

Hello World

Table of Contents

- 1. 仕様..... 1
- 2. 設計..... 1
 - 2.1. TODOリスト 1
 - 2.2. ユースケース図..... 1
 - 2.3. クラス図..... 1
 - 2.4. シーケンス図..... 1
- 3. 実装..... 2
- 4. 参照..... 5

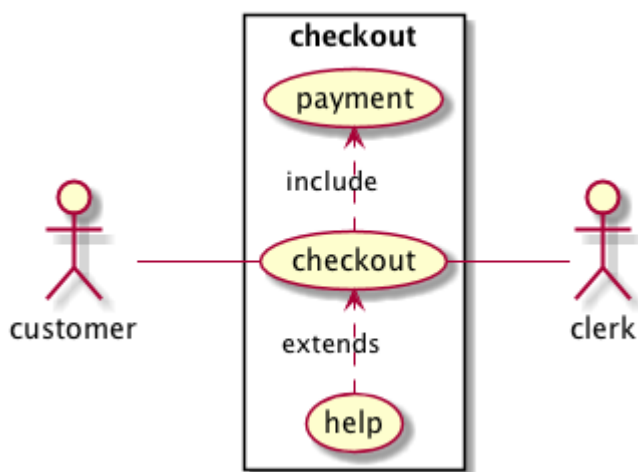
1. 仕様

2. 設計

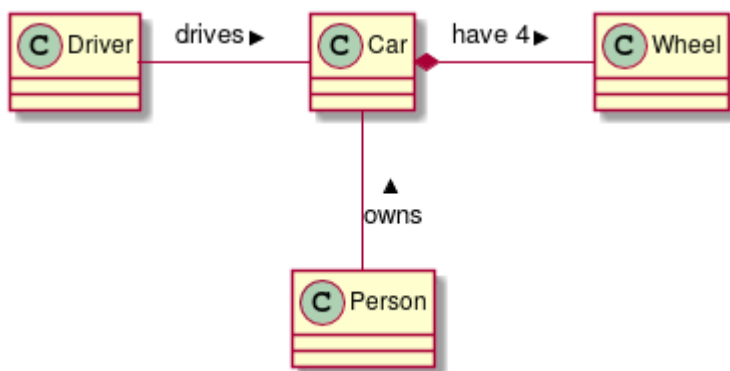
2.1. TODOリスト

- ☐ TODO
- ☒ ~~TODO-DONE~~

2.2. ユースケース図



2.3. クラス図



2.4. シーケンス図



3. 実装

```

package helloworld;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.IOException;
import java.net.URL;
import java.util.HashMap;
import java.util.Map;
import java.util.StringJoiner;

import com.amazonaws.services.lambda.runtime.Context;
import com.amazonaws.services.lambda.runtime.RequestHandler;

/**
 * Handler for requests to Lambda function.
 */
public class App implements RequestHandler<Object, Object> {

    public Object handleRequest(final Object input, final Context context) {
        Map<String, String> headers = new HashMap<>();
        headers.put("Content-Type", "application/json");
        headers.put("X-Custom-Header", "application/json");
        try {
            final String pageContents =
this.getPageContents("https://checkip.amazonaws.com");
            String output = String.format("{ \"message\": \"hello world\",
\"location\": \"%s\" }", pageContents);
            return new GatewayResponse(output, headers, 200);
        } catch (IOException e) {
            return new GatewayResponse("{} ", headers, 400);
        }
    }
}

```

```

private String getPageContents(String address) throws IOException{
    BufferedReader br = null;
    StringJoiner lines = new StringJoiner(System.lineSeparator());
    try {
        URL url = new URL(address);
        br = new BufferedReader(new InputStreamReader(url.openStream()));
        String line;
        while ((line = br.readLine()) != null) {
            lines.add(line);
        }
    } finally {
        if (br != null) {
            br.close();
        }
    }
    return lines.toString();
}
}

```

```

package helloworld;

import java.util.Collections;
import java.util.HashMap;
import java.util.Map;

/**
 * POJO containing response object for API Gateway.
 */
public class GatewayResponse {

    private final String body;
    private final Map<String, String> headers;
    private final int statusCode;

    public GatewayResponse(final String body, final Map<String, String> headers, final
int statusCode) {
        this.statusCode = statusCode;
        this.body = body;
        this.headers = Collections.unmodifiableMap(new HashMap<>(headers));
    }

    public String getBody() {
        return body;
    }

    public Map<String, String> getHeaders() {
        return headers;
    }

    public int getStatusCode() {
        return statusCode;
    }
}

```

```
package helloworld;

import static org.junit.Assert.assertEquals;
import static org.junit.Assert.assertNotNull;
import static org.junit.Assert.assertTrue;
import org.junit.Test;

public class AppTest {
    @Test
    public void successfulResponse() {
        App app = new App();
        GatewayResponse result = (GatewayResponse) app.handleRequest(null, null);
        assertEquals(result.getStatusCode(), 200);
        assertEquals(result.getHeaders().get("Content-Type"), "application/json");
        String content = result.getBody();
        assertNotNull(content);
        assertTrue(content.contains("\"message\""));
        assertTrue(content.contains("\"hello world\""));
        assertTrue(content.contains("\"location\""));
    }
}
```

4. 参照

- PlantUML[<http://plantuml.com>]