



포팅 메뉴얼 - 외부 서비스 정보

네이버 클로바 더빙

- 동화책 나레이션에 사용 [시연용]
- <https://clovadubbing.naver.com/>
- '새 프로젝트 생성' 후 대사에 맞는 목소리 선별하여 작업

네이버 CLOVA Speech Synthesis(CSS)

- 동화책 나레이션에 사용 [비시연용, 영어 나레이션]
- <https://www.gov-ncloud.com/product/aiService/css>
- 참고 java API

```
import java.io.*;
import java.net.HttpURLConnection;
import java.net.URL;

public class Main {

    public static void main(String[] args) {
        String clientId = "YOUR_CLIENT_ID";           // Application Client ID";
        String clientSecret = "YOUR_CLIENT_SECRET";    // Application Client Secret";

        try {
            String imgFile = "음성 파일 경로";
            File voiceFile = new File(imgFile);

            String language = "Kor";                  // 언어 코드 ( Kor, Jpn, Eng, Chn )
            String apiURL = "https://naveropenapi.apigw.gov-ntruss.com/recog/v1/stt?lang=" + language;
            URL url = new URL(apiURL);

            HttpURLConnection conn = (HttpURLConnection)url.openConnection();
            conn.setUseCaches(false);
            conn.setDoOutput(true);
            conn.setDoInput(true);
            conn.setRequestProperty("Content-Type", "application/octet-stream");
            conn.setRequestProperty("X-NCP-APIGW-API-KEY-ID", clientId);
            conn.setRequestProperty("X-NCP-APIGW-API-KEY", clientSecret);

            OutputStream outputStream = conn.getOutputStream();
            FileInputStream inputStream = new FileInputStream(voiceFile);
            byte[] buffer = new byte[4096];
            int bytesRead = -1;
            while ((bytesRead = inputStream.read(buffer)) != -1) {
                outputStream.write(buffer, 0, bytesRead);
            }
            outputStream.flush();
            inputStream.close();
            BufferedReader br = null;
            int responseCode = conn.getResponseCode();
```

```

        if(responseCode == 200) { // 정상 호출
            br = new BufferedReader(new InputStreamReader(conn.getInputStream()));
        } else { // 에러 발생
            System.out.println("error!!!!!! responseCode= " + responseCode);
            br = new BufferedReader(new InputStreamReader(conn.getInputStream()));
        }
        String inputLine;

        if(br != null) {
            StringBuffer response = new StringBuffer();
            while ((inputLine = br.readLine()) != null) {
                response.append(inputLine);
            }
            br.close();
            System.out.println(response.toString());
        } else {
            System.out.println("error !!!");
        }
    } catch (Exception e) {
        System.out.println(e);
    }
}
}

```

구글 SMTP

- AiRang 구글 계정 생성
- Google 로그인 [2단계 인증] 사용

← 2단계 인증

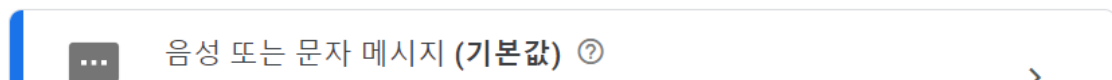
2022. 9. 22.에 2단계 인증을 사용하도록 설정했습니다.

사용 안함

사용 가능한 인증 2단계

비밀번호를 입력한 후 로그인하는 사용자의 본인 여부를 확인하는 2단계입니다. [자세히 알아보기](#)

참고: 지원 휴대전화에서 Google 계정에 로그인하면 Google 메시지가 2단계 인증을 위한 다른 수단으로 추가됩니다.



- 앱 비밀번호 생성
- application.yml 설정 추가

```
#smtp
mail:
  host: smtp.gmail.com
  port: 587
  username: airangb305@gmail.com//임시 비밀번호 발급
  password: *****
  properties:
    mail:
      smtp:
        auth: true
        timeout: 5000
        starttls:
          enable: true
```

- UserController에 mail메서드 생성

```
@NoJwt
@PostMapping("/mail")
public ResponseEntity<String> sendMail(@RequestBody Map<String, String> request) throws SQLException {
    String email = request.get("email");
    if(userService.findUserByEmail(email) != null && userService.makeTmpPw(email) != "") {
        return new ResponseEntity<String>("메일 전송 성공", HttpStatus.OK);
    }
    return new ResponseEntity<String>("메일 전송 실패", HttpStatus.NOT_FOUND);
}
```

- MailService

```
@Service
@RequiredArgsConstructor
@Transactional
public class MailService {

    private final JavaMailSender mailSender;

    private static final String title = "[아이랑] 임시 비밀번호 안내 이메일입니다.";

    private static final String message = "안녕하세요. [아이랑] 임시 비밀번호 안내 메일입니다. "
        + "\n" + "회원의 임시 비밀번호는 아래와 같습니다. 로그인 후 반드시 비밀번호를 변경해주세요." + "\n";
    private static final String fromAddress = "helpAiRang@gmail.com";

    public Mail createMail(String tmpPassword, String memberEmail) {
        Mail mail = Mail.builder()
            .toAddress(memberEmail)
            .title(title)
            .message(message + tmpPassword)
            .fromAddress(fromAddress)
            .build();

        return mail;
    }

    public void sendMail(Mail mail) {
        SimpleMailMessage mailMessage = new SimpleMailMessage();
        mailMessage.setTo(mail.getToAddress());
        mailMessage.setSubject(mail.getTitle());
        mailMessage.setText(mail.getMessage());
        mailMessage.setFrom(mail.getFromAddress());
    }
}
```

```
        mailMessage.setReplyTo(mail.getFromAddress());

        mailSender.send(mailMessage);
    }
}
```

- Mail Dto

```
@Getter
@Setter
@ToString
@Builder
public class Mail {
    private String toAddress; //수신인
    private String title; //제목
    private String message; //내용
    private String fromAddress; //송신인
}
```