

SearchingForConcepts Snippet

Java Code Snippet

```
//Create a Connection to the Grid Service
LexBIGServiceGrid lbs = new LexBIGServiceGridAdapter(gridServiceURL);

//Set up the CodingSchemeIdentification object to define the Coding Scheme</font>
CodingSchemeIdentification csid = new CodingSchemeIdentification();
csid.setName("NCI Thesaurus");

//Get the CodedNodeSet for that CodingScheme (This returns a CodedNodeSet Service Context)
CodedNodeSetGrid cnsgr = lbs.getCodingSchemeConcepts(csid, null);
//getCodingSchemeConcepts is a Grid Service Call

//Set the text to match
MatchCriteria matchText = new MatchCriteria();
matchText.setText("Gene");
//Define a SearchDesignationOption, if any
SearchDesignationOption searchOption = new SearchDesignationOption();

//Choose an algorithm to do the matching
ExtensionIdentification matchAlgorithm = new ExtensionIdentification();
matchAlgorithm.setLexBIGExtensionName("contains");

//Chose a language
LanguageIdentification language = new LanguageIdentification();
language.setIdentifier("en");

//Restrict the CodedNodeSet
cnsgr.restrictToMatchingDesignations(matchText, searchOption, matchAlgorithm, language);
//restrictToMatchingDesignations is a Grid Service Call

//Create a SetResolutionPolicy to handle the details of Resolving the CodedNodeSet
//Here, we will set the Maximum number of Concepts returned to 10.
SetResolutionPolicy resolvePolicy = new SetResolutionPolicy();
resolvePolicy.setMaximumToReturn(10);

//Do the resolve
ResolvedConceptReferenceList rcrl = cnsgr.resolveToList(resolvePolicy);
//resolveToList is a Grid Service Call

//Use the returned ResolvedConceptReferenceList to print some details about the concepts found
ResolvedConceptReference[] rcra = rcrl.getResolvedConceptReference();
for (int i = 0; i < rcra.length; i++) {
    System.out.println(rcra[i].getConceptCode());
    System.out.println(rcra[i].getReferencedEntry().
        getPresentation()[0].getText().getContent());
}
```