/\* ----------------------------------------------------------------------------

\* This file was automatically generated by SWIG (http://www.swig.org).

\* Version 4.0.2

\*

\* Do not make changes to this file unless you know what you are doing--modify

\* the SWIG interface file instead.

\* ----------------------------------------------------------------------------- \*/

package com.twitter.ann.faiss;

public class IndexIDMap extends Index {

private transient long swigCPtr;

protected IndexIDMap(long cPtr, boolean cMemoryOwn) {

super(swigfaissJNI.IndexIDMap\_SWIGUpcast(cPtr), cMemoryOwn);

swigCPtr = cPtr;

}

protected static long getCPtr(IndexIDMap obj) {

return (obj == null) ? 0 : obj.swigCPtr;

}

@SuppressWarnings("deprecation")

protected void finalize() {

delete();

}

public synchronized void delete() {

if (swigCPtr != 0) {

if (swigCMemOwn) {

swigCMemOwn = false;

swigfaissJNI.delete\_IndexIDMap(swigCPtr);

}

swigCPtr = 0;

}

super.delete();

}

public void setIndex(Index value) {

swigfaissJNI.IndexIDMap\_index\_set(swigCPtr, this, Index.getCPtr(value), value);

}

public Index getIndex() {

long cPtr = swigfaissJNI.IndexIDMap\_index\_get(swigCPtr, this);

return (cPtr == 0) ? null : new Index(cPtr, false);

}

public void setOwn\_fields(boolean value) {

swigfaissJNI.IndexIDMap\_own\_fields\_set(swigCPtr, this, value);

}

public boolean getOwn\_fields() {

return swigfaissJNI.IndexIDMap\_own\_fields\_get(swigCPtr, this);

}

public void setId\_map(SWIGTYPE\_p\_std\_\_vectorT\_int64\_t\_t value) {

swigfaissJNI.IndexIDMap\_id\_map\_set(swigCPtr, this, SWIGTYPE\_p\_std\_\_vectorT\_int64\_t\_t.getCPtr(value));

}

public SWIGTYPE\_p\_std\_\_vectorT\_int64\_t\_t getId\_map() {

long cPtr = swigfaissJNI.IndexIDMap\_id\_map\_get(swigCPtr, this);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_std\_\_vectorT\_int64\_t\_t(cPtr, false);

}

public IndexIDMap(Index index) {

this(swigfaissJNI.new\_IndexIDMap\_\_SWIG\_0(Index.getCPtr(index), index), true);

}

public void add\_with\_ids(long n, SWIGTYPE\_p\_float x, LongVector xids) {

swigfaissJNI.IndexIDMap\_add\_with\_ids(swigCPtr, this, n, SWIGTYPE\_p\_float.getCPtr(x), SWIGTYPE\_p\_long\_long.getCPtr(xids.data()), xids);

}

public void add(long n, SWIGTYPE\_p\_float x) {

swigfaissJNI.IndexIDMap\_add(swigCPtr, this, n, SWIGTYPE\_p\_float.getCPtr(x));

}

public void search(long n, SWIGTYPE\_p\_float x, long k, SWIGTYPE\_p\_float distances, LongVector labels) {

swigfaissJNI.IndexIDMap\_search(swigCPtr, this, n, SWIGTYPE\_p\_float.getCPtr(x), k, SWIGTYPE\_p\_float.getCPtr(distances), SWIGTYPE\_p\_long\_long.getCPtr(labels.data()), labels);

}

public void train(long n, SWIGTYPE\_p\_float x) {

swigfaissJNI.IndexIDMap\_train(swigCPtr, this, n, SWIGTYPE\_p\_float.getCPtr(x));

}

public void reset() {

swigfaissJNI.IndexIDMap\_reset(swigCPtr, this);

}

public long remove\_ids(IDSelector sel) {

return swigfaissJNI.IndexIDMap\_remove\_ids(swigCPtr, this, IDSelector.getCPtr(sel), sel);

}

public void range\_search(long n, SWIGTYPE\_p\_float x, float radius, RangeSearchResult result) {

swigfaissJNI.IndexIDMap\_range\_search(swigCPtr, this, n, SWIGTYPE\_p\_float.getCPtr(x), radius, RangeSearchResult.getCPtr(result), result);

}

public IndexIDMap() {

this(swigfaissJNI.new\_IndexIDMap\_\_SWIG\_1(), true);

}

}