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

\* 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 MultiIndexQuantizer2 extends MultiIndexQuantizer {

private transient long swigCPtr;

protected MultiIndexQuantizer2(long cPtr, boolean cMemoryOwn) {

super(swigfaissJNI.MultiIndexQuantizer2\_SWIGUpcast(cPtr), cMemoryOwn);

swigCPtr = cPtr;

}

protected static long getCPtr(MultiIndexQuantizer2 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\_MultiIndexQuantizer2(swigCPtr);

}

swigCPtr = 0;

}

super.delete();

}

public void setAssign\_indexes(SWIGTYPE\_p\_std\_\_vectorT\_faiss\_\_Index\_p\_t value) {

swigfaissJNI.MultiIndexQuantizer2\_assign\_indexes\_set(swigCPtr, this, SWIGTYPE\_p\_std\_\_vectorT\_faiss\_\_Index\_p\_t.getCPtr(value));

}

public SWIGTYPE\_p\_std\_\_vectorT\_faiss\_\_Index\_p\_t getAssign\_indexes() {

long cPtr = swigfaissJNI.MultiIndexQuantizer2\_assign\_indexes\_get(swigCPtr, this);

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

}

public void setOwn\_fields(boolean value) {

swigfaissJNI.MultiIndexQuantizer2\_own\_fields\_set(swigCPtr, this, value);

}

public boolean getOwn\_fields() {

return swigfaissJNI.MultiIndexQuantizer2\_own\_fields\_get(swigCPtr, this);

}

public MultiIndexQuantizer2(int d, long M, long nbits, SWIGTYPE\_p\_p\_faiss\_\_Index indexes) {

this(swigfaissJNI.new\_MultiIndexQuantizer2\_\_SWIG\_0(d, M, nbits, SWIGTYPE\_p\_p\_faiss\_\_Index.getCPtr(indexes)), true);

}

public MultiIndexQuantizer2(int d, long nbits, Index assign\_index\_0, Index assign\_index\_1) {

this(swigfaissJNI.new\_MultiIndexQuantizer2\_\_SWIG\_1(d, nbits, Index.getCPtr(assign\_index\_0), assign\_index\_0, Index.getCPtr(assign\_index\_1), assign\_index\_1), true);

}

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

swigfaissJNI.MultiIndexQuantizer2\_train(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.MultiIndexQuantizer2\_search(swigCPtr, this, n, SWIGTYPE\_p\_float.getCPtr(x), k, SWIGTYPE\_p\_float.getCPtr(distances), SWIGTYPE\_p\_long\_long.getCPtr(labels.data()), labels);

}

}