

# 1. Goals

Understand a simple Spring Web Configuration and learn how to tune the HTTP Message Converters.

## 2. Lesson Notes

### 2.1. How can the API be improved

The JSON data from the API should be pretty printed.

The API shouldn't accept JSON data with unknown fields.

### 2.2. Tuning the Web Configuration

We are going to start by having the basic web configuration of the system extend *WebMvcConfigurerAdapter*.

We are also going to override the hook provided by Spring: *extendMessageConverters* - and find the Jackson converter we want to configure:

```
converters.stream().filter(c -> c instanceof MappingJackson2HttpMessageConverter).findFirst();
```

We are then going to configure the converter:

```
AbstractJackson2HttpMessageConverter converter = (AbstractJackson2HttpMessageConverter) converterFound.get();
```

```
converter.getObjectMapper().enable(SerializationFeature.INDENT_OUTPUT);  
converter.getObjectMapper().enable(DeserializationFeature.FAIL_ON_UNKNOWN_PROPERTIES);
```

## 2.3. Tune the Jackson Converters with Spring Boot

Spring Boot can greatly simplify this configuration - by allowing every Jackson property to be set from the *application.properties* file:

```
spring.jackson.serialization.indent-output=true  
spring.jackson.deserialization.fail-on-unknown-properties=true
```

## 3. Resources

- [Spring - WebMvcConfigurer - Javadoc](#)
- [Spring Reference - HTTP Message Converters](#)