

Code Snippets for the HL7 Tutorial

Plugin service tutorial

Code Snippets

```
public CodeSystemCatalogEntry read(NameOrUri nameOrUri, ResolvedReadContext
readContext) {
    //NameOrUri can either be a CodeSystemName or its URI. For this example,
    //assume the request is 'by name'.
    String codeSystemName = nameOrUri.getName();

    if(StringUtils.isBlank(codeSystemName)){
        throw new UnsupportedOperationException("Only resolution by name is
supported.");
    }

    //We can fetch this from any data store we choose
    CodeSystemCatalogEntry entry = new CodeSystemCatalogEntry();
    JDBCConnection con = new JDBCConnection();
    Map<String, String> map = null;
    try {

        map = con.getResult(codeSystemName);
    } catch (SQLException e) {
        e.printStackTrace();
    }

    entry.setCodeSystemName(codeSystemName);
    entry.setAbout(con.getName(map) + " " + codeSystemName);

    EntryDescription ed = new EntryDescription();
    ed.setValue(ModelUtils.toTsAnyType(
        con.getURI(map) + " " + codeSystemName));

    entry.setResourceSynopsis(ed);

    return entry;
}
```

Pom File Snippets

```
<!--This adds the third party jar to the plugin -->
<dependency>
```

```

        <groupId>mysql</groupId>
        <artifactId>mysql-connector-java</artifactId>
        <version>5.1.6</version>
    </dependency>

```

<!--Changes to this plug in configuration allow us to include only those dependencies in the build that we need for the plugin-->

```

<plugin>
    <groupId>org.apache.felix</groupId>
    <artifactId>maven-bundle-plugin</artifactId>
    <extensions>true</extensions>
    <configuration>
        <instructions>
            <Spring-Context>*;timeout:=60</Spring-Context>
            <Import-Package>
                edu.mayo.cts2.framework.service.provider,
                *;resolution:=optional
            </Import-Package>
            <Embed-Dependency>
                mysql-connector-java
            </Embed-Dependency>
        </instructions>
    </configuration>
</plugin>

```

Rest Client Snippets

Get Value Sets

```

public void getValueSets(){
    String uri =
        "http://informatics.mayo.edu/cts2/rest/valuesets";
    URL url;
    try {
        url = new URL(uri);

        HttpURLConnection connection =
            (HttpURLConnection) url.openConnection();
        if (connection.getResponseCode() != 200) {
            throw new RuntimeException("Failed : The HTTP error code is : "
                + connection.getResponseCode());
        }

        BufferedReader br = new BufferedReader(new InputStreamReader(
            (connection.getInputStream())));

        String output;
        System.out.println("Output from Server .... \n");
    }
}

```

```

        while ((output = br.readLine()) != null) {
            System.out.println(output);
        }

        connection.disconnect();

    } catch (MalformedURLException e) {
        e.printStackTrace();
    } catch (ProtocolException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

```

Get Value Set Deserializer

```

    public void getValueSet() {
        String uri =
"http://informatics.mayo.edu/cts2/rest/valuesets?matchvalue=Sequence&format=json";
        URL url;
        try {
            url = new URL(uri);

            HttpURLConnection connection = (HttpURLConnection) url
                .openConnection();
            connection.setRequestProperty("Accept", "text/json");

            if (connection.getResponseCode() != 200) {
                throw new RuntimeException("Failed : The HTTP error code is : "
                    + connection.getResponseCode());
            }

            BufferedReader br = new BufferedReader(new InputStreamReader(
                connection.getInputStream()));
            String output;
            StringBuilder builder = new StringBuilder();

            while ((output = br.readLine()) != null) {
                builder.append(output);
            }
            JsonConverter converter = new JsonConverter();
            ValueSetCatalogEntryDirectory valuesetcat = converter.fromJson(
                builder.toString(), ValueSetCatalogEntryDirectory.class);

            List<ValueSetCatalogEntrySummary> sum = valuesetcat
                .getEntryAsReference();
            for (ValueSetCatalogEntrySummary s : sum) {
                System.out.println(s.getFormalName());
                System.out.println(s.getCurrentDefinition());
                System.out.println(s.getResourceName());
                System.out.println(s.getValueSetName());
                System.out.println(s.getHref());
            }
        }
    }

```

```
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
}
```

Maven dependency

```
<dependency>  
    <groupId>edu.mayo.cts2.framework</groupId>  
    <artifactId>core</artifactId>  
    <version>0.8.0</version>  
</dependency>
```

Maven Repository

```
<repositories>  
    <repository>  
        <id>edu.informatics.maven.release</id>  
        <name>Informatics Maven Release Repository</name>  
        <url>http://informatics.mayo.edu/maven/content/repositories/releases</url>  
    </repository>  
</repositories>
```