package com.twitter.search.common.schema;

import org.apache.lucene.document.Field;

import org.apache.lucene.document.FieldType;

import org.apache.lucene.index.IndexOptions;

/\*\*

\* A Lucene numeric field, similar to the LegacyIntField, LegacyLongField, etc. Lucene classes that

\* were removed in Lucene 7.0.0.

\*/

public final class NumericField extends Field {

private static final FieldType NUMERIC\_FIELD\_TYPE = new FieldType();

static {

NUMERIC\_FIELD\_TYPE.setTokenized(true);

NUMERIC\_FIELD\_TYPE.setOmitNorms(true);

NUMERIC\_FIELD\_TYPE.setIndexOptions(IndexOptions.DOCS);

NUMERIC\_FIELD\_TYPE.freeze();

}

/\*\*

\* Creates a new integer field with the given name and value.

\*/

public static NumericField newIntField(String fieldName, int value) {

NumericField field = new NumericField(fieldName);

field.fieldsData = Integer.valueOf(value);

return field;

}

/\*\*

\* Creates a new long field with the given name and value.

\*/

public static NumericField newLongField(String fieldName, long value) {

NumericField field = new NumericField(fieldName);

field.fieldsData = Long.valueOf(value);

return field;

}

// We could replace the static methods with constructors, but I think that would make it much

// easier to accidentally use NumericField(String, int) instead of NumericField(String, long),

// for example, leading to hard to debug errors.

private NumericField(String fieldName) {

super(fieldName, NUMERIC\_FIELD\_TYPE);

}

}