Transitive Closure Snippet

Java Code Snippet

```
    public void getTransitiveClosure() {
        //Calculate the transitive closure (all subconcepts) of a given concept
        ResolvedConceptReferencesIterator iterator = null;
        try {
            String codingSchemeName = "NCI Thesaurus";
            String code = "C20181";
            String associationName = "subClassOf";
            boolean resolveForward = false;
            boolean excludeAnonymous = true;
            CodingSchemeVersionOrTag csvt = new CodingSchemeVersionOrTag();
            csvt.setTag("PRODUCTION");
            ConceptReference graphFocus = new ConceptReference();
            graphFocus.setConceptCode(code);
            CodedNodeGraph cng = lbSvc.getNodeGraph(codingSchemeName, csvt, null);
            NameAndValueList asso_list =
                    Constructors.createNameAndValueList(new String[] { associationName }, null);
            cng = cng.restrictToAssociations(asso_list, null);
            boolean resolveBackward = false;
            if (!resolveForward) {
                resolveBackward = true;
            }
            int resolveAssociationDepth = -1;
            int maxReturns = -1;
            CodedNodeSet cns = cng.toNodeList(graphFocus, resolveForward, resolveBackward,
                    resolveAssociationDepth, maxReturns);
            if (excludeAnonymous) {
                CodedNodeSet.AnonymousOption restrictToAnonymous = CodedNodeSet.AnonymousOption.
NON_ANONYMOUS_ONLY;
                cns = cns.restrictToAnonymous(restrictToAnonymous);
            }
            iterator = cns.resolve(null, null, null, null, false);
            while (iterator.hasNext()){
                ResolvedConceptReference rcr = iterator.next();
                String codeReturn = rcr.getCode();
                System.out.println(codeReturn);
            }
        } catch (Exception ex) {
            ex.printStackTrace();
        }
    }
```