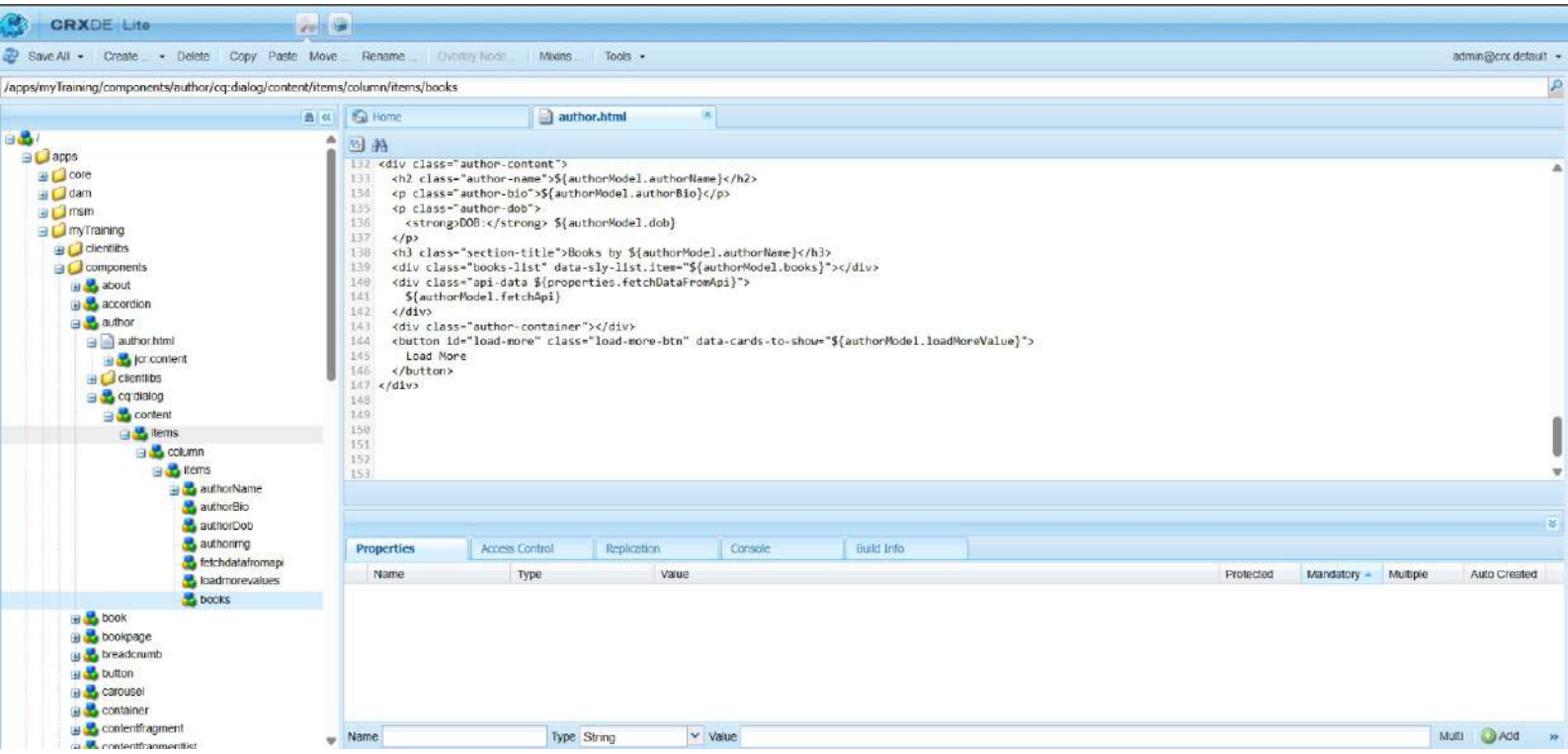

CRXDE Lite
Save All
Create ...
Delete
Copy
Paste
Move ...
Rename ...
Overlay Node
Mixins ...
Tools
admin@crx.default
/apps/myTraining/components/author/cq:dialog/content/items/column/items/books
Home
author.html
/
apps
core
dam
msm
myTraining
clientlibs
components
about
accordion
author
author.html
jcr:content
clientlibs
cq:dialog
content
items
column
items
authorName
authorBio
authorDob
authorImg
fetchdatafromapi
loadmorevalues
books
book
bookpage
breadcrumb
button
carousel
container
contentfragment

<h2 class="author-name">${authorModel.authorName}</h2>
${authorModel.authorBio}

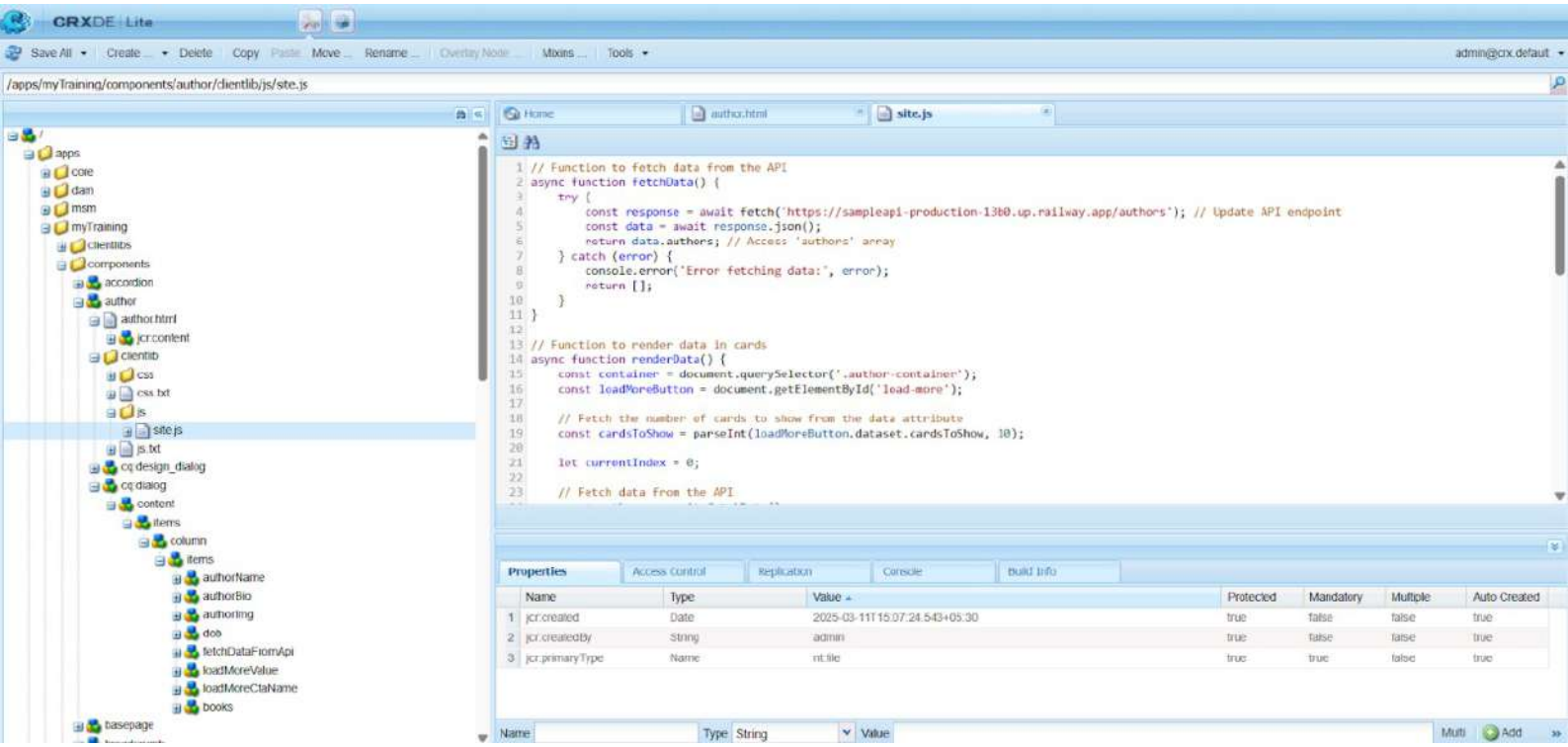
DOB: ${authorModel.dob}

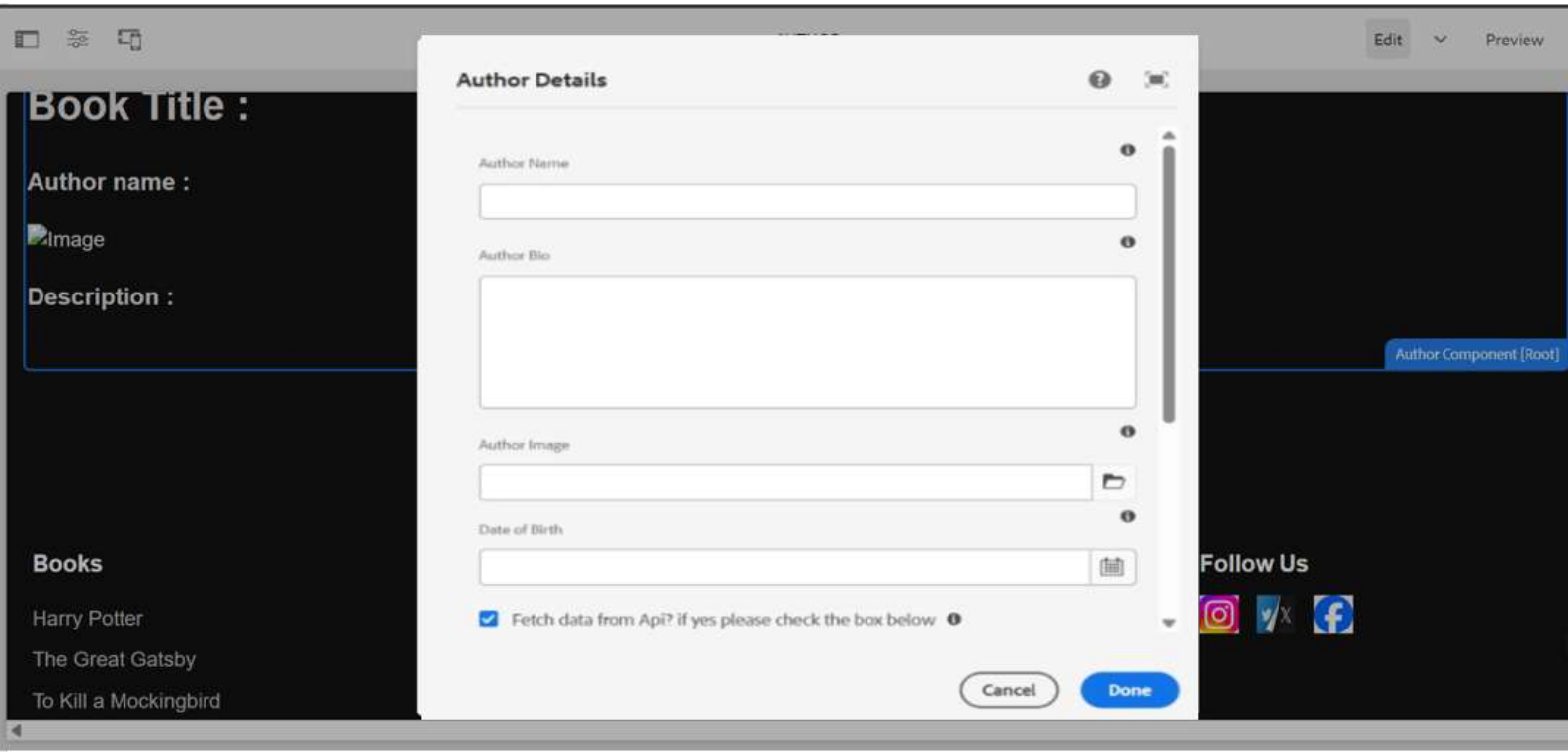
<h3 class="section-title">Books by ${authorModel.authorName}</h3>
<div class="books-list" data-sly-list.item="${authorModel.books}">

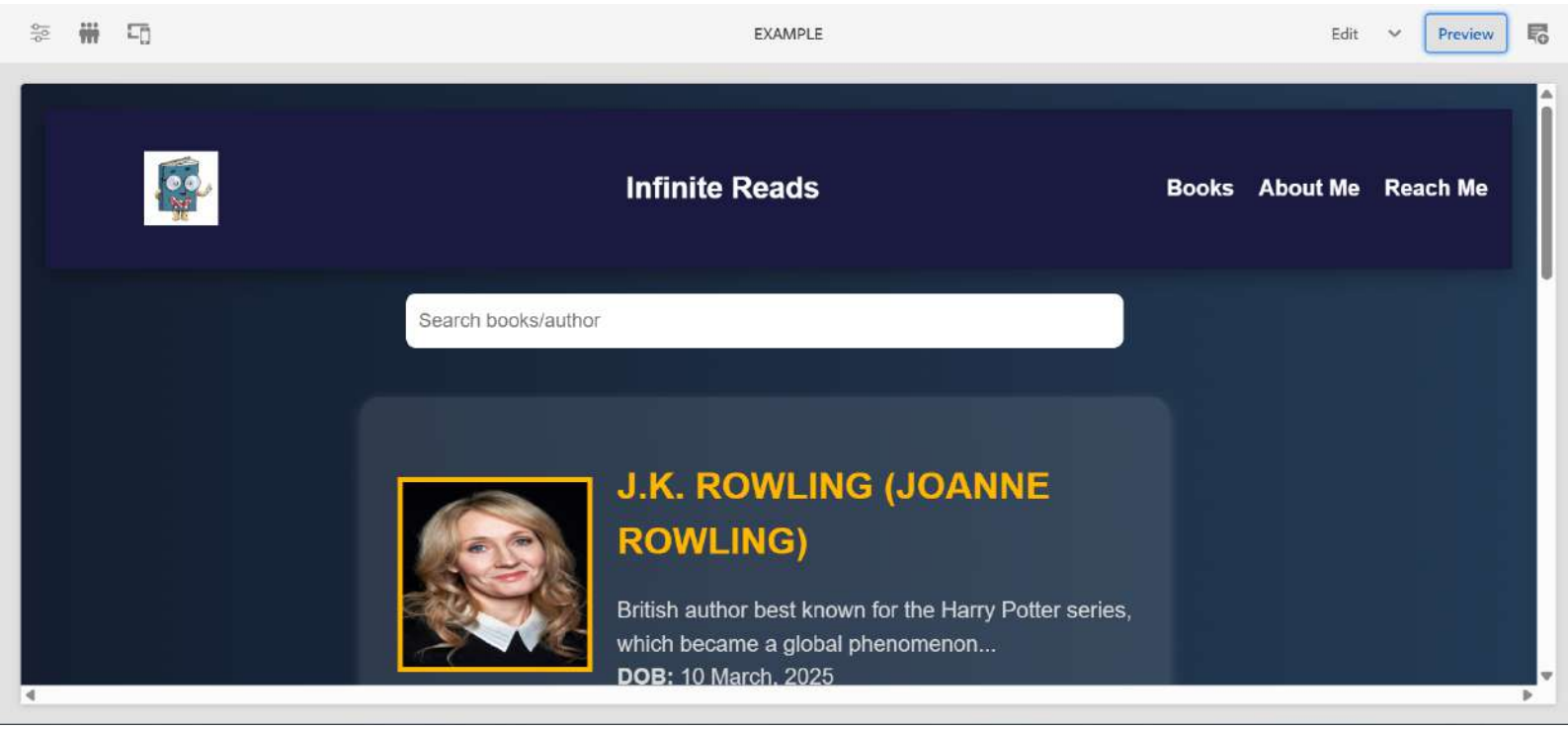
${authorModel.fetchApi}

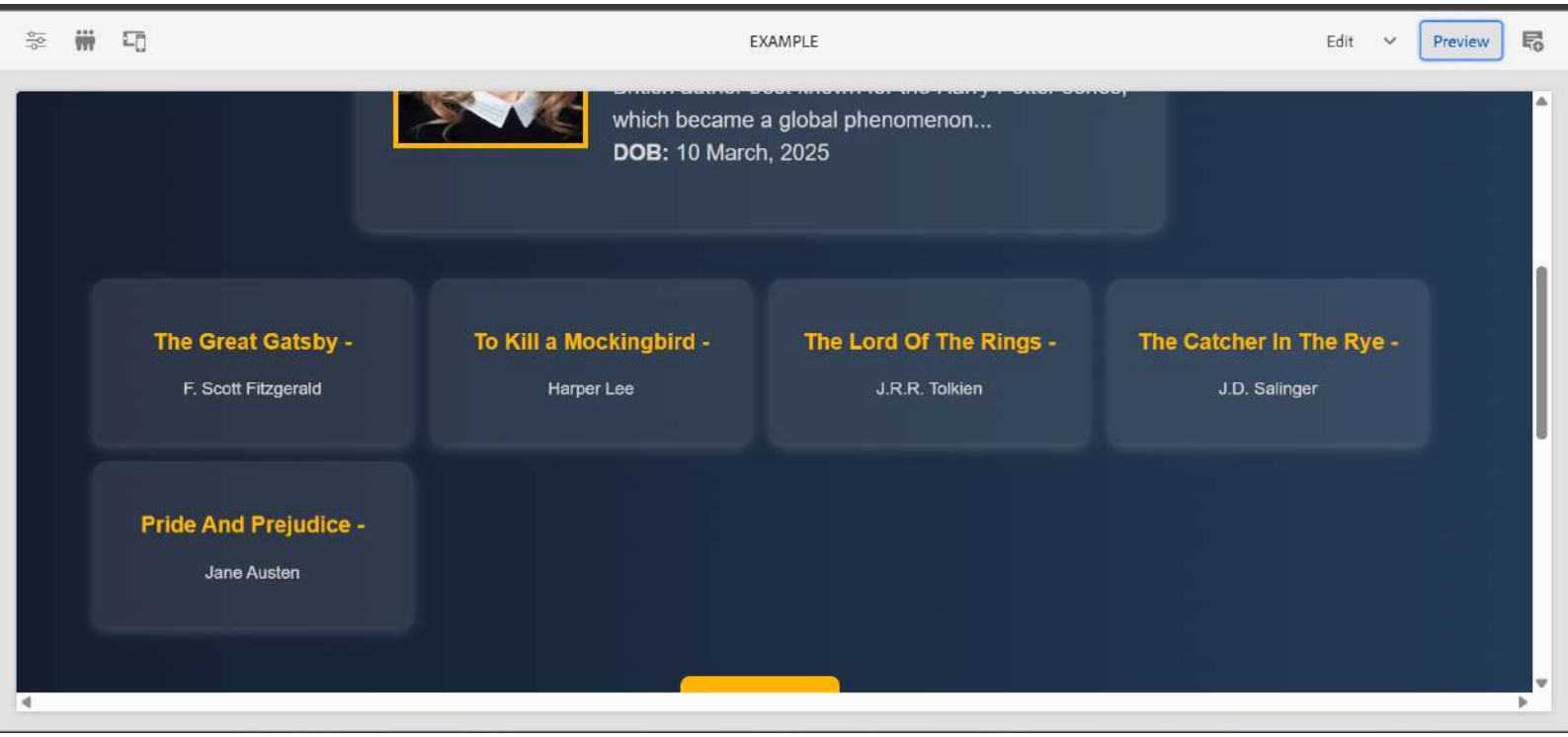
<button id="load-more" class="load-more-btn" data-cards-to-show="${authorModel.loadMoreValue}">
Load More
</button>

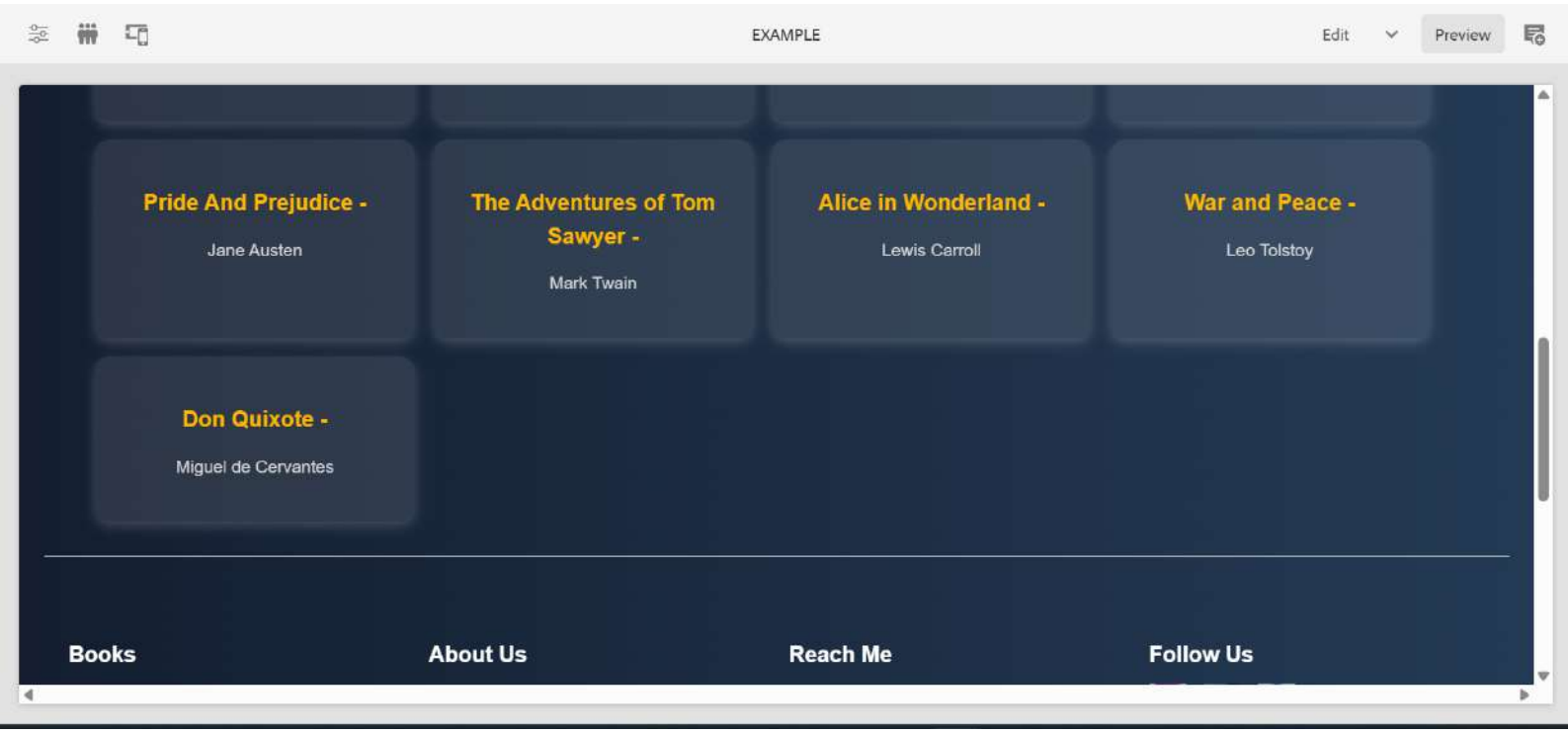
Properties
Access Control
Replication
Console
Build Info
Name
Type
Value
Protected
Mandatory
Multiple
Auto Created
Name
Type
String
Value
Multi
Add

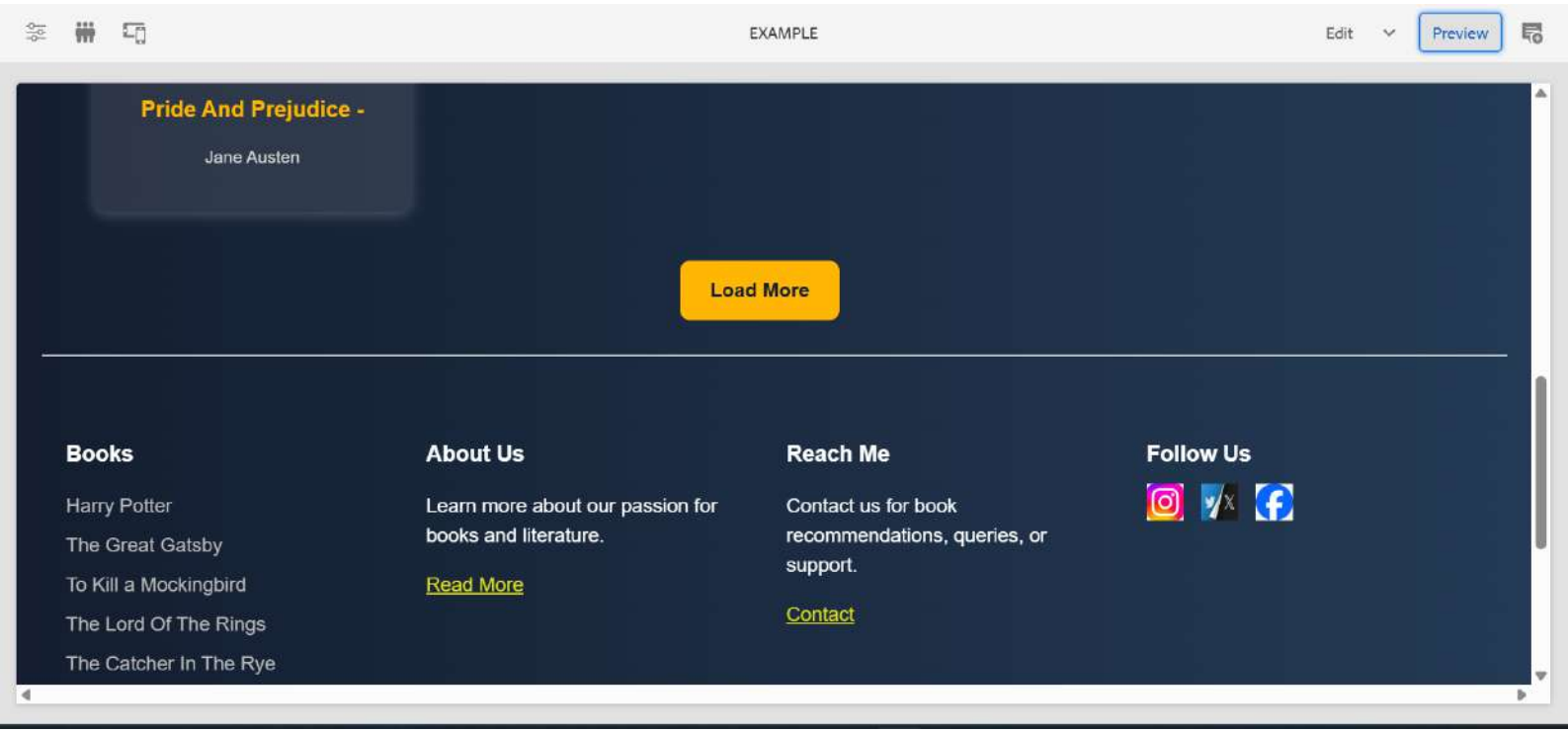

CRXDE Lite
Save All
Create ...
Delete
Copy
Paste
Move ...
Rename ...
Overlay Node ...
Mixins ...
Tools
admin@crx.default
/apps/myTraining/components/author/clientlib/js/site.js
/
apps
core
dam
msm
myTraining
clientlibs
components
accordion
author
author.html
jcr:content
clientlib
css
css.txt
js
site.js
js.txt
cq:design_dialog
cq:dialog
content
items
column
items
authorName
authorBio
authorImg
dob
fetchDataFromApi
loadMoreValue
loadMoreCtaName
books
basepage
Home
author.html
site.js
// Function to fetch data from the API
async function fetchData() {
try {
const response = await fetch('https://sampleapi-production-13b0.up.railway.app/authors'); // Update API endpoint
const data = await response.json();
return data.authors; // Access 'authors' array
} catch (error) {
console.error('Error fetching data:', error);
return [];
}
}
// Function to render data in cards
async function renderData() {
const container = document.querySelector('.author-container');
const loadMoreButton = document.getElementById('load-more');
// Fetch the number of cards to show from the data attribute
const cardsToShow = parseInt(loadMoreButton.dataset.cardsToShow, 10);
let currentIndex = 0;
// Fetch data from the API
Properties
Access Control
Replication
Console
Build Info
Name
Type
Value
Protected
Mandatory
Multiple
Auto Created
1 jcr:created Date 2025-03-11T15:07:24.543+05:30 true false false true
2 jcr:createdBy String admin true false false true
3 jcr:primaryType Name nt:file true true false true
Name
Type String
Value
Multi
Add


Edit
Preview
Book Title :
Author name :
Image
Description :
Author Component [Root]
Author Details
Author Name
Author Bio
Author Image
Date of Birth
Fetch data from Api? if yes please check the box below
Cancel
Done
Books
Harry Potter
The Great Gatsby
To Kill a Mockingbird
Follow Us


EXAMPLE
Edit
Preview
Infinite Reads
Books
About Me
Reach Me
Search books/author
J.K. ROWLING (JOANNE ROWLING)
British author best known for the Harry Potter series, which became a global phenomenon...
DOB: 10 March, 2025


EXAMPLE
Edit
Preview
which became a global phenomenon...
DOB: 10 March, 2025
The Great Gatsby -
F. Scott Fitzgerald
To Kill a Mockingbird -
Harper Lee
The Lord Of The Rings -
J.R.R. Tolkien
The Catcher In The Rye -
J.D. Salinger
Pride And Prejudice -
Jane Austen


EXAMPLE
Edit
Preview
Pride And Prejudice -
Jane Austen
The Adventures of Tom Sawyer -
Mark Twain
Alice in Wonderland -
Lewis Carroll
War and Peace -
Leo Tolstoy
Don Quixote -
Miguel de Cervantes
Books
About Us
Reach Me
Follow Us

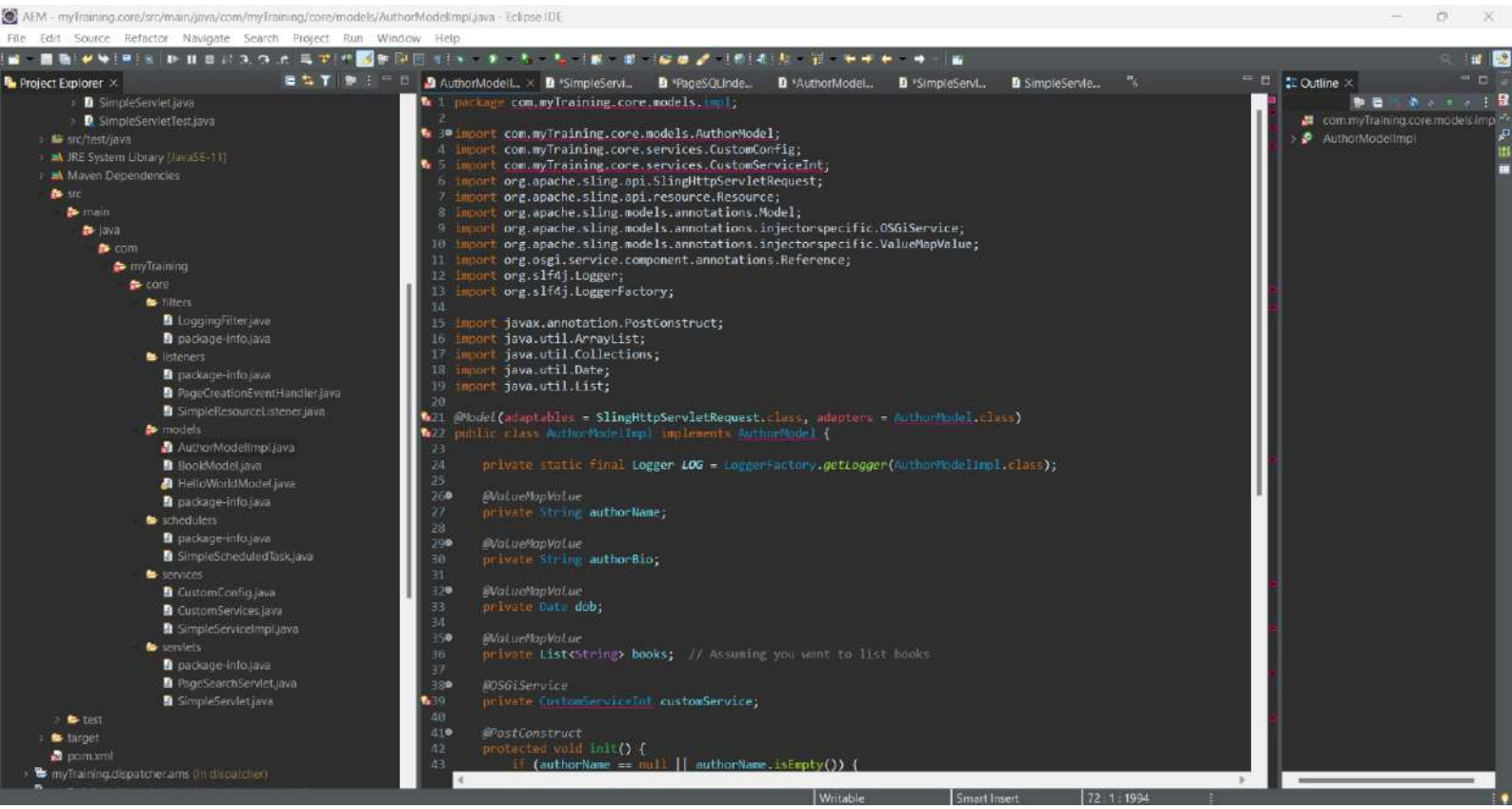

EXAMPLE
Edit
Preview
Pride And Prejudice -
Jane Austen
Load More
Books
Harry Potter
The Great Gatsby
To Kill a Mockingbird
The Lord Of The Rings
The Catcher In The Rye
About Us
Learn more about our passion for books and literature.
Read More
Reach Me
Contact us for book recommendations, queries, or support.
Contact
Follow Us

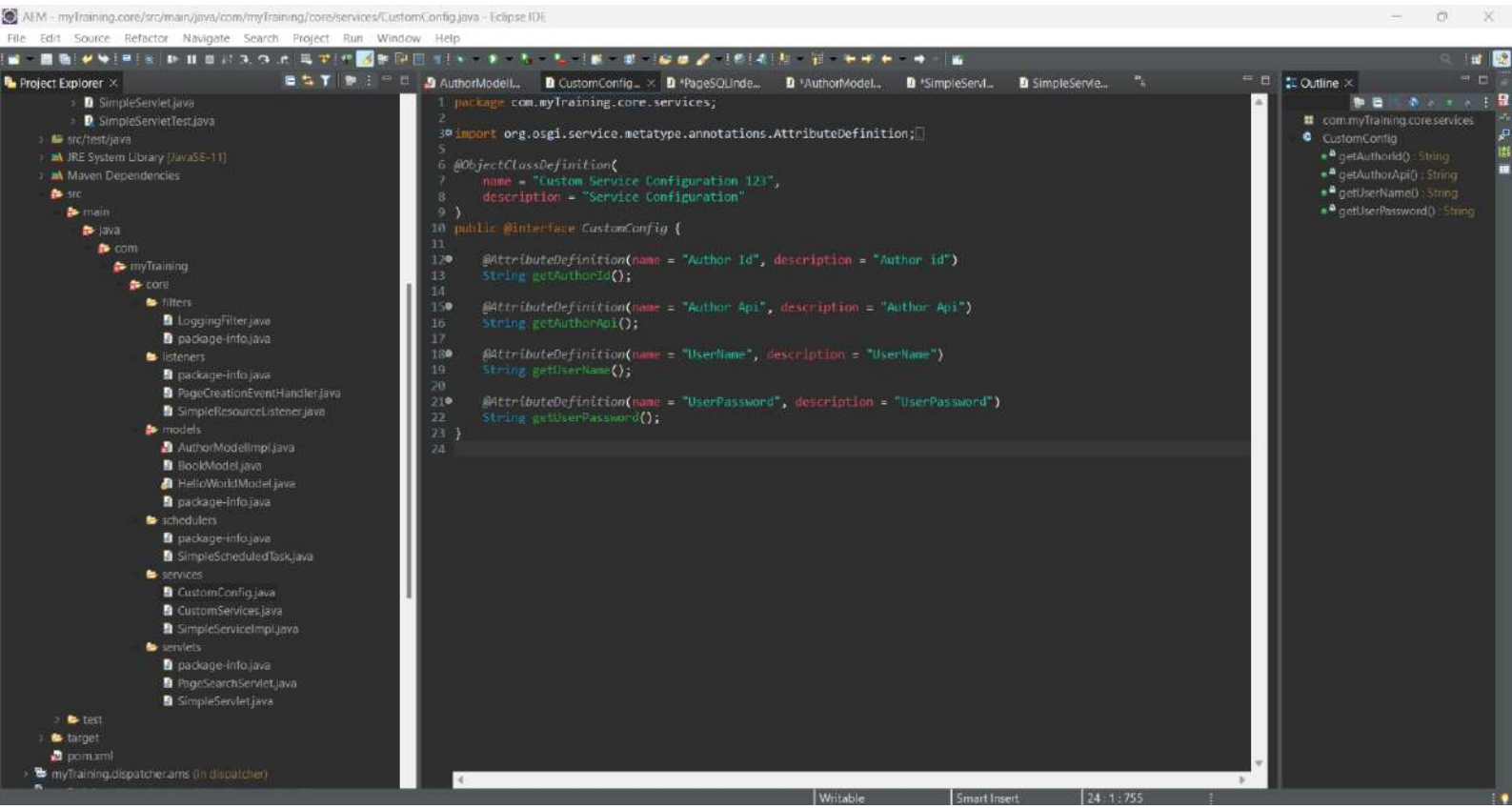
Pretty-print ☑

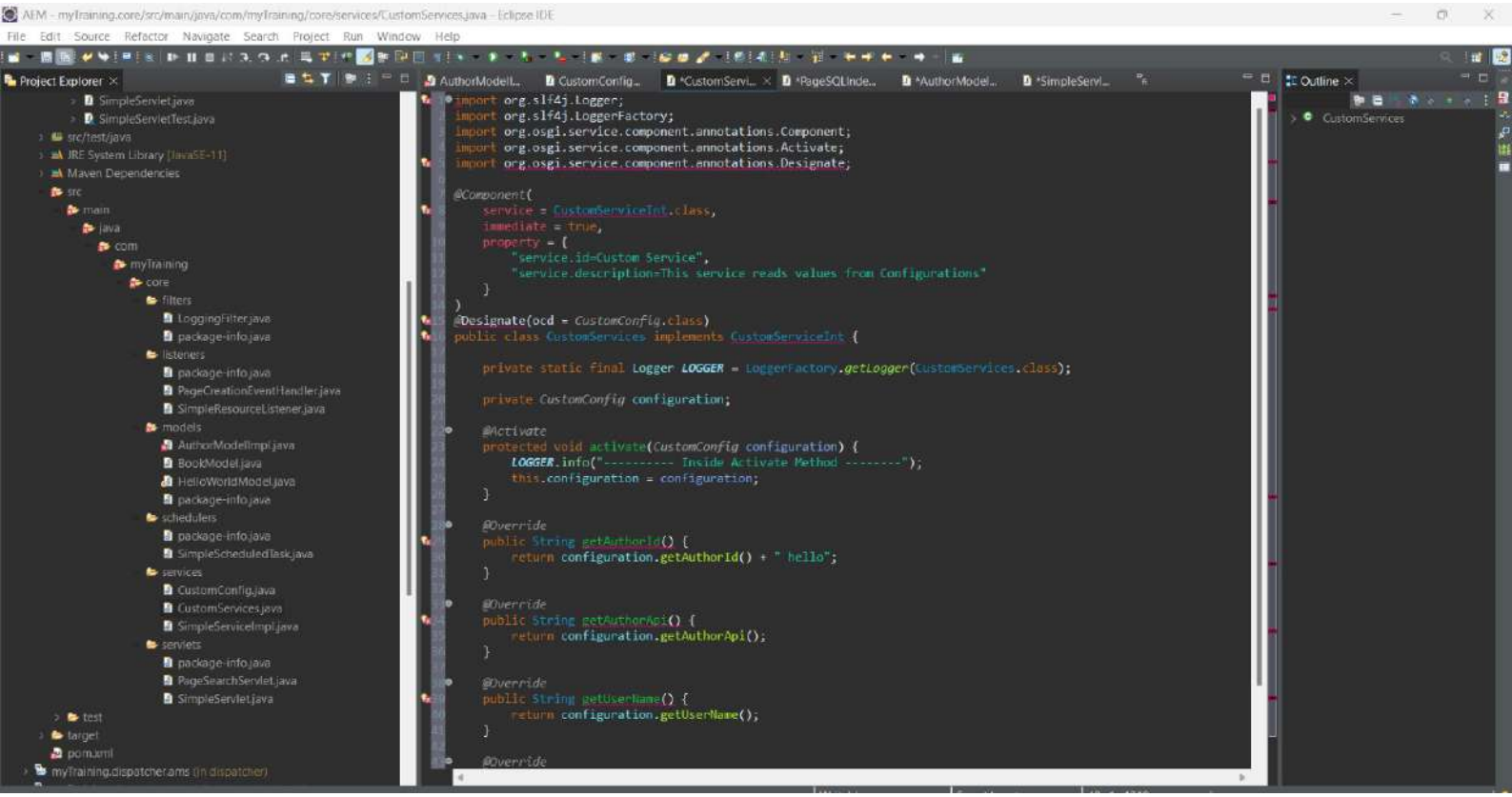

```
    },
    {
      "title": "The Catcher In The Rye",
      "author": "J.D. Salinger"
    },
    {
      "title": "Pride And Prejudice",
      "author": "Jane Austen"
    },
    {
      "title": "The Adventures of Tom Sawyer",
      "author": "Mark Twain"
    },
    {
      "title": "Alice in Wonderland",
      "author": "Lewis Carroll"
    },
    {
      "title": "War and Peace",
      "author": "Leo Tolstoy"
    },
    {
      "title": "Don Quixote",
      "author": "Miguel de Cervantes"
    }
  ]
}
```

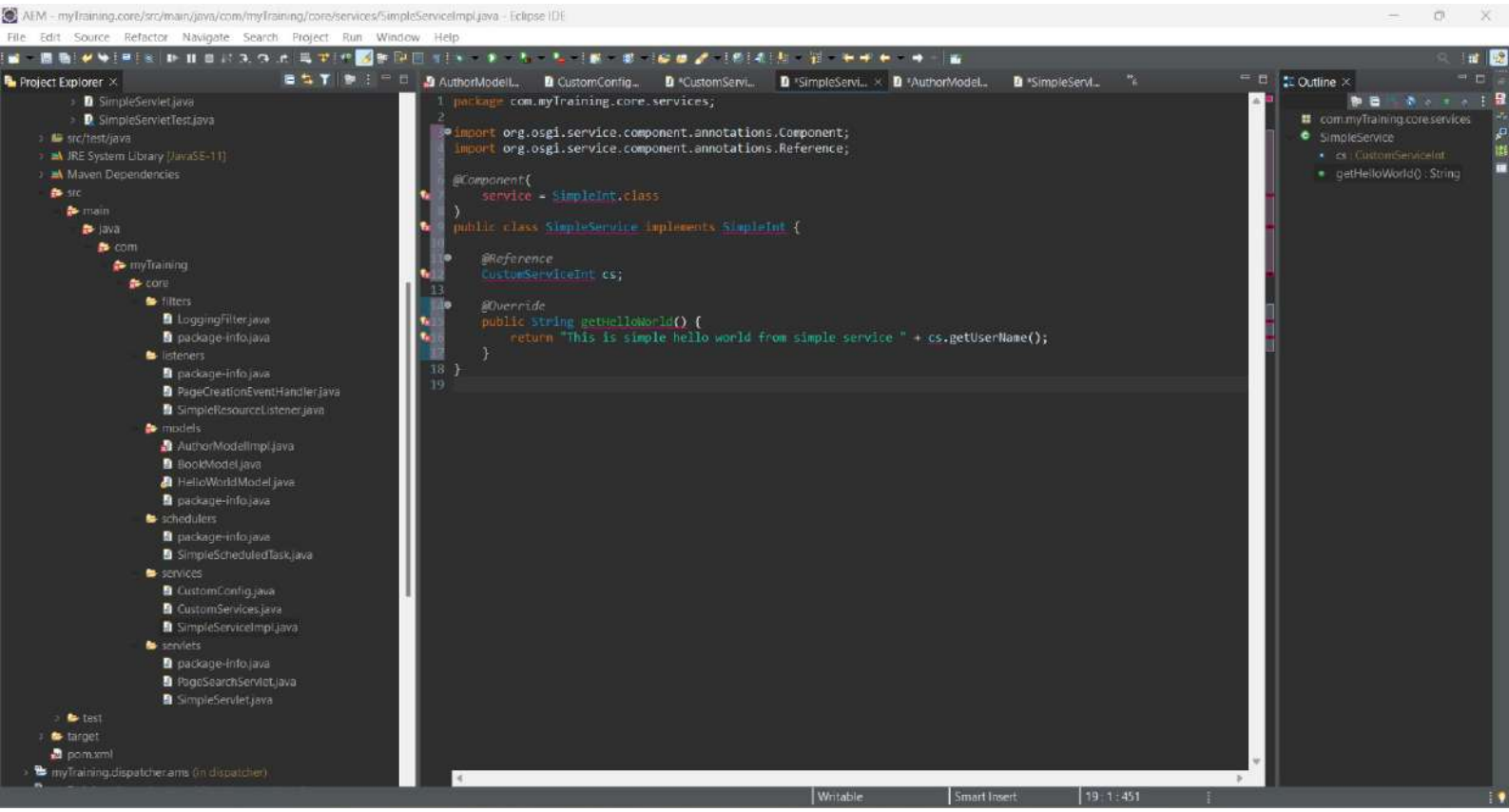
Pretty-print ☑

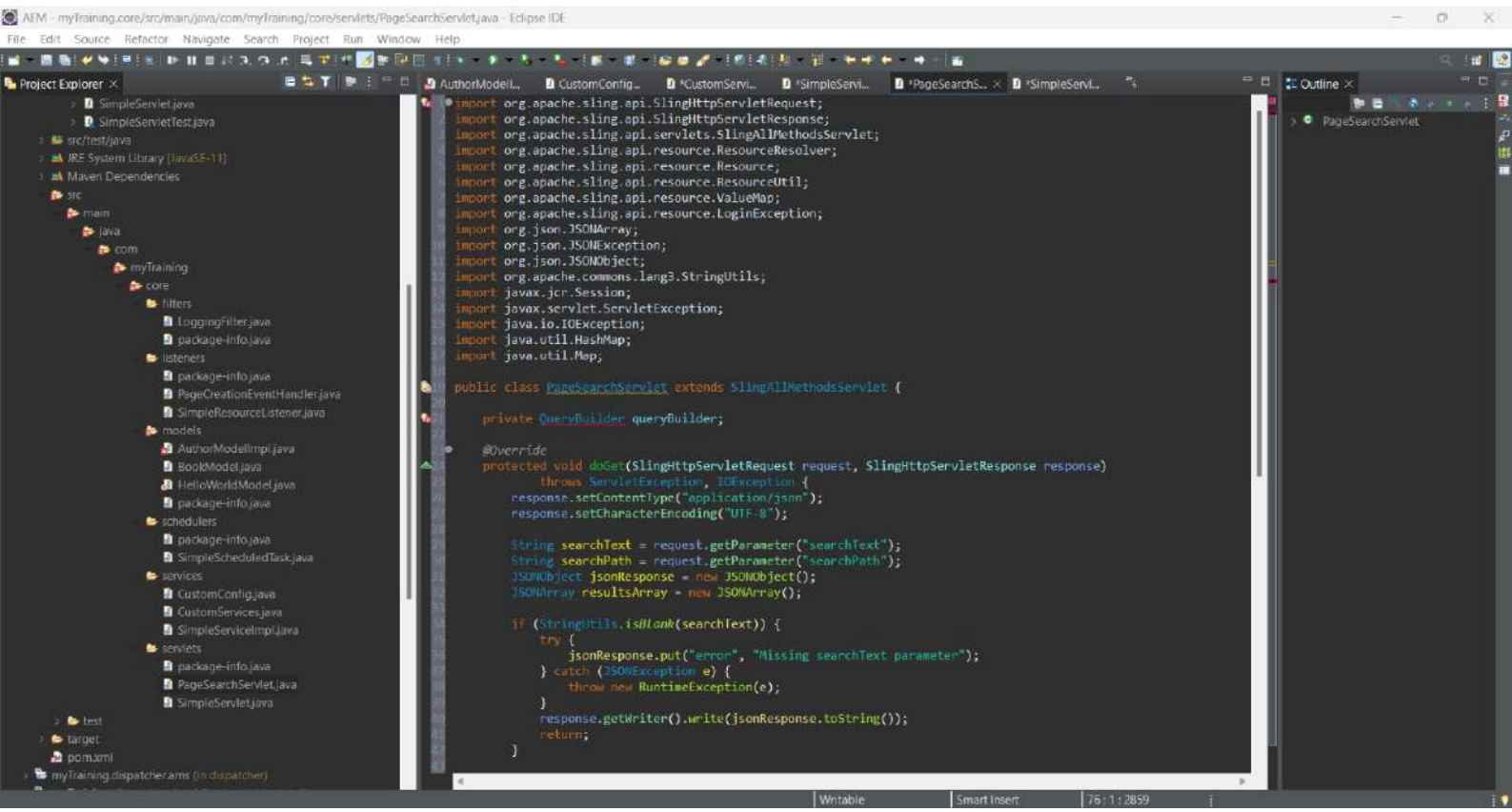

```
{
  "books": [
    {
      "title": "Harry Potter",
      "author": "J.K. Rowling"
    },
    {
      "title": "The Great Gatsby",
      "author": "F. Scott Fitzgerald"
    },
    {
      "title": "To Kill a Mockingbird",
      "author": "Harper Lee"
    },
    {
      "title": "The Lord Of The Rings",
      "author": "J.R.R. Tolkien"
    },
    {
      "title": "The Catcher In The Rye",
      "author": "J.D. Salinger"
    },
    {
      "title": "Pride And Prejudice",
      "author": "Jane Austen"
    },
    {
      "title": "The Adventures of Tom Sawyer",
```

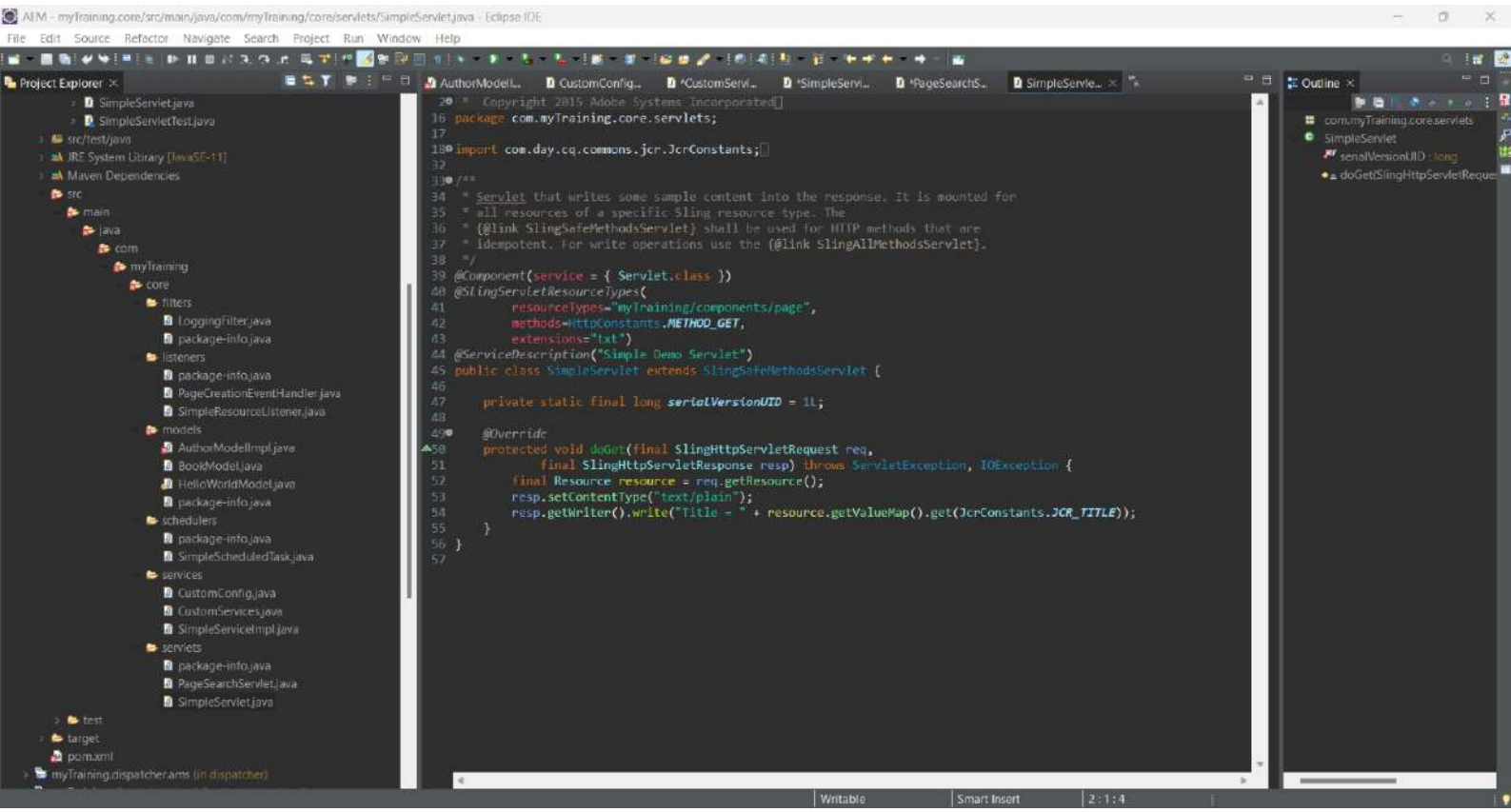

AEM - myTraining.core/src/main/java/com/myTraining/core/models/AuthorModelImpl.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
SimpleResourceListener.java
models
AuthorModelImpl.java
BookModel.java
HelloWorldModel.java
package-info.java
schedulers
package-info.java
SimpleScheduledTask.java
services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
servlets
package-info.java
PageSearchServlet.java
SimpleServlet.java
test
target
pom.xml
myTraining.dispatcher.ams (in dispatcher)
AuthorModelI...
*SimpleServi...
*PageSQLInde...
*AuthorModel...
*SimpleServl...
SimpleServle...
1 package com.myTraining.core.models.impl;
2
3 import com.myTraining.core.models.AuthorModel;
4 import com.myTraining.core.services.CustomConfig;
5 import com.myTraining.core.services.CustomServiceInt;
6 import org.apache.sling.api.SlingHttpServletRequest;
7 import org.apache.sling.api.resource.Resource;
8 import org.apache.sling.models.annotations.Model;
9 import org.apache.sling.models.annotations.injectorspecific.OSGiService;
10 import org.apache.sling.models.annotations.injectorspecific.ValueMapValue;
11 import org.osgi.service.component.annotations.Reference;
12 import org.slf4j.Logger;
13 import org.slf4j.LoggerFactory;
14
15 import javax.annotation.PostConstruct;
16 import java.util.ArrayList;
17 import java.util.Collections;
18 import java.util.Date;
19 import java.util.List;
20
21 @Model(adaptables = SlingHttpServletRequest.class, adapters = AuthorModel.class)
22 public class AuthorModelImpl implements AuthorModel {
23
24 private static final Logger LOG = LoggerFactory.getLogger(AuthorModelImpl.class);
25
26 @ValueMapValue
27 private String authorName;
28
29 @ValueMapValue
30 private String authorBio;
31
32 @ValueMapValue
33 private Date dob;
34
35 @ValueMapValue
36 private List<String> books; // Assuming you want to list books
37
38 @OSGiService
39 private CustomServiceInt customService;
40
41 @PostConstruct
42 protected void init() {
43 if (authorName == null || authorName.isEmpty()) {
Outline
com.myTraining.core.models.impl
AuthorModelImpl
Writable
Smart Insert
72 : 1 : 1994

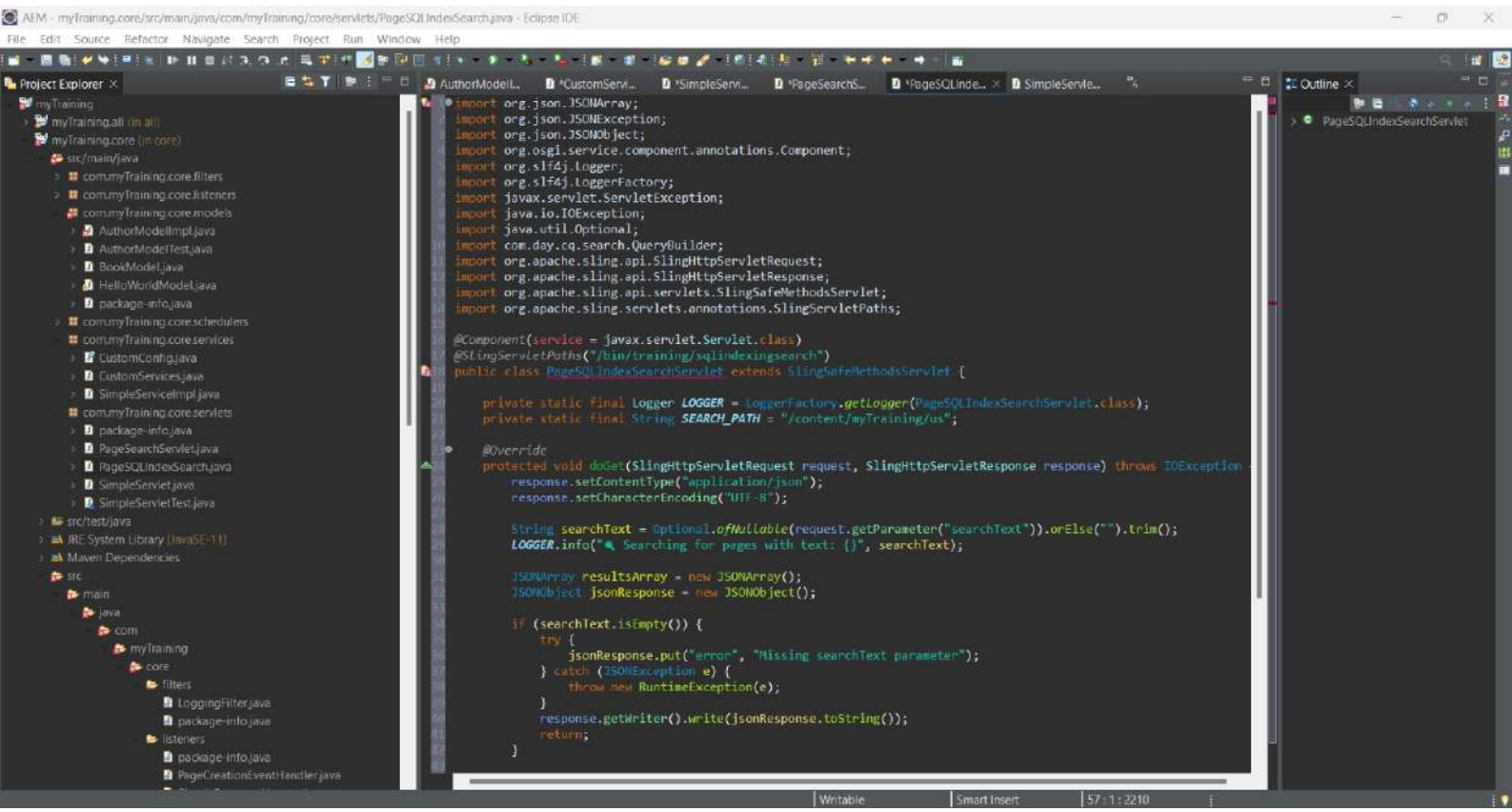

AEM - myTraining.core/src/main/java/com/myTraining/core/services/CustomConfig.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
SimpleResourceListener.java
models
AuthorModelImpl.java
BookModel.java
HelloWorldModel.java
package-info.java
schedulers
package-info.java
SimpleScheduledTask.java
services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
servlets
package-info.java
PageSearchServlet.java
SimpleServlet.java
test
target
pom.xml
myTraining.dispatcher.ams (in dispatcher)
AuthorModelI...
CustomConfig...
*PageSQLInde...
*AuthorModel...
*SimpleServl...
SimpleServle...
1 package com.myTraining.core.services;
2
3 import org.osgi.service.metatype.annotations.AttributeDefinition;
5
6 @ObjectClassDefinition(
7 name = "Custom Service Configuration 123",
8 description = "Service Configuration"
9)
10 public @interface CustomConfig {
11
12 @AttributeDefinition(name = "Author Id", description = "Author id")
13 String getAuthorId();
14
15 @AttributeDefinition(name = "Author Api", description = "Author Api")
16 String getAuthorApi();
17
18 @AttributeDefinition(name = "UserName", description = "UserName")
19 String getUserName();
20
21 @AttributeDefinition(name = "UserPassword", description = "UserPassword")
22 String getUserPassword();
23 }
24
Outline
com.myTraining.core.services
CustomConfig
getAuthorId() : String
getAuthorApi() : String
getUserName() : String
getUserPassword() : String
Writable
Smart Insert
24 : 1 : 755

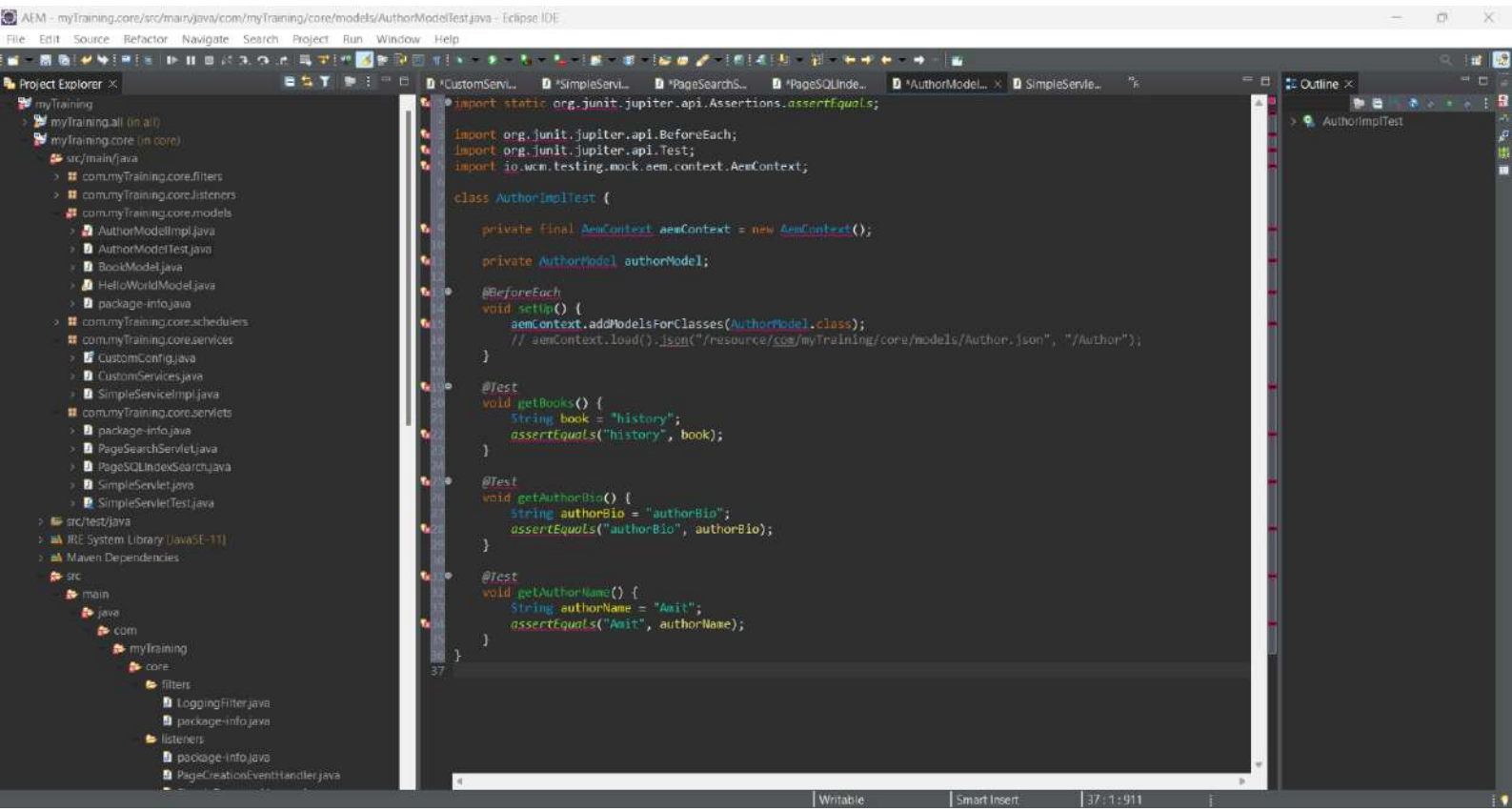

AEM - myTraining.core/src/main/java/com/myTraining/core/services/CustomServices.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
SimpleResourceListener.java
models
AuthorModelImpl.java
BookModel.java
HelloWorldModel.java
package-info.java
schedulers
package-info.java
SimpleScheduledTask.java
services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
servlets
package-info.java
PageSearchServlet.java
SimpleServlet.java
test
target
pom.xml
myTraining.dispatcher.ams (in dispatcher)
AuthorModel...
CustomConfig...
*CustomServi...
*PageSQLInde...
*AuthorModel...
*SimpleServl...
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.osgi.service.component.annotations.Component;
import org.osgi.service.component.annotations.Activate;
import org.osgi.service.component.annotations.Designate;
@Component(
service = CustomServiceInt.class,
immediate = true,
property = {
"service.id=Custom $ervice",
"service.description=This service reads values from Configurations"
}
)
@Designate(ocd = CustomConfig.class)
public class CustomServices implements CustomServiceInt {
private static final Logger LOGGER = LoggerFactory.getLogger(CustomServices.class);
private CustomConfig configuration;
@Activate
protected void activate(CustomConfig configuration) {
LOGGER.info("---------- Inside Activate Method --------");
this.configuration = configuration;
}
@Override
public String getAuthorId() {
return configuration.getAuthorId() + " hello";
}
@Override
public String getAuthorApi() {
return configuration.getAuthorApi();
}
@Override
public String getUserName() {
return configuration.getUserName();
}
@Override
Outline
CustomServices

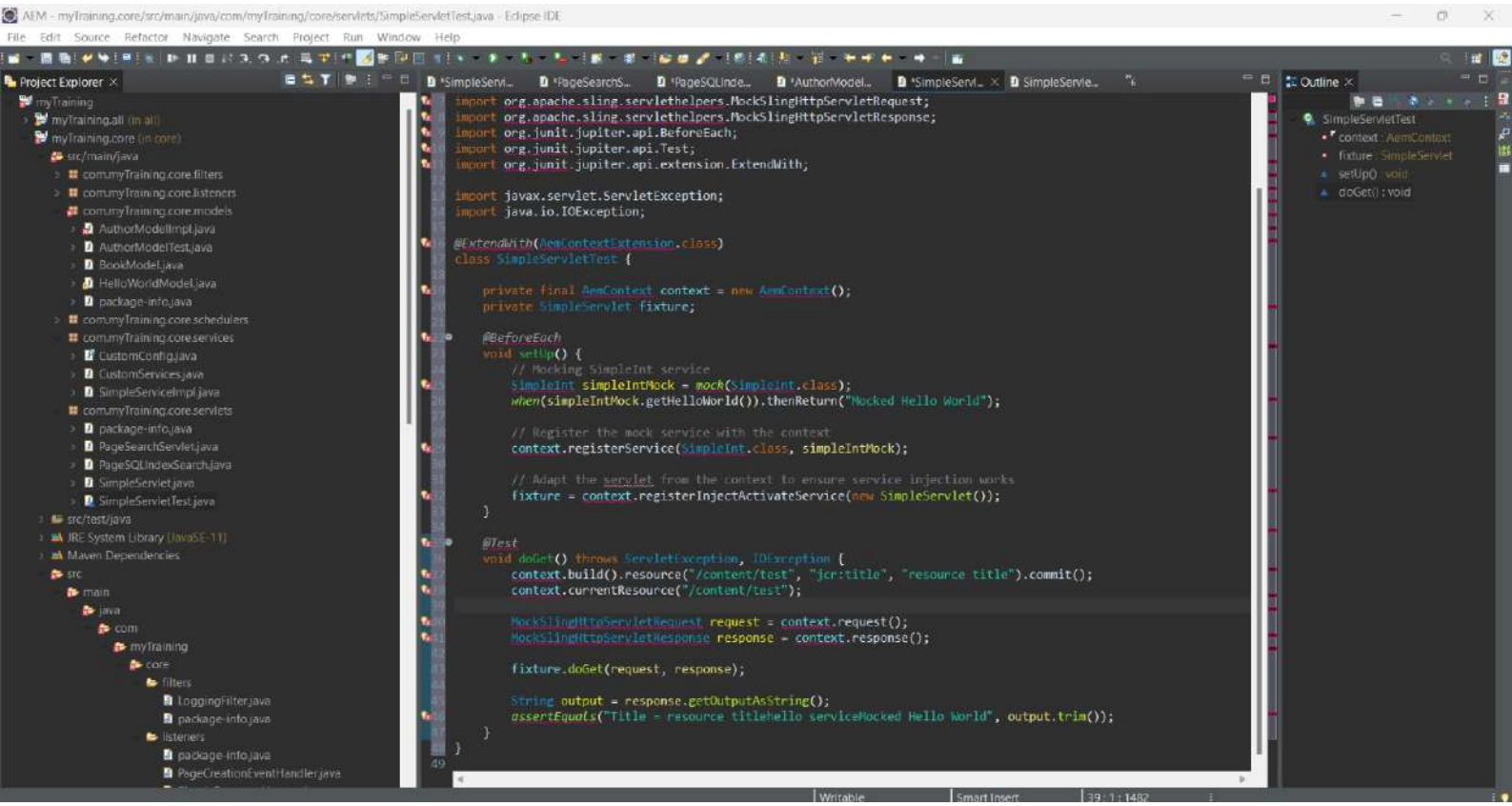

AEM - myTraining.core/src/main/java/com/myTraining/core/services/SimpleServiceImpl.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
SimpleResourceListener.java
models
AuthorModelImpl.java
BookModel.java
HelloWorldModel.java
package-info.java
schedulers
package-info.java
SimpleScheduledTask.java
services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
servlets
package-info.java
PageSearchServlet.java
SimpleServlet.java
test
target
pom.xml
myTraining.dispatcher.ams (in dispatcher)
AuthorModelI...
CustomConfig...
*CustomServi...
*SimpleServi...
*AuthorModel...
*SimpleServl...
1 package com.myTraining.core.services;
2
3 import org.osgi.service.component.annotations.Component;
4 import org.osgi.service.component.annotations.Reference;
5
6 @Component(
7 service = SimpleInt.class
8)
9 public class SimpleService implements SimpleInt {
10
11 @Reference
12 CustomServiceInt cs;
13
14 @Override
15 public String getHelloWorld() {
16 return "This is simple hello world from simple service " + cs.getUserName();
17 }
18 }
19
Outline
com.myTraining.core.services
SimpleService
cs : CustomServiceInt
getHelloWorld() : String
Writable
Smart Insert
19 : 1 : 451


AEM - myTraining.core/src/main/java/com/myTraining/core/servlets/PageSearchServlet.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
SimpleResourceListener.java
models
AuthorModelImpl.java
BookModel.java
HelloWorldModel.java
package-info.java
schedulers
package-info.java
SimpleScheduledTask.java
services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
servlets
package-info.java
PageSearchServlet.java
SimpleServlet.java
test
target
pom.xml
myTraining.dispatcher.ams (in dispatcher)
AuthorModelI...
CustomConfig...
*CustomServi...
*SimpleServi...
*PageSearchS...
*SimpleServl...
import org.apache.sling.api.SlingHttpServletRequest;
import org.apache.sling.api.SlingHttpServletResponse;
import org.apache.sling.api.servlets.SlingAllMethodsServlet;
import org.apache.sling.api.resource.ResourceResolver;
import org.apache.sling.api.resource.Resource;
import org.apache.sling.api.resource.ResourceUtil;
import org.apache.sling.api.resource.ValueMap;
import org.apache.sling.api.resource.LoginException;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.apache.commons.lang3.StringUtils;
import javax.jcr.Session;
import javax.servlet.ServletException;
import java.io.IOException;
import java.util.HashMap;
import java.util.Map;
public class PageSearchServlet extends SlingAllMethodsServlet {
private QueryBuilder queryBuilder;
@Override
protected void doGet(SlingHttpServletRequest request, SlingHttpServletResponse response)
throws ServletException, IOException {
response.setContentType("application/json");
response.setCharacterEncoding("UTF-8");
String searchText = request.getParameter("searchText");
String searchPath = request.getParameter("searchPath");
JSONObject jsonResponse = new JSONObject();
JSONArray resultsArray = new JSONArray();
if (StringUtils.isBlank(searchText)) {
try {
jsonResponse.put("error", "Missing searchText parameter");
} catch (JSONException e) {
throw new RuntimeException(e);
}
response.getWriter().write(jsonResponse.toString());
return;
}
Outline
PageSearchServlet
Writable
Smart Insert
76 : 1 : 2859


AEM - myTraining.core/src/main/java/com/myTraining/core/servlets/SimpleServlet.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
SimpleResourceListener.java
models
AuthorModelImpl.java
BookModel.java
HelloWorldModel.java
package-info.java
schedulers
package-info.java
SimpleScheduledTask.java
services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
servlets
package-info.java
PageSearchServlet.java
SimpleServlet.java
test
target
pom.xml
myTraining.dispatcher.ams (in dispatcher)
AuthorModelI...
CustomConfig...
*CustomServi...
*SimpleServi...
*PageSearchS...
SimpleServle...
* Copyright 2015 Adobe Systems Incorporated
package com.myTraining.core.servlets;
import com.day.cq.commons.jcr.JcrConstants;
/**
* Servlet that writes some sample content into the response. It is mounted for
* all resources of a specific Sling resource type. The
* {@link SlingSafeMethodsServlet} shall be used for HTTP methods that are
* idempotent. For write operations use the {@link SlingAllMethodsServlet}.
*/
@Component(service = { Servlet.class })
@SlingServletResourceTypes(
resourceTypes="myTraining/components/page",
methods=HttpConstants.METHOD_GET,
extensions="txt")
@ServiceDescription("Simple Demo Servlet")
public class SimpleServlet extends SlingSafeMethodsServlet {
private static final long serialVersionUID = 1L;
@Override
protected void doGet(final SlingHttpServletRequest req,
final SlingHttpServletResponse resp) throws ServletException, IOException {
final Resource resource = req.getResource();
resp.setContentType("text/plain");
resp.getWriter().write("Title = " + resource.getValueMap().get(JcrConstants.JCR_TITLE));
}
}
Outline
com.myTraining.core.servlets
SimpleServlet
serialVersionUID : long
doGet(SlingHttpServletReque
Writable
Smart Insert
2 : 1 : 4


AEM - myTraining.core/src/main/java/com/myTraining/core/servlets/PageSQLIndexSearch.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
myTraining
myTraining.all (in all)
myTraining.core (in core)
src/main/java
com.myTraining.core.filters
com.myTraining.core.listeners
com.myTraining.core.models
AuthorModelImpl.java
AuthorModelTest.java
BookModel.java
HelloWorldModel.java
package-info.java
com.myTraining.core.schedulers
com.myTraining.core.services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
com.myTraining.core.servlets
package-info.java
PageSearchServlet.java
PageSQLIndexSearch.java
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
AuthorModelI...
*CustomServi...
*SimpleServi...
*PageSearchS...
*PageSQLInde...
SimpleServle...
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.osgi.service.component.annotations.Component;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import javax.servlet.ServletException;
import java.io.IOException;
import java.util.Optional;
import com.day.cq.search.QueryBuilder;
import org.apache.sling.api.SlingHttpServletRequest;
import org.apache.sling.api.SlingHttpServletResponse;
import org.apache.sling.api.servlets.SlingSafeMethodsServlet;
import org.apache.sling.servlets.annotations.SlingServletPaths;
@Component(service = javax.servlet.Servlet.class)
@SlingServletPaths("/bin/training/sqlindexingsearch")
public class PageSQLIndexSearchServlet extends SlingSafeMethodsServlet {
private static final Logger LOGGER = LoggerFactory.getLogger(PageSQLIndexSearchServlet.class);
private static final String SEARCH_PATH = "/content/myTraining/us";
@Override
protected void doGet(SlingHttpServletRequest request, SlingHttpServletResponse response) throws IOException
response.setContentType("application/json");
response.setCharacterEncoding("UTF-8");
String searchText = Optional.ofNullable(request.getParameter("searchText")).orElse("").trim();
LOGGER.info("🔍 Searching for pages with text: {}", searchText);
JSONArray resultsArray = new JSONArray();
JSONObject jsonResponse = new JSONObject();
if (searchText.isEmpty()) {
try {
jsonResponse.put("error", "Missing searchText parameter");
} catch (JSONException e) {
throw new RuntimeException(e);
}
response.getWriter().write(jsonResponse.toString());
return;
}
Outline
PageSQLIndexSearchServlet
Writable
Smart Insert
57 : 1 : 2210


AEM - myTraining.core/src/main/java/com/myTraining/core/models/AuthorModelTest.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
myTraining
myTraining.all (in all)
myTraining.core (in core)
src/main/java
com.myTraining.core.filters
com.myTraining.core.listeners
com.myTraining.core.models
AuthorModelImpl.java
AuthorModelTest.java
BookModel.java
HelloWorldModel.java
package-info.java
com.myTraining.core.schedulers
com.myTraining.core.services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
com.myTraining.core.servlets
package-info.java
PageSearchServlet.java
PageSQLIndexSearch.java
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
*CustomServi...
*SimpleServi...
*PageSearchS...
*PageSQLInde...
*AuthorModel...
SimpleServle...
import static org.junit.jupiter.api.Assertions.assertEquals;
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.Test;
import io.wcm.testing.mock.aem.context.AemContext;
class AuthorImplTest {
private final AemContext aemContext = new AemContext();
private AuthorModel authorModel;
@BeforeEach
void setUp() {
aemContext.addModelsForClasses(AuthorModel.class);
// aemContext.load().json("/resource/com/myTraining/core/models/Author.json", "/Author");
}
@Test
void getBooks() {
String book = "history";
assertEquals("history", book);
}
@Test
void getAuthorBio() {
String authorBio = "authorBio";
assertEquals("authorBio", authorBio);
}
@Test
void getAuthorName() {
String authorName = "Amit";
assertEquals("Amit", authorName);
}
}
Outline
AuthorImplTest
Writable
Smart Insert
37 : 1 : 911


AEM - myTraining.core/src/main/java/com/myTraining/core/servlets/SimpleServletTest.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
myTraining
myTraining.all (in all)
myTraining.core (in core)
src/main/java
com.myTraining.core.filters
com.myTraining.core.listeners
com.myTraining.core.models
AuthorModelImpl.java
AuthorModel.java
BookModel.java
HelloWorldModel.java
package-info.java
com.myTraining.core.schedulers
com.myTraining.core.services
CustomConfig.java
CustomServices.java
SimpleServiceImpl.java
com.myTraining.core.servlets
package-info.java
PageSearchServlet.java
PageSQLIndexSearch.java
SimpleServlet.java
SimpleServletTest.java
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
main
java
com
myTraining
core
filters
LoggingFilter.java
package-info.java
listeners
package-info.java
PageCreationEventHandler.java
*SimpleServi...
*PageSearchS...
*PageSQLInde...
*AuthorModel...
*SimpleServl...
SimpleServle...
import org.apache.sling.servlethelpers.MockSlingHttpServletRequest;
import org.apache.sling.servlethelpers.MockSlingHttpServletResponse;
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.Test;
import org.junit.jupiter.api.extension.ExtendWith;
import javax.servlet.ServletException;
import java.io.IOException;
@ExtendWith(AemContextExtension.class)
class SimpleServletTest {
private final AemContext context = new AemContext();
private SimpleServlet fixture;
@BeforeEach
void setUp() {
// Mocking SimpleInt service
SimpleInt simpleIntMock = mock(SimpleInt.class);
when(simpleIntMock.getHelloWorld()).thenReturn("Mocked Hello World");
// Register the mock service with the context
context.registerService(SimpleInt.class, simpleIntMock);
// Adapt the servlet from the context to ensure service injection works
fixture = context.registerInjectActivateService(new SimpleServlet());
}
@Test
void doGet() throws ServletException, IOException {
context.build().resource("/content/test", "jcr:title", "resource title").commit();
context.currentResource("/content/test");
MockSlingHttpServletRequest request = context.request();
MockSlingHttpServletResponse response = context.response();
fixture.doGet(request, response);
String output = response.getOutputAsString();
assertEquals("Title - resource titlehello serviceMocked Hello World", output.trim());
}
}
Outline
SimpleServletTest
context : AemContext
fixture : SimpleServlet
setUp() : void
doGet() : void
Writable
Smart Insert
39 : 1 : 1482