : hylee@repia.com
- lastupdate : 2020-08-21

Annotation

A

B

C

> @ComponentScan

- @Component   @Service, @Repository, @Controller, @Configuration   Bean
  Context   bean   Annotation
- @Component   @Repository, @Service, @Controller
  @Repository   DAO   unchecked exception
  DataAccessException   .
  .

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

@EnableAutoConfiguration

-   .

classpath   .   tomcat-embed-core.jar
setting   .

@Configuration

- Configuration   @Bean   @Autowired
                         .

@Required

- setter   .

@Qualifier("id123")

- @Autowired   ,

(ex)   ,   ,   ,   ,   )

@Resource

- @Autowired   Autowired   , Resource
               .

@PostConstruct, @PreConstruct

-   /   (pre/post)
               .

@PreDestroy

-                          (pre)   .

@PropertySource

-                              Environment   .

               @PropertySource("classpath:/settings.properties")   @Resource
Environment   .

@ConfigurationProperties

- yaml   . default   classpath:application.properties   .

                                        (prefix="mail")

mail.host = mailserver@mail.com

mail.port = 9000

mail.defaultRecipients[0] = admin@mail.com

mail.defaultRecipients[1] = customer@mail.com

@Lazy

-                                    .

@Component   @Bean   bean
                                                              .

@Value

- properties   .

@Value("${value.from.file}")

private String valueFromFile;   value.from.file
        .

spEL   .

@Value(#{systemProperties['priority'] ?: 'some default'})

@SpringBootApplication

- @Configuration, @EnableAutoConfiguration, @ComponentScan 3
        .

@RequestMapping

-        URL   mapping   .

                                        .

@RequestMapping("/list"), @RequestMapping("/home, /about");

@RequestMapping("/admin", method=RequestMethod.GET)

@CookieValue

-

                        500   .

        required   default   true . false   null
                .

public String view(@CookieValue(value="auth")String auth){...};   key   auth
        *@CrossOrigin - CORS   origin
AJAX   . @RequestMapping
ajax   . @CrossOrigin(origins = "http://jeong-pro.tistory.com", maxAge = 3600) →
            http://jeong-pro.tistory.com   ajax   . @ModelAttribute - view
                            (VO/DTO)   binding
<input name="id" />   name   set
                    . class Person{ String id; public void setId(String id){ this.id = id;} }
@GetMapping - @RequestMapping(Method=RequestMethod.GET)   @PostMapping,
@PutMapping, @PatchMapping, @DeleteMapping   . @SessionAttributes -
                        @SessionAttributes("name")   Model   key   "name"
                            . @Valid -   .
@InitBinder - @Valid
                . @RequestAttribute - Request   .
@RequestBody -   (JSON   XML   )   model
        @RequestHeader - Request   header   .
@RequestHeader(value="Accept-Language")String acceptLanguage   ko-KR,ko;q=0.8,en-
US;q=0.6

@RequestParam

- @PathVariable   . request   parameter   .

@RequestPart

- Request   MultipartFile   .

@RequestPart("file")MultipartFile file   .

@ResponseBody

- view   JSON   http
response header   response body   .

                   return   JACKSON   .

context   resolver   . (viewResolver)

@PathVariable

-   URL   {   }   .

@ExceptionHandler(ExceptionClassName.class)

-   .

@ControllerAdvice

-                ControllerAdvice
@ExceptionHandler(.class)   .

@RestControllerAdvice

-                @ControllerAdvice + @ResponseBody   .

@ResponseStatus

-                    response code   reason

@ResponseStatus(value = HttpStatus.NOT_FOUND, reason = "my page URL changed..") ⇒
                 .

@EnableEurekaServer

- Eureka   .

@EnableDiscoveryClient

- Eureka

@Transactional

-

@Transaction(readOnly=true, rollbackFor=Exception.class) : readOnly
rollbackFor　　　　　Exception

@Transaction(noRollbackFor=Exception.class)　　　　Exception

@Transaction(timeout = 10)　　10　　　　　　　　　　　　　　　　　　　.

@Cacheable

-

　　　　　　　　　　　　　　　　　　　　　(=　　　　　　)　　　　　　　.

　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　.

　.

@Cacheable(value="cacheKey"), @Cacheable(key="cacheKey")

@CachePut

-

　　　　　　　　　　　　　　　　　　.　　　　　　@Cacheable
　.

@CacheEvict

-

　　　　　　　　　　　　　　.
　(　　　　　　　)　　　　　　　　　　　.

　　　　　　　　　　　　　　.

@CacheEvict(value="cacheKey"), @CacheEvict(value="cacheKey", allEntries=true)

allEntries　　　　　　　　　　　　.

@CacheConfig

-　　　　　　　　　　　　　　　　.

@Scheduled

- Linux　　crontab　　　　　　　　　　　　　　.

@Scheduled(cron = "0 0 07 * * ?") "　　　　　　　　　　(　　)

　　　　, Springboot, annotation

From:
http://125.132.25.164/dokuwiki/ -   . - 2023.12

Permanent link:
http://125.132.25.164/dokuwiki/doku.php?id=wiki:springboot:spring_boot_annotation:spring_boot_annotation&rev=1597983313

Last update: **2022/03/10 19:52**