

HTTP Client Interfaces

Spring Boot 3 Workshop - Devnexus 2023 - Atlanta

Jonatan Ivanov & Phil Webb

About the Sample

It's another very silly application 🤪

- Just enough code to demo what we want!
- API using HttpServiceProxy
- Another MVC controller
- Some configuration

HTTP Client Interfaces

Are really interfaces + metadata

```
@GetExchange ("/dogs")  
DogsResponse dogs (  
    @RequestParam (name="aregood") boolean areGood) ;
```

```
@GetExchange ("/owner/{name}/dogs")  
List<String> ownedDogs (@PathVariable String name) ;
```

HTTP Client Interfaces

Records!

```
public record DogsResponse (String message) {  
}
```

HTTP Client Interfaces

How to build the client?

```
WebClient webClient =  
    webClientBuilder.baseUrl ("...").build();
```

```
WebClientAdapter adapter =  
    WebClientAdapter.forClient(webClient);
```

```
HttpServiceProxyFactory factory =  
    HttpServiceProxyFactory.builder(adapter).build();
```

```
return factory.createClient(Api.class);
```

HTTP Client Interfaces

How to use the client?

```
// Signature
```

```
DogsResponse dogs (boolean areGood) ;
```

```
List<String> ownedDogs (String name) ;
```

```
// Usage
```

```
api.dogs (true) ;
```

```
api.ownedDogs ("Phil") ;
```

ApplicationRunner

Prints information when the application starts

- Functional interface called when the app starts:

```
void run(ApplicationArguments args) throws  
Exception
```

Run the application

Check the startup output

```
$ ./mvnw -pl dog-service clean spring-boot:run
```

```
$ ./mvnw -pl dog-client clean spring-boot:run
```


Checkpoint

Client application works