

Rushikesh Shinde

Add dependencies:

```
<dependencies>
  <!-- Spring Boot Starter -->
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter</artifactId>
  </dependency>

  <!-- Camel Spring Boot Starter -->
  <dependency>
    <groupId>org.apache.camel</groupId>
    <artifactId>camel-spring-boot-starter</artifactId>
  </dependency>

  <!-- Camel HTTP Component -->
  <dependency>
    <groupId>org.apache.camel</groupId>
    <artifactId>camel-http</artifactId>
  </dependency>

  <!-- JSON processing -->
  <dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-databind</artifactId>
  </dependency>
</dependencies>
```

Add CamelRoute(ApiRoute):-

```
import org.apache.camel.builder.RouteBuilder;
import org.springframework.stereotype.Component;
```

@Component

```
public class ApiRoute extends RouteBuilder {
```

```
    @Override
```

```
    public void configure() throws Exception {
```

```
        from("timer:trigger?repeatCount=1")
            .setHeader("pageNo", constant(2))
            .setHeader("ids", constant("7,8"))
            .to("direct:callApi");
```

```
        from("direct:callApi")
```

```
            .process(exchange -> {
                int pageNo = exchange.getIn().getHeader("pageNo", Integer.class);
                String ids = exchange.getIn().getHeader("ids", String.class);
```

```
                String apiUrl = "https://reqres.in/api/users?page=" + pageNo;
```

```
                exchange.getIn().setHeader("CamelHttpMethod", constant("GET"));
                exchange.getIn().setHeader("CamelHttpUri", apiUrl);
            });
    }
```

```
    })  
    .to("http4://dummy") // dummy endpoint, will be overridden by the header  
  
    // Process the API response and filter based on IDs  
    .unmarshal().json()  
    .split().jsonpath("$.data[?(@.id in [" + "${header.ids}" + "])]")  
    .marshal().json()  
    .to("https://reqres.in/api/users?page=2");  
  }  
}
```