Countries Of The World App 1.5 - Java

CS 3310 – Dr. Donna Kaminski

Index

[Log.txt 2](#_Toc401649394)

[CountryData.bin 4](#_Toc401649395)

[Setup 5](#_Toc401649396)

[UserApp 6](#_Toc401649397)

[DataTable 7](#_Toc401649398)

[DataTableRecord 11](#_Toc401649399)

[Log 15](#_Toc401649400)

[PrettyPrintUtility (modified with approval) 16](#_Toc401649401)

# FILE STATUS > Log FILE opened

CODE STATUS > Setup started

FILE STATUS > RawData FILE opened (A3RawDataSample.csv)

FILE STATUS > CountryData FILE opened

FILE STATUS > RawData FILE closed

CODE STATUS > Setup finished - 26 countries processed

FILE STATUS > Log FILE closed

STATUS > Log FILE opened

STATUS > CountryData FILE opened

DATA STORAGE

MAX\_N\_HOME\_LOC: 20 | nHome: 11 | nColl: 15

LOC/ CDE ID- NAME----------- CONTINENT---- ------AREA ---POPULATION LIFE LINK

001/ ...

002/ ...

003/ ...

004/ MEX 012 Mexico North America 1,958,201 98,881,000 71.5 030

005/ ...

006/ ...

007/ ...

008/ LBN 060 Lebanon Asia 10,400 3,282,000 71.3 -01

009/ SOM 069 Somalia Africa 637,657 10,097,000 46.2 -01

010/ KEN 102 Kenya Africa 580,367 30,080,000 48.0 033

011/ EGY 030 Egypt Africa 1,001,449 68,470,000 63.3 -01

012/ CHN 023 China Asia 9,572,900 1,277,558,000 71.4 027

013/ ...

014/ OMN 042 Oman Asia 309,500 2,542,000 71.8 -01

015/ USA 006 United States North America 9,363,520 278,357,000 77.1 -01

016/ ...

017/ YEM 090 Yemen Asia 527,968 18,112,000 59.8 029

018/ NIC 010 Nicaragua North America 130,000 5,074,000 68.7 -01

019/ ...

020/ JPN 013 Japan Asia 377,829 126,714,000 80.7 035

021/ FRA 019 France Europe 551,500 59,225,700 78.8 -01

022/ BEL 027 Belgium Europe 30,518 10,239,000 77.8 -01

023/ GBR 032 United Kingdom Europe 242,900 59,623,400 77.7 -01

024/ HTI 039 Haiti North America 27,750 8,222,000 49.2 022

025/ VEN 043 Venezuela South America 912,050 24,170,000 73.1 023

026/ POL 020 Poland Europe 323,250 38,653,600 73.2 021

027/ IND 072 India Asia 3,287,263 1,013,662,000 62.5 025

028/ DEU 053 Germany Europe 357,022 82,164,700 77.4 026

029/ WSM 037 Samoa Oceania 2,831 180,000 69.2 -01

030/ XVI 001 Hex Land North America 1 1,966,514,816 0.0 024

031/ QAT 035 Qatar Asia 11,000 599,000 72.4 028

032/ ZWE 132 Zimbabwe Africa 390,757 11,669,000 37.8 -01

033/ RUS 120 Russian Federat Europe 17,075,400 146,934,000 67.2 032

034/ TCA 029 Turks and Caico North America 430 17,000 73.3 031

035/ ATA 059 Antarctica Antarctica 13,120,000 0 0.0 034

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

STATUS > CountryData FILE closed

STATUS > Log FILE closed

FILE STATUS > Log FILE opened

CODE STATUS > UserApp started

FILE STATUS > TransData FILE opened (A3TransData1.txt)

FILE STATUS > CountryData FILE opened

SC MEX

MEX 012 Mexico North America 1,958,201 98,881,000 71.5

SC JPN

JPN 013 Japan Asia 377,829 126,714,000 80.7

SC FRA

FRA 019 France Europe 551,500 59,225,700 78.8

SC ATA

ATA 059 Antarctica Antarctica 13,120,000 0 0.0

SC LBN

LBN 060 Lebanon Asia 10,400 3,282,000 71.3

SC WSM

WSM 037 Samoa Oceania 2,831 180,000 69.2

SC NPJ

SORRY, invalid code

SC NLB

SORRY, invalid code

SC EMY

SORRY, invalid code

SC QQQ

SORRY, invalid code

SC YEM

YEM 090 Yemen Asia 527,968 18,112,000 59.8

SC IQQ

SORRY, invalid code

DC USA

[SORRY, Select By Code module not yet working]

DC FRA

[SORRY, Select By Code module not yet working]

DC WMU

[SORRY, Select By Code module not yet working]

SC JOR

SORRY, invalid code

SC GRC

SORRY, invalid code

SC MYS

SORRY, invalid code

SC QAT

QAT 035 Qatar Asia 11,000 599,000 72.4

SC ATA

ATA 059 Antarctica Antarctica 13,120,000 0 0.0

SC WSM

WSM 037 Samoa Oceania 2,831 180,000 69.2

SC BEL

BEL 027 Belgium Europe 30,518 10,239,000 77.8

FILE STATUS > TransData FILE closed

CODE STATUS > UserApp finished - 22 transactions processed

FILE STATUS > Log FILE closed

# Offset(d) 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37

00000000 00 14 00 0B 00 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ......................................

00000038 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ......................................

00000076 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ......................................

00000114 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ......................................

00000152 00 00 00 00 00 00 00 4D 45 58 00 0C 4D 65 78 69 63 6F 20 20 20 20 20 20 20 20 20 4E 6F 72 74 68 20 41 6D 65 72 69 .......MEX..Mexico North Ameri

00000190 63 61 00 1D E1 39 00 00 00 00 05 E4 CD E8 42 8F 00 00 00 1E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ca..á9.....äÍèB.......................

00000228 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ......................................

00000266 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ......................................

00000304 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ......................................

00000342 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 4C 42 4E 00 3C 4C 65 62 61 6E 6F 6E 20 20 20 20 20 .....................LBN.<Lebanon

00000380 20 20 20 41 73 69 61 20 20 20 20 20 20 20 20 20 00 00 28 A0 00 00 00 00 00 32 14 50 42 8E 99 9A FF FF 53 4F 4D 00 Asia ..( .....2.PBŽ™šÿÿSOM.

00000418 45 53 6F 6D 61 6C 69 61 20 20 20 20 20 20 20 20 41 66 72 69 63 61 20 20 20 20 20 20 20 00 09 BA D9 00 00 00 00 00 ESomalia Africa ..ºÙ.....

00000456 9A 11 68 42 38 CC CD FF FF 4B 45 4E 00 66 4B 65 6E 79 61 20 20 20 20 20 20 20 20 20 20 41 66 72 69 63 61 20 20 20 š.hB8ÌÍÿÿKEN.fKenya Africa

00000494 20 20 20 20 00 08 DB 0F 00 00 00 00 01 CA FC 00 42 40 00 00 00 21 45 47 59 00 1E 45 67 79 70 74 20 20 20 20 20 20 ..Û......Êü.B@...!EGY..Egypt

00000532 20 20 20 20 41 66 72 69 63 61 20 20 20 20 20 20 20 00 0F 47 E9 00 00 00 00 04 14 C4 F0 42 7D 33 33 FF FF 43 48 4E Africa ..Gé......ÄðB}33ÿÿCHN

00000570 00 17 43 68 69 6E 61 20 20 20 20 20 20 20 20 20 20 41 73 69 61 20 20 20 20 20 20 20 20 20 00 92 12 24 00 00 00 00 ..China Asia .’.$....

00000608 4C 25 FC F0 42 8E CC CD 00 1B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 L%üðBŽÌÍ..............................

00000646 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 4F 4D 4E 00 2A 4F 6D 61 6E 20 20 20 20 20 20 .......................OMN.\*Oman

00000684 20 20 20 20 20 41 73 69 61 20 20 20 20 20 20 20 20 20 00 04 B8 FC 00 00 00 00 00 26 C9 B0 42 8F 99 9A FF FF 55 53 Asia ..¸ü.....&É°B.™šÿÿUS

00000722 41 00 06 55 6E 69 74 65 64 20 53 74 61 74 65 73 20 20 4E 6F 72 74 68 20 41 6D 65 72 69 63 61 00 8E E0 40 00 00 00 A..United States North America.Žà@...

00000760 00 10 97 64 08 42 9A 33 33 FF FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..—d.Bš33ÿÿ...........................

00000798 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 59 45 4D 00 5A 59 65 6D 65 6E 20 20 20 20 ........................YEM.ZYemen

00000836 20 20 20 20 20 20 41 73 69 61 20 20 20 20 20 20 20 20 20 00 08 0E 60 00 00 00 00 01 14 5E 00 42 6F 33 33 00 1D 4E Asia ...`......^.Bo33..N

00000874 49 43 00 0A 4E 69 63 61 72 61 67 75 61 20 20 20 20 20 20 4E 6F 72 74 68 20 41 6D 65 72 69 63 61 00 01 FB D0 00 00 IC..Nicaragua North America..ûÐ..

00000912 00 00 00 4D 6C 50 42 89 66 66 FF FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...MlPB‰ffÿÿ..........................

00000950 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 4A 50 4E 00 0D 4A 61 70 61 6E 20 20 20 .........................JPN..Japan

00000988 20 20 20 20 20 20 20 41 73 69 61 20 20 20 20 20 20 20 20 20 00 05 C3 E5 00 00 00 00 07 8D 80 90 42 A1 66 66 00 23 Asia ..Ãå......€.B¡ff.#

00001026 46 52 41 00 13 46 72 61 6E 63 65 20 20 20 20 20 20 20 20 20 45 75 72 6F 70 65 20 20 20 20 20 20 20 00 08 6A 4C 00 FRA..France Europe ..jL.

00001064 00 00 00 03 87 B6 64 42 9D 99 9A FF FF 42 45 4C 00 1B 42 65 6C 67 69 75 6D 20 20 20 20 20 20 20 20 45 75 72 6F 70 ....‡¶dB.™šÿÿBEL..Belgium Europ

00001102 65 20 20 20 20 20 20 20 00 00 77 36 00 00 00 00 00 9C 3C 18 42 9B 99 9A FF FF 47 42 52 00 20 55 6E 69 74 65 64 20 e ..w6.....œ<.B›™šÿÿGBR. United

00001140 4B 69 6E 67 64 6F 6D 20 45 75 72 6F 70 65 20 20 20 20 20 20 20 00 03 B4 D4 00 00 00 00 03 8D C7 E8 42 9B 66 66 FF Kingdom Europe ..´Ô......ÇèB›ffÿ

00001178 FF 48 54 49 00 27 48 61 69 74 69 20 20 20 20 20 20 20 20 20 20 4E 6F 72 74 68 20 41 6D 65 72 69 63 61 00 00 6C 66 ÿHTI.'Haiti North America..lf

00001216 00 00 00 00 00 7D 75 30 42 44 CC CD 00 16 56 45 4E 00 2B 56 65 6E 65 7A 75 65 6C 61 20 20 20 20 20 20 53 6F 75 74 .....}u0BDÌÍ..VEN.+Venezuela Sout

00001254 68 20 41 6D 65 72 69 63 61 00 0D EA B2 00 00 00 00 01 70 CE 10 42 92 33 33 00 17 50 4F 4C 00 14 50 6F 6C 61 6E 64 h America..ê².....pÎ.B’33..POL..Poland

00001292 20 20 20 20 20 20 20 20 20 45 75 72 6F 70 65 20 20 20 20 20 20 20 00 04 EE B2 00 00 00 00 02 4D CE A0 42 92 66 66 Europe ..î².....MÎ B’ff

00001330 00 15 49 4E 44 00 48 49 6E 64 69 61 20 20 20 20 20 20 20 20 20 20 41 73 69 61 20 20 20 20 20 20 20 20 20 00 32 28 ..IND.HIndia Asia .2(

00001368 DF 00 00 00 00 3C 6B 41 30 42 7A 00 00 00 19 44 45 55 00 35 47 65 72 6D 61 6E 79 20 20 20 20 20 20 20 20 45 75 72 ß....<kA0Bz....DEU.5Germany Eur

00001406 6F 70 65 20 20 20 20 20 20 20 00 05 72 9E 00 00 00 00 04 E5 BB DC 42 9A CC CD 00 1A 57 53 4D 00 25 53 61 6D 6F 61 ope ..rž.....å»ÜBšÌÍ..WSM.%Samoa

00001444 20 20 20 20 20 20 20 20 20 20 4F 63 65 61 6E 69 61 20 20 20 20 20 20 00 00 0B 0F 00 00 00 00 00 02 BF 20 42 8A 66 Oceania ..........¿ BŠf

00001482 66 FF FF 58 56 49 00 01 48 65 78 20 4C 61 6E 64 20 20 20 20 20 20 20 4E 6F 72 74 68 20 41 6D 65 72 69 63 61 00 00 fÿÿXVI..Hex Land North America..

00001520 00 01 00 00 00 00 75 36 A2 80 00 00 00 00 00 18 51 41 54 00 23 51 61 74 61 72 20 20 20 20 20 20 20 20 20 20 41 73 ......u6¢€......QAT.#Qatar As

00001558 69 61 20 20 20 20 20 20 20 20 20 00 00 2A F8 00 00 00 00 00 09 23 D8 42 90 CC CD 00 1C 5A 57 45 00 84 5A 69 6D 62 ia ..\*ø......#ØB.ÌÍ..ZWE.„Zimb

00001596 61 62 77 65 20 20 20 20 20 20 20 41 66 72 69 63 61 20 20 20 20 20 20 20 00 05 F6 65 00 00 00 00 00 B2 0E 08 42 17 abwe Africa ..öe.....²..B.

00001634 33 33 FF FF 52 55 53 00 78 52 75 73 73 69 61 6E 20 46 65 64 65 72 61 74 45 75 72 6F 70 65 20 20 20 20 20 20 20 01 33ÿÿRUS.xRussian FederatEurope .

00001672 04 8C C8 00 00 00 00 08 C2 08 F0 42 86 66 66 00 20 54 43 41 00 1D 54 75 72 6B 73 20 61 6E 64 20 43 61 69 63 6F 4E .ŒÈ.....Â.ðB†ff. TCA..Turks and CaicoN

00001710 6F 72 74 68 20 41 6D 65 72 69 63 61 00 00 01 AE 00 00 00 00 00 00 42 68 42 92 99 9A 00 1F 41 54 41 00 3B 41 6E 74 orth America...®......BhB’™š..ATA.;Ant

00001748 61 72 63 74 69 63 61 20 20 20 20 20 41 6E 74 61 72 63 74 69 63 61 20 20 20 00 C8 32 00 00 00 00 00 00 00 00 00 00 arctica Antarctica .È2..........

00001786 00 00 00 00 22 ...."

**package** edu.wmich.cs3310.a3;

**import** java.io.IOException;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Class containing method to setup CDT from the RawData files

\* Countries Of The World App 1.5

\* **@author** Caleb Viola

\*/

# **public** **class** Setup {

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Setup DataTable and NameIndex from RawData files

\* **@param** fileNameSufix

\* **@throws** IOException

\*/

**public** **static** **void** **main** (String fileNameSufix) **throws** IOException {

Log log = **new** Log(**true**);

log.displayThis("CODE STATUS > Setup started");

RawData rd = **new** RawData(fileNameSufix, log);

DataTable dt = **new** DataTable("CountryData.bin", log);

**int** N = 0;

rd.grabCountry();

**while** (!rd.doneWithInput){

**if**(rd.countryGrabbed){

dt.insertCountry(rd.getCode(), rd.getId(), rd.getName(),

rd.getContinent(), rd.getArea(), rd.getPopulation(),

rd.getLifeExpectancy(), log);

}

rd.grabCountry();

}

N = rd.getTransactions();

rd.finishUp(log);

dt.finishUp();

log.displayThis("CODE STATUS > Setup finished - "+N+" countries processed");

log.finishUp();

}

}

**package** edu.wmich.cs3310.a3;

**import** java.io.\*;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Interact with DataTable from TransData files

\* Countries Of The World App 1.5

\* **@author** Caleb Viola

\*/

# **public** **class** UserApp {

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* **@param** fileNameSufix

\* **@throws** IOException

\*/

**public** **static** **void** **main**(**int** fileNameSufix) **throws** IOException{

Log log = **new** Log(**true**);

log.displayThis("CODE STATUS > UserApp started");

UI ui = **new** UI(fileNameSufix, log);

DataTable dt = **new** DataTable("CountryData.bin", log);

**int** N = 0;

ui.grabCommand();

**while** (!ui.doneWithTrans){

**switch** (ui.getTransCode()) {

**case** "IN":

log.displayThis("IN " + ui.getCode() + "," + ui.getId() + ","

+ ui.getName() + "," + ui.getArea() + ","

+ ui.getPopulation() + "," + ui.getLifeExpectancy());

dt.insertCountry(ui.getCode(), ui.getId(), ui.getName(),

ui.getContinent(), ui.getArea(), ui.getPopulation(),

ui.getLifeExpectancy(), log);

log.displayThis(" OK, country inserted");

**break**;

**case** "DC":

log.displayThis("DC " + ui.getCode());

dt.deleteByCode(ui.getCode(), log);

**break**;

**case** "SC":

log.displayThis("SC " + ui.getCode());

dt.selectByCode(ui.getCode(), log);

**break**;

**default**:

log.displayThis(ui.getTransCode() + "\n"

+ " ERROR, invalid command");

**break**;

}

ui.grabCommand();

}

N = ui.getTransactions();

ui.finishUp(log);

dt.finishUp();

log.displayThis("CODE STATUS > UserApp finished - " + N

+ " transactions processed");

log.finishUp();

}

}

**package** edu.wmich.cs3310.a3;

**import** java.io.\*;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Countries Of The World App 1.5

\* **@author** Caleb Viola

\*/

# **public** **class** DataTable {

**private** **short** MAX\_N\_HOME\_LOC;

**private** **short** nHome;

**private** **short** nColl;

**private** **int** homeAddress;

**private** RandomAccessFile file;

**private** DataTableRecord dtr;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Constructor

\* **@param** fileName

\* **@param** log

\* **@throws** IOException

\*/

**public** **DataTable**(String fileName, Log log) **throws** IOException{

file = **new** RandomAccessFile(fileName, "rw");

file.seek(0);

**try** {

MAX\_N\_HOME\_LOC = file.readShort();

nHome = file.readShort();

nColl = file.readShort();

} **catch** (Exception e) {

MAX\_N\_HOME\_LOC = 20;

nHome = 0;

nColl = 0;

file.writeShort(MAX\_N\_HOME\_LOC);

file.writeShort(nHome);

file.writeShort(nColl);

}

log.displayThis("FILE STATUS > CountryData FILE opened");

dtr = **new** DataTableRecord();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* hashFunction which returns homeAddress

\* **@param** code

\* **@return** homeAddress

\*/

**private** **void** **hashFunction**(**char**[] code){

homeAddress = (code[0] \* code[1] \* code[2]) % MAX\_N\_HOME\_LOC;

homeAddress = (homeAddress == 0) ? MAX\_N\_HOME\_LOC : homeAddress;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* IN command

\* **@param** code

\* **@param** id

\* **@param** name

\* **@param** continent

\* **@param** area

\* **@param** population

\* **@param** lifeExpectancy

\* **@param** userApp

\* **@param** log

\* **@throws** IOException

\*/

**public** **int** **insertCountry**(String code, **short** id, String name,

String continent, **int** area, **long** population, **float** lifeExpectancy,

Log log) **throws** IOException {

**boolean**[] status;

**short** oldLink;

hashFunction(code.toCharArray());

dtr.byteOffset(homeAddress);

status = dtr.locateWithStatus(file);

**if** ((status[0] && status[1]) || (!status[0] && status[1])){

dtr.write1Country(file, code, id, name, continent, area,

population, lifeExpectancy, (**short**) -1);

nHome++;

}

**else** **if** (status[0] && !status[1]){

nColl++;

**int** collPos = MAX\_N\_HOME\_LOC + nColl;

dtr.readLink(file);

oldLink = dtr.getLink();

dtr.byteOffset(homeAddress);

dtr.writeLink(file, collPos);

dtr.byteOffset(collPos);

dtr.locateWithStatus(file);

dtr.write1Country(file, code, id, name, continent, area,

population, lifeExpectancy, oldLink);

}

**return** homeAddress;

}

/\*\*\*\*\*\*\*\*\*\*\*\*

\* DC command

\* **@param** code

\* **@param** log

\* **@throws** IOException

\*/

**public** **void** **deleteByCode**(String code, Log log) **throws** IOException {

log.displayThis(" [SORRY, Select By Code module not yet working]");

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* SC command

\* **@param** code

\* **@param** log

\* **@throws** IOException

\*/

**public** **void** **selectByCode**(String code, Log log) **throws** IOException {

hashFunction(code.toCharArray());

dtr.byteOffset(homeAddress);

**boolean**[] status = dtr.locateWithStatus(file);

**if** (status[0] && !status[1]){

dtr.read1Country(file);

**if** (dtr.getCode().equals(code))

log.displayThis(" "

+ log.country(dtr.getCode(), dtr.getId(),

dtr.getName(), dtr.getContinent(),

dtr.getArea(), dtr.getPopulation(), dtr.getLifeExpectancy()));

**else**

**if** (dtr.getLink() == -1)

log.displayThis(" SORRY, invalid code");

**else**

**if** (findInColl(dtr.getLink(), code) == -1)

log.displayThis(" SORRY, invalid code");

**else**

log.displayThis(" "

+ log.country(dtr.getCode(), dtr.getId(),

dtr.getName(), dtr.getContinent(),

dtr.getArea(), dtr.getPopulation(),

dtr.getLifeExpectancy()));

}

**else**

log.displayThis(" SORRY, invalid code");

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* recursive method for finding elements in collision area

\* **@param** link

\* **@param** code

\* **@return**

\* **@throws** IOException

\*/

**public** **int** **findInColl**(**int** link, String code) **throws** IOException{

**if** (link != -1){

dtr.byteOffset(link);

dtr.locateWithStatus(file);

dtr.read1Country(file);

**if** (dtr.getCode().equals(code))

**return** 1;

**else**

**return** findInColl(dtr.getLink(), code);

}

**else**

**return** -1;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Dumps header data into CountryData.bin

\* **@throws** IOException

\*/

**public** **void** **finishUp**() **throws** IOException{

file.seek(0);

file.writeShort(MAX\_N\_HOME\_LOC);

file.writeShort(nHome);

file.writeShort(nColl);

file.close();

}

}

**package** edu.wmich.cs3310.a3;

**import** java.io.\*;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Manages file handling

\* Countries Of The World App 1.5

\* **@author** Caleb Viola

\*/

# **public** **class** DataTableRecord {

**private** String code;

**private** **short** id;

**private** String name;

**private** String continent;

**private** **int** area;

**private** **long** population;

**private** **float** lifeExpectancy;

**private** **short** link;

**private** **int** byteOffset;

**private** **int** sizeOfHeaderRec = 2 /\*MAX\_N\_HOME\_LOC\*/ + 2 /\*nHome\*/ + 2 /\*nColl\*/;

**private** **int** sizeOfDataRec = 3 /\*code\*/+ 2 /\*id\*/+ 15 /\*name\*/

+ 13 /\*continent\*/+ 4 /\*area\*/+ 8 /\*population\*/+ 4 /\*lifeExpectancy\*/ + 2 /\*link\*/;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* For reading a country from record

\* **@param** file

\* **@throws** IOException

\*/

**public** **void** **read1Country**(RandomAccessFile file) **throws** IOException{

code = *readString*(3, file);

id = file.readShort();

name = *readString*(15, file);

continent = *readString*(13, file);

area = file.readInt();

population = file.readLong();

lifeExpectancy = file.readFloat();

link = file.readShort();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Reads specified length as string

\* **@param** length

\* **@param** file

\* **@return** string

\* **@throws** IOException

\*/

**public** **static** String **readString**(**int** length, RandomAccessFile file)

**throws** IOException {

String string = "";

**for** (**int** x = 0; x < length; x++)

string += (**char**) file.readByte();

**return** string;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* read link field at current offset

\* **@param** file

\* **@throws** IOException

\*/

**public** **void** **readLink**(RandomAccessFile file) **throws** IOException{

offsetForLink();

file.seek(byteOffset);

link = file.readShort();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* write link field at current offset

\* **@param** file

\* **@param** link

\* **@throws** IOException

\*/

**public** **void** **writeLink**(RandomAccessFile file, **int** link) **throws** IOException{

offsetForLink();

file.seek(byteOffset);

file.writeShort(link);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* For writing a country to record

\* **@param** file

\* **@param** id

\* **@param** code

\* **@param** name

\* **@param** continent

\* **@param** area

\* **@param** population

\* **@param** lifeExpectancy

\* **@throws** IOException

\*/

**public** **void** **write1Country**(RandomAccessFile file, String code, **short** id,

String name, String continent, **int** area,

**long** population, **float** lifeExpectancy, **short** link) **throws** IOException {

file.writeBytes(code);

file.writeShort(id);

file.writeBytes(String.*format*("%-15.15s", name));

file.writeBytes(String.*format*("%-13.13s", continent));

file.writeInt(area);

file.writeLong(population);

file.writeFloat(lifeExpectancy);

file.writeShort(link);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Discover nature of file at given offset and return status

\* **@param** file

\* **@return**

\*/

**public** **boolean**[] **locateWithStatus**(RandomAccessFile file){

**try** {

file.seek(byteOffset);

**try** {

**char** test = (**char**)file.readByte();

file.seek(byteOffset);

**if** (Character.*isLetter*(test))

**return** **new** **boolean**[] { **true**, **false** }; // Occupied spot

**return** **new** **boolean**[] { **false**, **true** }; // Empty spot

} **catch** (IOException e) {

**return** **new** **boolean**[] { **true**, **true** };// New spot

}

} **catch** (IOException e) {

**return** **new** **boolean**[] { **false**, **false** }; // Out of range

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Calculate offset

\* **@param** rrn

\*/

**public** **void** **byteOffset**(**int** rrn){

byteOffset = sizeOfHeaderRec + ((rrn - 1) \* sizeOfDataRec);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Calculate offset for link

\* **@param** rrn

\*/

**public** **void** **offsetForLink**(){

byteOffset += (sizeOfDataRec - 2);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for code

\* **@return**

\*/

**public** String **getCode**(){

**return** code;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for id

\* **@return**

\*/

**public** **short** **getId**(){

**return** id;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for name

\* **@return**

\*/

**public** String **getName**(){

**return** name;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for continent

\* **@return**

\*/

**public** String **getContinent**(){

**return** continent;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for area

\* **@return**

\*/

**public** **int** **getArea**(){

**return** area;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for population

\* **@return**

\*/

**public** **long** **getPopulation**(){

**return** population;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for lifeExpectancy

\* **@return**

\*/

**public** **float** **getLifeExpectancy**(){

**return** lifeExpectancy;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for link

\* **@return**

\*/

**public** **short** **getLink**() {

**return** link;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Getter for byteOffset

\* **@return**

\*/

**public** **int** **getByteOffset**() {

**return** byteOffset;

}

}

**package** edu.wmich.cs3310.a3;

**import** java.io.\*;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Manages Log.txt

\* Countries Of The World App 1.5

\* **@author** Caleb Viola

\*/

# **public** **class** Log {

**private** PrintWriter log;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Opens Log.txt in append or truncate mode

\* **@param** append

\* **@throws** IOException

\*/

**public** **Log**(**boolean** append) **throws** IOException{

**if** (!append)

log = **new** PrintWriter("Log.txt");

**else**

log = **new** PrintWriter(**new** FileOutputStream(**new** File("Log.txt"),

append));

log.println("FILE STATUS > Log FILE opened");

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Formatted string for printing country

\* **@param** code

\* **@param** id

\* **@param** name

\* **@param** continent

\* **@param** area

\* **@param** population

\* **@param** lifeExpectancy

\* **@return**

\*/

**public** String **country**(String code, **short** id, String name, String continent,

**int** area, **long** population, **float** lifeExpectancy) {

**return** String.*format*("%s %03d %-18.18s %-13.13s %,10d %,13d %4.1f",

code, id, name, continent, area, population, lifeExpectancy);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Prints line to file

\* **@param** message string to write to file

\* **@throws** IOException

\*/

**public** <**T**> **void** **displayThis**(**T** message) **throws** IOException{

log.println(message);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Creates log entry and closes file

\* **@throws** IOException

\*/

**public** **void** **finishUp**() **throws** IOException{

log.println("FILE STATUS > Log FILE closed");

log.close();

}

}

**package** edu.wmich.cs3310.a3;

**import** java.io.\*;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Countries Of The World App 1.0

\* Modified version (Kaminski's approval)

\* **@author** Caleb Viola

\*/

# **public** **class** PrettyPrintUtility {

**private** **static** RandomAccessFile *file*;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Displays the countrydata binary file to the log file

\* **@throws** IOException on I/O error

\*/

**public** **static** **void** **main**() **throws** IOException {

*file* = **new** RandomAccessFile("CountryData.bin", "r");

PrintWriter log = **new** PrintWriter(**new** FileOutputStream(

**new** File("log.txt"), **true**));

log.println("STATUS > Log FILE opened");

log.println("STATUS > CountryData FILE opened");

*file*.seek(0);

**short** MAX\_N\_HOME\_LOC = *file*.readShort();

**short** nHome = *file*.readShort();

**short** nColl = *file*.readShort();

log.println("DATA STORAGE");

log.println(String.*format*("MAX\_N\_HOME\_LOC: %02d | nHome: %02d | nColl: %02d",

MAX\_N\_HOME\_LOC, nHome, nColl));

log.println("LOC/ CDE ID- NAME----------- CONTINENT---- ------AREA ---POPULATION LIFE LINK");

String toPrint;

**for** (**short** x = 1; x <= (MAX\_N\_HOME\_LOC + nColl); x++){

toPrint = String.*format*("%03d/ %s %03d %s %s %,10d %,13d %4.1f %03d",

x, *readString*(3), *file*.readShort(), *readString*(15), *readString*(13),

*file*.readInt(), *file*.readLong(), *file*.readFloat(), *file*.readShort());

toPrint = (Character.*isLetter*(toPrint.charAt(5))

? toPrint

: String.*format*("%03d/ ...", x));

log.println(toPrint);

}

log.println("+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++");

log.println("STATUS > CountryData FILE closed");

log.println("STATUS > Log FILE closed");

*file*.close();

log.close();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Reads specified length as string

\* **@param** length

\* **@param** file

\* **@return**

\* **@throws** IOException

\*/

**public** **static** String **readString**(**int** length)

**throws** IOException {

String string = "";

**for** (**int** x = 0; x < length; x++)

string += (**char**) *file*.readByte();

**return** string;

}

}