이진호[서울_1반_A10...
나

이상현컨설턴트 (호스트)

이상현컨설턴트님의 화면 보…

이진호[서울_1반_...
나

이상현컨설턴트
호스트

권영린[서울_1반_A102...

김민재[서울_1반_A102]

김용훈[서울_1반_A105]

김지훈[서울_1반_A105...

나승호[서울_1반_A101...

SAMSUNG SW ACADEMY FOR YOUTH

How to use

@ExceptionHandler같은 경우는 @Controller, @RestController가 적용된 Bean내에서 발생하는 예외를 잡아서 하나의 메서드에서 처리해주는 기능을 한다.

```
@RestController
public class MyRestController {
    ...
    ...
    @ExceptionHandler(NullPointerException.class)
    public Object nullex(Exception e) {
        System.err.println(e.getClass());
        return "myService";
    }
}
```

보낼 곳: 모두

여기에 채팅 메시지를 입력하십시오.

Webex

미팅 정보

이진호[서울_1반_A10...
나

이상현컨설턴트 (호스트)

참가자 (29)

검색

이진호[서울_1반_...
나

이상현컨설턴트
호스트

권영린[서울_1반_A102...

김민재[서울_1반_A102]

김승현[서울_1반_A105...

김용훈[서울_1반_A105]

김지훈[서울_1반_A105...

채팅

SAMSUNG SW ACADEMY FOR YOUTH

Controller Advice

@ControllerAdvice

@ExceptionHandler가 하나의 클래스에 대한 것이라면, @ControllerAdvice는 모든 @Controller 즉, 전역에서 발생할 수 있는 예외를 잡아 처리해주는 annotation이다.

```
@RestControllerAdvice
public class MyAdvice {
    @ExceptionHandler(CustomException.class)
    public String custom() {
        return "hello custom";
    }
}
```

보낼 곳: 모두

여기에 채팅 메시지를 입력하십시오.

음소거 해제 | 비디오 시작 | 공유 | 참가자 | 채팅

Webex

미팅 정보

이진호[서울_1반_A10...
나

참가자 (29)

검색

이진호[서울_1반_...
나

이상현컨설턴트
호스트

권영린[서울_1반_A102...

김민재[서울_1반_A102]

김승현[서울_1반_A105...

김용훈[서울_1반_A105]

김지훈[서울_1반_A105...

채팅

SAMSUNG SW ACADEMY FOR YOUTH

청출어람

```java
@ControllerAdvice
public class ExceptionController {
        @Autowired
        private NotificationManager notificationManager;
        @ExceptionHandler(Exception.class)
        public ResponseEntity exceptionTest(Exception e, HttpServletRequest req) {
                StringBuilder params = new StringBuilder();
                Enumeration<String> keys = req.getParameterNames();
                while (keys.hasMoreElements()) {
                        String key = keys.nextElement();
                        params.append("- ").append(key).append(" : ").append(req.getParameter(key)).append('\n');
                }

                ErrorResponse result = new ErrorResponse();
                e.printStackTrace();
                notificationManager.sendNotification(e, req.getRequestURI(), params.toString());

                result.status = false;
                result.msg = "잘못된 요청입니다!";
                return new ResponseEntity<>(result, HttpStatus.BAD_REQUEST);
        }
}
```

보낼 곳: 모두

여기에 채팅 메시지를 입력하십시오.

음소거 해제

비디오 시작

공유

참가자

채팅

Webex

미팅 정보

이진호[서울_1반_A10...
나

이상현컨설턴트 (호스트)

참가자 (29)

검색

이진호[서울_1반_...
나

이상현컨설턴트
호스트

권영린[서울_1반_A102...

김민재[서울_1반_A102]

김승현[서울_1반_A105...

김용훈[서울_1반_A105]

김지훈[서울_1반_A105...

채팅

SAMSUNG SW ACADEMY FOR YOUTH

청출어람

```java
@Component
@ToString
public class MatterMostSender {
	private Logger log = LoggerFactory.getLogger(MatterMostSender.class);

	private boolean mmEnabled = true;
	private String webhookUrl = "https://meeting.ssafy.com/hooks/ou9oee6tq3ytuyzwnid1gmzkar";

	public void sendMessage(Exception excpetion, String uri, String params) {
		if (mmEnabled) {
			try {
				Attachments attachments = new Attachments(excpetion, uri, params);

				String payload = new Gson().toJson(attachments);
				System.out.println(payload);

				RestTemplate restTemplate = new RestTemplate();
				HttpHeaders headers = new HttpHeaders();
				headers.set("Content-type", MediaType.APPLICATION_JSON_VALUE);

				HttpEntity<String> entity = new HttpEntity<>(payload, headers);
				restTemplate.postForEntity(webhookUrl, entity, String.class);
			} catch (Exception e) {
				log.error("ERROR!! While send MatterMost Message : " + e.getMessage());
			}
		}
	}

}
```

보낼 곳: 모두

여기에 채팅 메시지를 입력하십시오.

음소거 해제

비디오 시작

공유

참가자

채팅