

```
1 Lab. Lambda, API Gateway Version Control
2
3 [실습목표]
4 1. Lambda의 버전 관리를 할 수 있다.
5 2. Lambda 별칭을 사용하여 API Gateway와 연동할 수 있다.
6 3. Lambda 별칭을 활용하여 운영소스, 개발소스, 테스트 소스를 관리할 수 있다.
7
8
9 [실습순서]
10 1. Lambda 버전 생성 , 별칭 생성
11 2. API Gateway - Lambda 별칭 연결
12 3. API Gateway 배포
13 4. Lambda 소스 수정 및 신규 버전 생성
14 5. Lambda 버전 - 별칭 변경
15
16
17 1. Lambda 버전 및 별칭 생성
18 1)[Lambda] > [함수] > 'Lab. Lambda, API Gateway'에서 생성한 함수에서
19 2)[버전] 탭 클릭
20 3)[새 버전 발행] 버튼 클릭
21 4)[$LATEST의 새 버전 발행] 팝업에서 [게시] 버튼 클릭
22 5)[버전: 1]이 새로 생성된 것을 확인
23 6)버전 생성 확인 후 함수명을 클릭하여 함수 관리 화면으로 이동
24 7)[별칭] 탭 클릭
25
26 8)[별칭 생성] 버튼 클릭
27   -[이름] : production
28   -[버전] : '1' 선택
29   -[저장] 버튼 클릭
30
31 9)[별칭] 생성(production) 확인
32 10)함수명을 클릭하여 함수 관리 화면으로 이동
33
34 11)[별칭 생성] 버튼 클릭
35   -위와 같은 방법으로 별칭 추가 생성 -> 별칭 : dev, test
36
37 12)[별칭] 탭 클릭
38   -'production', 'test', 'dev' 총 3개의 [별칭]이 생성되었는지 확인
39
40 13)[버전] 탭 클릭
41   -[버전 1]에 'production', 'test', 'dev' 별칭이 매핑 되었는지 확인
42
43
44
45 2. API Gateway 리소스 생성
46 1)[API Gateway] > [API] > 'Lab. Lambda, API Gateway'에서 생성한 API Gateway 선택
47 2)[작업] > [리소스 생성] 메뉴 클릭
48
49 3)우측 [새 하위 리소스] 에서
50   -[리소스 이름] : {계정이름}-greeting-production
51   -[리소스 경로] : {계정이름}-greeting-production
52   -[리소스 생성] 버튼 클릭
53
54 4)[리소스] 영역에 위에서 만든 [리소스] 선택
55
56 5)[작업] > [메소드 생성] 메뉴 클릭
57   -[메소드] > 'GET' 선택 후 체크 버튼 클릭
58
59 6)우측 [GET 설정] 영역의 [Lambda 함수]에 'Lab. Lambda, API Gateway'에서 생성한 Lambda의 별칭 production의 ARN을 복사하여 붙임
60   -ex)arn:aws:lambda:ap-northeast-2:503237308475:function:dgb-00-demo:production
61
62 7)[저장] 버튼 클릭
63 8)[Lambda 함수 권한 추가] 팝업창에서 [확인] 클릭
64
65 9)위에서 만든 'GET' 메소드 선택
66   -[테스트]를 클릭하여 [GET - 메서드 테스트] 화면으로 이동
67   -[테스트] 버튼을 클릭
68   -우측의 테스트 결과 확인 -> Production 별칭 Lambda의 결과가 리턴 되는지 확인
69   --"Hello World!"
70
71
72
73 3. API Gateway 배포
74 1)[API Gateway] > [API] > 위의 2번에서 작업한 API
75 2)위의 2번에서 작업한 리소스 선택
76 3)[작업] > [API 배포] 메뉴 선택
77 4)[API 배포] 팝업창에서, [배포 스테이지] : 'Lab. Lambda, API Gateway'에서 생성한 스테이지 선택 선택
78 5)[배포] 버튼 클릭
79 6)[스테이지] > 스테이지 이름 > 위의 2번에서 작업한 리소스명 > 'GET' 메소드 클릭
80 7)[URL 호출]을 클릭하여 브라우저에서 접속 확인
81   "Hello World!"
82
83
84
```

```

85 4. Lambda 소스 수정 및 신규 버전 생성
86 1)[Lambda] > [함수] > 'Lab. Lambda, API Gateway'에서 생성한 함수
87 2)[코드] 탭에서 소스 수정
88     import json
89
90     print ('Lamba Start!')
91
92     def lambda_handler(event, context):
93         print('Received Event : ' + json.dumps(event, indent=2))
94         greet = 'Hello'
95         name = 'World'
96
97         if 'greet' in event:
98             greet = event['greet']
99
100        if 'name' in event:
101            name = event['name']
102
103        greetings = greet + ' ' + name + '!!' + ':ver2'    <---여기만 수정
104        print(greetings)
105        return greetings
106
107 3)[Deploy] 버튼을 클릭하여 소스 배포
108 4)[버전] 탭 클릭
109 5)[새 버전 발행] 버튼 클릭
110 6)[$LATEST의 새 버전 발행] 창에서 [게시] 버튼 클릭
111 7)[버전 2]가 생성된 것을 확인
112 8)Web Browser브라우저에 위의 3번에서 배포한 API URL을 접속하여 수정된 소스 반영여부 확인
113 9)하지만, 수정된 소스에 'production' 별칭이 매핑되지 않았으므로 변경된 소스가 반영되지 않는 것이 정상
114
115
116
117 5. Lambda 버전 - 별칭 변경
118 1)[Lambda] > [함수] > 'Lab. Lambda, API Gateway'에서 생성한 함수
119 2)[별칭] 탭 클릭
120 3)[production] 별칭 클릭
121 4)[구성] > [일반 구성] > [편집] 버튼 클릭
122     -[버전] : '2'로 선택
123     -[저장] 버튼 클릭
124 5)다시 Web Browser에 위의 3에서 배포한 API URL을 접속하여 변경된 별칭 버전 반영 여부 확인
125     -> 버전2 소스에 production 별칭이 매핑 되었으므로 버전2 소스가 반영되어야 정상
126     "Hello World!!:ver2"

```