

(IoT Network) Practice -4-

CoAP Method Option



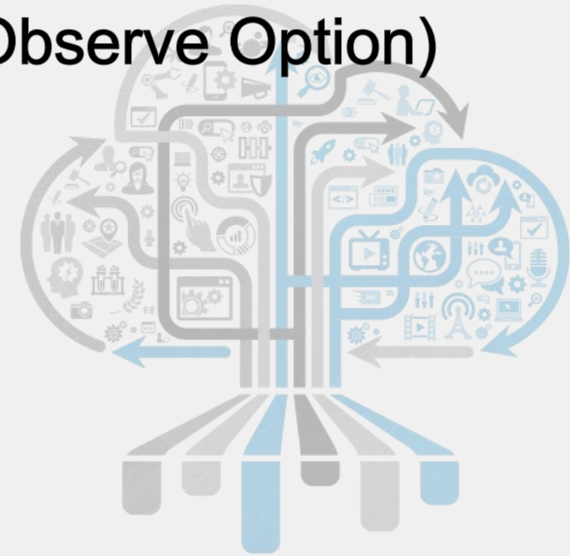
Index

I. Practice 3 Review

II. CoAP Observe

III. jCoAP Observe Basic Example

1. Define Temperature Resource
2. Server (Add Temperature Resource)
3. Client (Add a Button for GET Request with Observe Option)



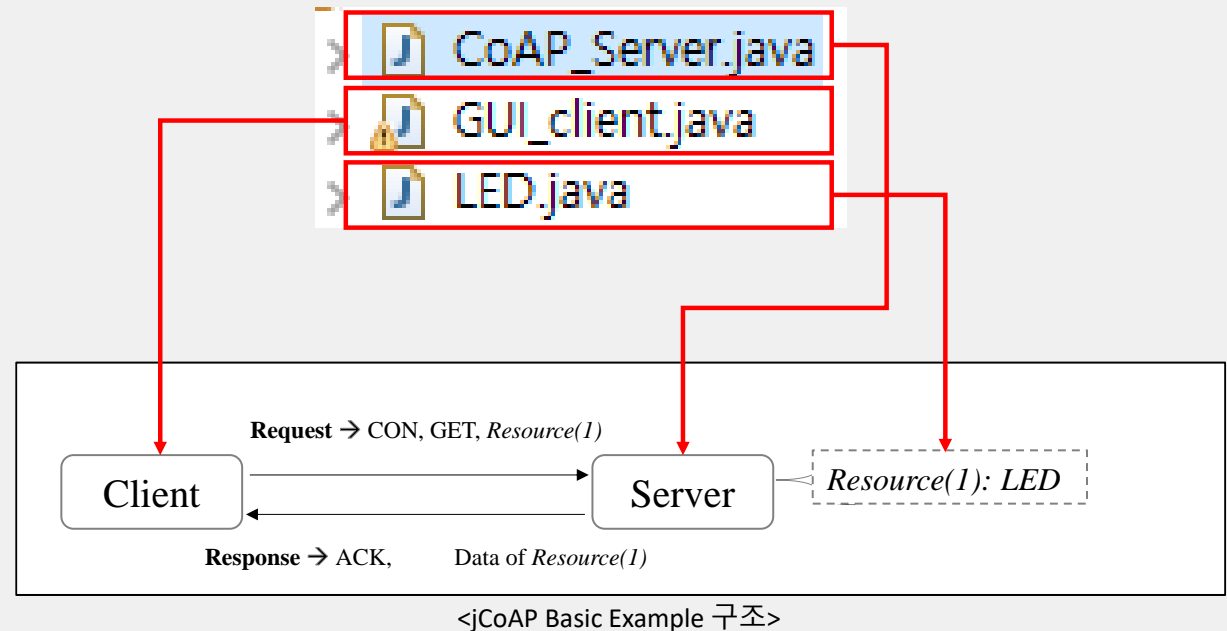
Practice 3 Review

- **CoAP Messages**

- Confirmable (CON): 신뢰성을 보장하는 메시지
- Non-confirmable (NON): 신뢰성을 보장하지 않는 메시지
- Acknowledgement (ACK): CON 메시지에 대한 응답 메시지
- Reset (RST): 요청을 처리할 수 없을 때, 처리불가를 알리는 메시지

- **CoAP Methods**

- GET: 리소스 값 읽어오기
- PUT: 리소스 값 변경하기
- POST: 새로운 리소스 생성
- DELETE: 리소스 삭제

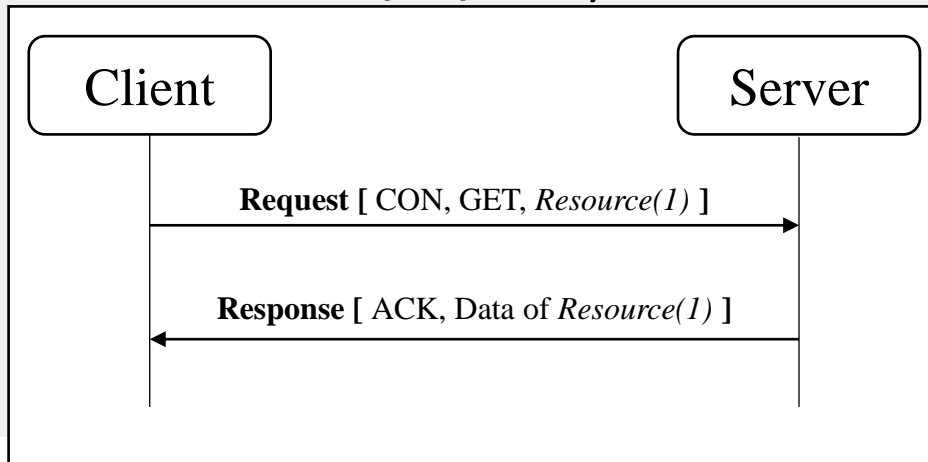


CoAP Observe

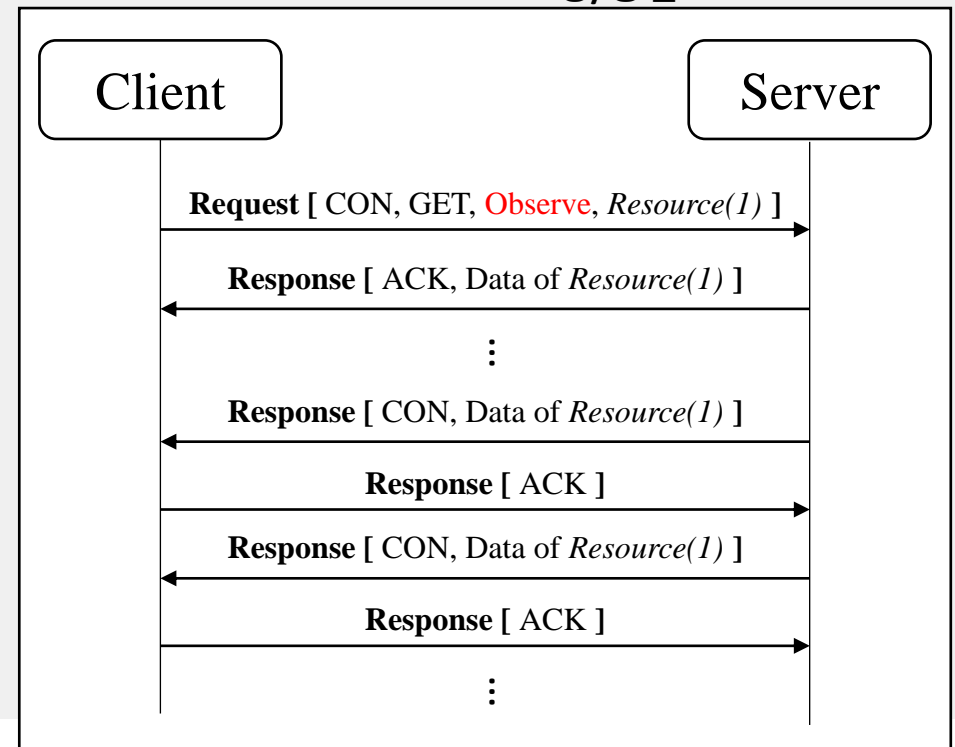
• Observe

- Client가 특정 리소스를 지속적으로 관찰하기 위한 Option
 - ✓ Client가 특정 리소스에 대해 Observe 옵션을 포함한 GET 요청
 - ✓ Server가 Observe 옵션을 포함한 GET 요청을 수신함
 - ✓ 리소스의 데이터값이 변화하면, Server는 요청이 없어도 클라이언트에게 리소스 데이터를 포함하는 응답 메시지 전송

<CoAP 기본적인 요청/응답>

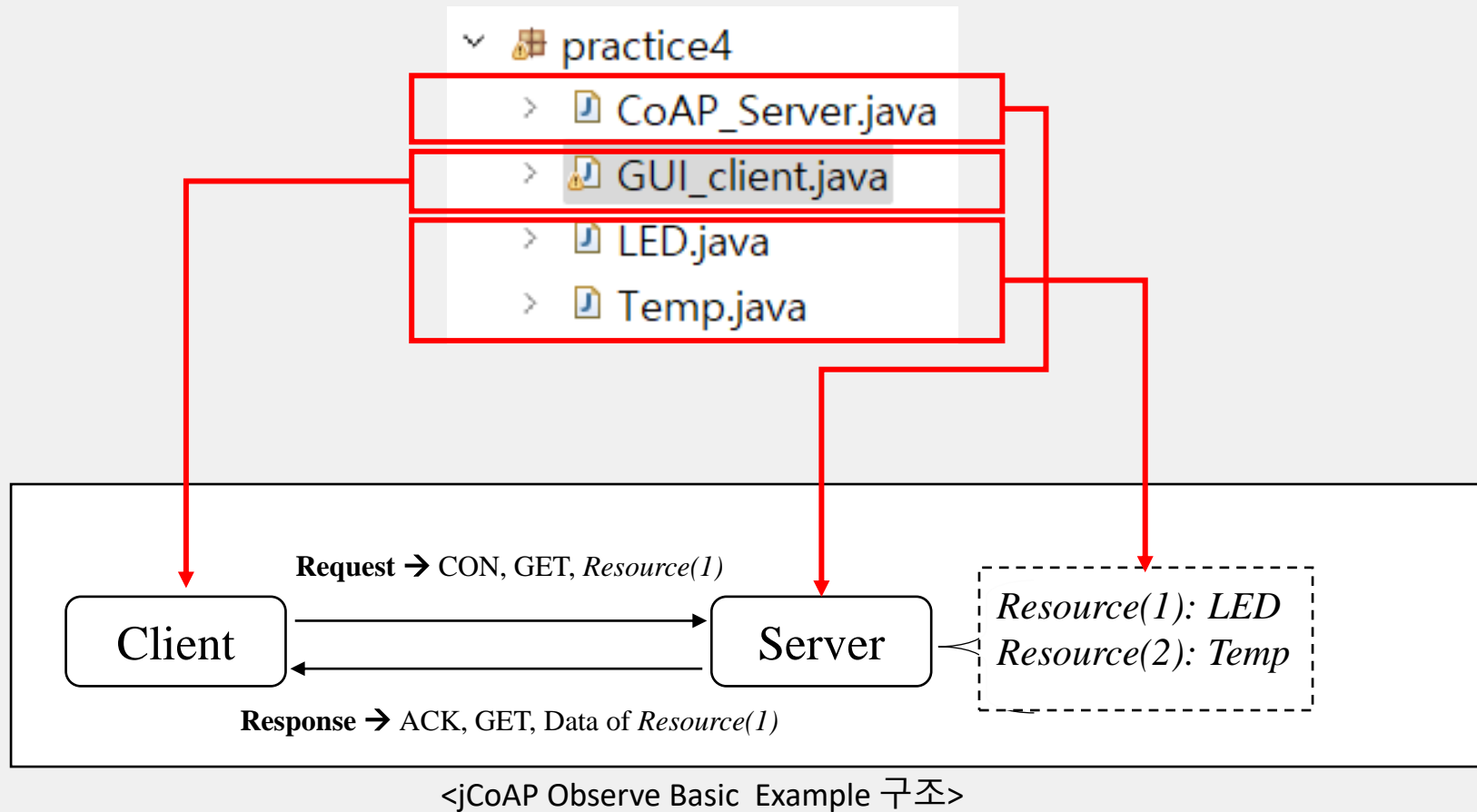


<CoAP Observe 요청/응답>



jCoAP Observe Basic Example

- jCoAP Observe Basic Example 구조



jCoAP Observe Basic Example

- jCoAP Basic Example – Define Temperature Resource

```
1 package practice4;
2 import java.util.List;
3 import org.ws4d.coap.core.enumerations.CoapMediaType;
4 import org.ws4d.coap.core.rest.BasicCoapResource;
5 import org.ws4d.coap.core.rest.CoapData;
6 import org.ws4d.coap.core.tools.Encoder;
7
8 public class Temp extends BasicCoapResource{
9     private String value = "0";
10
11     private Temp(String path, String value, CoapMediaType mediaType){
12         super(path, value, mediaType);
13     }
14     public Temp() {
15         this("/temp", "0", CoapMediaType.text_plain);
16     }
17     @Override
18     public synchronized CoapData get(List<String> query, List<CoapMediaType> mediaTypesAccepted) {
19         return get(mediaTypesAccepted);
20     }
21     @Override
22     public synchronized CoapData get(List<CoapMediaType> mediaTypesAccepted) {
23         int sensing_data = (int)((Math.random()*10000%10));
24         this.value = Integer.toString(sensing_data);
25         return new CoapData(Encoder.StringToByte(this.value), CoapMediaType.text_plain);
26     }
27     @Override
28     public synchronized boolean setValue(byte[] value) {
29         this.value = Encoder.ByteToString(value);
30         return true;
31     }
32     @Override
33     public synchronized boolean post(byte[] data, CoapMediaType type) {
34         return this.setValue(data);
35     }
36     @Override
37     public synchronized boolean put(byte[] data, CoapMediaType type) {
38         return this.setValue(data);
39     }
40     @Override
41     public synchronized String getResourceType() {
42         return "Temperature";
43     }
44 }
```

jCoAP Observe Basic Example

- jCoAP Basic Example – **Server (Add Temperature Resource)**

```
public void start() {
    System.out.println("===Run Test Server ===");

    // 1. create server
    if (this.resourceServer != null)    this.resourceServer.stop();
    this.resourceServer = new CoapResourceServer();

    // 2. initialize resource
    LED led = new LED();
    Temp temp = new Temp();
    temp.setObservable(true);

    // 3. add resource to server
    this.resourceServer.createResource(led);
    this.resourceServer.createResource(temp);
    temp.registerServerListener(resourceServer);

    // run the server
    try {
        this.resourceServer.start();
    } catch (Exception e) {
        e.printStackTrace();
    }

    //Observe 조건 - 5초에 한번씩 온도센서 값 자동 전달
    while(true) {
        try {
            Thread.sleep(5000);
        }
        catch (Exception e) {
            // TODO: handle exception
        }
        temp.changed();
    }
}
```

jCoAP Observe Basic Example

- jCoAP Basic Example – Client (Add a Button for GET Request with Observe Option)

```
JButton btn_get = new JButton("GET");
JButton btn_post = new JButton("POST");
JButton btn_put = new JButton("PUT");
JButton btn_delete = new JButton("DELETE");
JButton btn_obsget = new JButton("Observe GET");
```

```
//btn
btn_get.setBounds(20, 670, 100, 50);
btn_put.setBounds(130, 670, 100, 50);
btn_post.setBounds(240, 670, 100, 50);
btn_delete.setBounds(350, 670, 100, 50);
btn_obsget.setBounds(460, 670, 130, 50);
```

```
this.add(btn_get);
this.add(btn_post);
this.add(btn_put);
this.add(btn_delete);
this.add(btn_obsget);
this.add(path_text);
this.add(path_label);
this.add(payload_label);
this.add(payload_text);
this.add(display_text_jp);
this.add(display_label);
```

```
btn_obsget.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        String path = path_text.getText();
        String payload = payload_text.getText();
        CoapRequest request = clientChannel.createRequest(CoapRequestCode.GET, path, true);
        request.setObserveOption(0);
        displayRequest(request);
        clientChannel.sendMessage(request);
    }
});
```