

TCP3151

Integrative Programming and Technologies

**Assignment Report**

Lab Section: IP1A

BY

|  |  |  |
| --- | --- | --- |
| **STUDENT ID** | **STUDENT NAME** | **MAJOR** |
| 1131119882 | Koh Kah Siang | ST |
| 1151105086 | Daphne Bunga Dwiputriane | ST |
| 1151105293 | Wong Chang He | ST |
| 1151104900 | Cheng Kean Wah | ST |

**LECTURER:**

**DR. CHONG LEE YING**

MULTIMEDIA UNIVERSITY

SESSION 2019/2020

TRIMESTER 1

Table of Contents

[Part 1 – Data related to Lost Item Reporting System 1](#_Toc18674614)

[1.1 Brief Introduction of the Data used 1](#_Toc18674615)

[1.2 XML Documents 3](#_Toc18674616)

[1.3 Logical View 25](#_Toc18674617)

[1.4 DTD Documents 27](#_Toc18674618)

[1.5 XSLT Codes 29](#_Toc18674619)

[1.6 HTML Outputs 34](#_Toc18674620)

[Part 2 – JNI Program 35](#_Toc18674621)

[2.1 JNI Program Codes 35](#_Toc18674623)

[Question 1 35](#_Toc18674624)

[Question 2 44](#_Toc18674625)

[2.2 Print Screen of Outputs 51](#_Toc18674626)

[Question 1 51](#_Toc18674627)

[Question 2 52](#_Toc18674628)

# Part 1 – Data related to Lost Item Reporting System

## Brief Introduction of the Data used

The system uses two separates XML files for lost items and found items respectively. For the first XML file – LostItemReport.xml, it shows the items that are reported lost to the Lost and Found Station in Multimedia University Melaka. This XML file using an external DTD file named – LostItemReport.dtd to define its structure and the legal elements and attributes. Besides that, an XSL file – ListOfLostItem.xsl is also linked together to transform the XML elements into XHTML to be displayed to end users.

The root element for LostItemReport.xml is *lostReport*, it consists of zero or more *item* element, each of the *item* has an *id* attribute as primary key of the *item*, *include* attribute is an *IDREFS* used to link related items that are lost together. In the *item,* there are 9 child elements, each of it with its own purpose. The *status* element used to mark the status of the item, its attribute – *choice* may be supplied with“Open” to indicate an item that have lost, “Found” to indicate the item have been found and awaiting to be returned, and lastly “Closed” to indicate the item have been found and returned. The *description* element represents the short description of the item lost. For the *report:datetime* element, it store the date and time when the item is reported lost to the Lost and Found Station, *lost:datetime* imply the date and time range, when the item is suspected lost at, *claim:datetime* reveal the date and time the owner come and claim the item back, shall it is found. The *locations* element may include one or multiple *location* element, it shows the locations where the item is suspected lost at. The *detail* element may include a detail description of the item lost. The *reportBy* shows the owner of the item, the Lost and Found Station may contact this person shall the item is found, by using the email or phone number provided here. The *returnBy* shows the person who found the lost item, and some other information about the person.

For another XML file – FoundItemReport.xml, it shows the found items that are reported to the Lost and Found Station. This XML file linked to an XSL file – ItemTobeClaimed.xsl to display it in XHTML format, and it use an internal DTD to define the elements structure. It had *foundReport* as the root element, and it consist zero or multiple *item* element. The *id* attribute of the *item* element is used to indicate the primary key of the item, and the *include* attribute may link with other items’ *id* attribute to indicate the relativity between the items. The *choice* attribute of *status* element is used to indicate the status of the item, where the “Open” imply the item is awaiting to be claimed, and “Closed” imply that the item have been claimed. The *description* element shows a short description of the item found. For the *report:datetime* element it store the date and time the item is reported to the Lost and Found Station, *found:datetime* shows the date and time the item is being found by the reporting person, *claim:datetime* element shows when the item is being claimed by its owner. The *location* element marks where the item is found at. If there are other detail description of the item found, it may be showed in *detail* element. For the *reportBy* element, it shows the person who found the item and report it to the Lost and Found Station, it also includes some contact information of the person. For *claimBy* element, it shows the person who claimed the item, and some of his/her information.

## XML Documents

**LostItemReport.xml**

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" standalone="no" ?>  <!-- This report is to record the items that are reported LOST to the Lost and Found Station -->  <!-- External DTD -->  <!DOCTYPE lostReport SYSTEM "LostItemReport.dtd" >  <!-- Link to XSL -->  <?xml-stylesheet type="text/xsl" href="ListOfLostItem.xsl" ?>  <lostReport>  <item id="L1001" include="L1002">  <status choice="Closed" />  <description>Wallet</description>  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="03" />  <report:time hour="11" minute="59" />  </report:datetime>    <lost:datetime xmlns:lost="http://www.w3.org/xml/lost/">  <lost:dateFrom year="2019" month="07" date="03" />  <lost:dateTo year="2019" month="07" date="03" />  <lost:timeFrom hour="08" minute="00" />  <lost:timeTo hour="11" minute="30" />  </lost:datetime>  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="04" />  <claim:time hour="08" minute="04" />  </claim:datetime>    <locations>  <location>MSMX0001</location>  <location>CLC Concourse</location>  </locations>  <detail> <CDATA[  Black colour, around RM300 inside, include student id - 1131119882, ic - 950101-01-0001, driving license, and aeon member card  ]]></detail>  <reportBy>  <name>  <firstName>Kah Siang</firstName>  <lastName>Koh</lastName>  </name>  <email>kahsiang95@hotmail.com</email>  <phone>0176815218</phone>  <id>1131119882</id>  </reportBy>  <returnBy>  <name>  <firstName>Daphne</firstName>  <lastName>Bunga Dwiputriane</lastName>  </name>  <email>1151105086@student.mmu.edu.my</email>  <phone>0176356898</phone>  <id>1151105086</id>  </returnBy>  </item>    <item id="L1002"> <!-- indicate an item that have been lost; id incidate the primary key -->  <status choice="Closed" /> <!-- Open: Item is awaiting to be found and returned; Found: Item have been found and awaiting to be returned; Closed: Item have been found and returned -->  <description>Student ID</description> <!-- short description of the item lost -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="03" />  <report:time hour="11" minute="59" />  </report:datetime>  <!-- when the item is suspected lost -->  <lost:datetime xmlns:lost="http://www.w3.org/xml/lost/">  <lost:dateFrom year="2019" month="07" date="03" />  <lost:dateTo year="2019" month="07" date="03" />  <lost:timeFrom hour="08" minute="00" />  <lost:timeTo hour="11" minute="30" />  </lost:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="04" />  <claim:time hour="08" minute="04" />  </claim:datetime>  <!-- where the item is suspected lost -->  <locations>  <location>MSMX0001</location>  <location>CLC Concourse</location>  </locations>  <!-- detail description of the item lost -->  <detail> <![CDATA[  Student Id: 1131119882  ]]></detail>  <!-- item lost is reported by this person -->  <reportBy>  <name>  <firstName>Kah Siang</firstName>  <lastName>Koh</lastName>  </name>  <email>kahsiang95@hotmail.com</email>  <phone>0176815218</phone>  <id>1131119882</id>  </reportBy>  <!-- item is being found by this person -->  <returnBy>  <name>  <firstName>Daphne</firstName>  <lastName>Bunga Dwiputriane</lastName>  </name>  <email>1151105086@student.mmu.edu.my</email>  <phone>0176356898</phone>  <id>1151105086</id>  </returnBy>  </item>  <item id="L1003"> <!-- indicate an item that have been lost; id incidate the primary key -->  <status choice="Open" /> <!-- Open: Item is awaiting to be found and returned; Found: Item have been found and awaiting to be returned; Closed: Item have been found and returned -->  <description>Necklace</description> <!-- short description of the item lost -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="09" />  <report:time hour="11" minute="30" />  </report:datetime>  <!-- when the item is suspected lost -->  <lost:datetime xmlns:lost="http://www.w3.org/xml/lost/">  <lost:dateFrom year="2019" month="07" date="09" />  <lost:dateTo year="2019" month="07" date="09" />  <lost:timeFrom hour="09" minute="00" />  <lost:timeTo hour="11" minute="00" />  </lost:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="" month="" date="" />  <claim:time hour="" minute="" />  </claim:datetime>  <!-- where the item is suspected lost -->  <locations>  <location>MAMR4020</location>  </locations>  <!-- detail description of the item lost -->  <detail> <![CDATA[  Necklace Colour - Silver, Necklace Symbol - Cross, Necklace Length - Short  ]]></detail>  <!-- item lost is reported by this person -->  <reportBy>  <name>  <firstName>Mary Louise</firstName>  <lastName>Wright</lastName>  </name>  <email>maryloui@gmail.com</email>  <phone>0103456789</phone>  <id>118120212</id>  </reportBy>  <!-- item is being found by this person -->  <returnBy>  <name>  <firstName></firstName>  <lastName></lastName>  </name>  <email></email>  <phone></phone>  <id></id>  </returnBy>  </item>    <item id="L1004">  <status choice="Found" />  <description>Laptop Bag</description>  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="16" />  <report:time hour="12" minute="50" />  </report:datetime>    <lost:datetime xmlns:lost="http://www.w3.org/xml/lost/">  <lost:dateFrom year="2019" month="07" date="16" />  <lost:dateTo year="2019" month="07" date="16" />  <lost:timeFrom hour="11" minute="00" />  <lost:timeTo hour="12" minute="30" />  </lost:datetime>  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="" month="" date="" />  <claim:time hour="" minute="" />  </claim:datetime>  <locations>  <location>MSMX2008</location>  </locations>  <detail> <![CDATA[  Black colour, messenger bag, Laptop brand - Dell, Laptop Username - Amabella Klein, Laptop Colour - Grey  ]]></detail>  <reportBy>  <name>  <firstName>Amabella</firstName>  <lastName>Klein</lastName>  </name>  <email>amabellaklein21@gmail.com</email>  <phone>0144705658</phone>  <id>1151105020</id>  </reportBy>  <returnBy>  <name>  <firstName>Chang He</firstName>  <lastName>Wong</lastName>  </name>  <email>1151105293@student.mmu.edu.my</email>  <phone>0129161580</phone>  <id>1151105293</id>  </returnBy>  </item>    <item id="L1005"> <!-- indicate an item that have been lost; id incidate the primary key -->  <status choice="Open" /> <!-- Open: Item is awaiting to be found and returned; Found: Item have been found and awaiting to be returned; Closed: Item have been found and returned -->  <description>Water Bottle</description> <!-- short description of the item lost -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="12" />  <report:time hour="14" minute="30" />  </report:datetime>  <!-- when the item is suspected lost -->  <lost:datetime xmlns:lost="http://www.w3.org/xml/lost/">  <lost:dateFrom year="2019" month="07" date="12" />  <lost:dateTo year="2019" month="07" date="12" />  <lost:timeFrom hour="11" minute="00" />  <lost:timeTo hour="13" minute="00" />  </lost:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="" month="" date="" />  <claim:time hour="" minute="" />  </claim:datetime>  <!-- where the item is suspected lost -->  <locations>  <location>MNBR2007</location>  </locations>  <!-- detail description of the item lost -->  <detail> <![CDATA[  Brand - Tupperware, Colour -Blue, Volume of water - Half empty  ]]></detail>  <!-- item lost is reported by this person -->  <reportBy>  <name>  <firstName>Joseph</firstName>  <lastName>Schooling</lastName>  </name>  <email>joeschooling@yahoo.com</email>  <phone>0165486569</phone>  <id>115130444</id>  </reportBy>  <!-- item is being found by this person -->  <returnBy>  <name>  <firstName></firstName>  <lastName></lastName>  </name>  <email></email>  <phone></phone>  <id></id>  </returnBy>  </item>    <item id="L1006">  <status choice="Closed" />  <description>Wallet</description>  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="03" />  <report:time hour="12" minute="00" />  </report:datetime>    <lost:datetime xmlns:lost="http://www.w3.org/xml/lost/">  <lost:dateFrom year="2019" month="07" date="03" />  <lost:dateTo year="2019" month="07" date="03" />  <lost:timeFrom hour="10" minute="00" />  <lost:timeTo hour="11" minute="50" />  </lost:datetime>  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="10" />  <claim:time hour="11" minute="00" />  </claim:datetime>  <locations>  <location>ACR1006</location>  </locations>  <detail> <![CDATA[  Colour - Grey, Cash - RM10, Driving License - 100100234, CIMB ATM Card, Hong Leong Credit Card  ]]></detail>  <reportBy>  <name>  <firstName>Kosuke</firstName>  <lastName>Hagino</lastName>  </name>  <email>koshaginojp@yahoo.com</email>  <phone>0155587202</phone>  <id>1121202121</id>  </reportBy>  <returnBy>  <name>  <firstName>Frenkie</firstName>  <lastName>de Jong</lastName>  </name>  <email>fdj21@gmail.com</email>  <phone>016-6567898</phone>  <id>1181104554</id>  </returnBy>  </item>  <item id="L1007"> <!-- indicate an item that have been lost; id incidate the primary key -->  <status choice="Open" /> <!-- Open: Item is awaiting to be found and returned; Found: Item have been found and awaiting to be returned; Closed: Item have been found and returned -->  <description>House Keys</description> <!-- short description of the item lost -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="15" />  <report:time hour="16" minute="00" />  </report:datetime>  <!-- when the item is suspected lost -->  <lost:datetime xmlns:lost="http://www.w3.org/xml/lost/">  <lost:dateFrom year="2019" month="07" date="15" />  <lost:dateTo year="2019" month="07" date="15" />  <lost:timeFrom hour="08" minute="30" />  <lost:timeTo hour="12" minute="00" />  </lost:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="" month="" date="" />  <claim:time hour="" minute="" />  </claim:datetime>  <!-- where the item is suspected lost -->  <locations>  <location>ACR3002</location>  </locations>  <!-- detail description of the item lost -->  <detail> <![CDATA[  Number of Keys - 3, Keychain words - Enjoy Spain  ]]></detail>  <!-- item lost is reported by this person -->  <reportBy>  <name>  <firstName>Aaron</firstName>  <lastName>Wan-Bissaka</lastName>  </name>  <email>spiderwan97@gmail.com</email>  <phone>0124202029</phone>  <id>1141100420</id>  </reportBy>  <!-- item is being found by this person -->  <returnBy>  <name>  <firstName></firstName>  <lastName></lastName>  </name>  <email></email>  <phone></phone>  <id></id>  </returnBy>  </item>    <item id="L1008">  <status choice="Closed" />  <description>Backpack</description>  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="24" />  <report:time hour="14" minute="10" />  </report:datetime>    <lost:datetime xmlns:lost="http://www.w3.org/xml/lost/">  <lost:dateFrom year="2019" month="07" date="23" />  <lost:dateTo year="2019" month="07" date="24" />  <lost:timeFrom hour="18" minute="00" />  <lost:timeTo hour="14" minute="00" />  </lost:datetime>  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="26" />  <claim:time hour="15" minute="00" />  </claim:datetime>  <locations>  <location>Exam Hall</location>  </locations>  <detail> <![CDATA[  Colour - Black, Network Security Lecture Notes, Final Year Project drafts, black pencil case, Sony brand powerbank  ]]></detail>  <reportBy>  <name>  <firstName>Mohammad</firstName>  <lastName>Salah</lastName>  </name>  <email>mosalah29@hotmail.com</email>  <phone>0176822910</phone>  <id>1181100106</id>  </reportBy>  <returnBy>  <name>  <firstName>Penny</firstName>  <lastName>Oleksiak</lastName>  </name>  <email>typicalpenny@gmail.com</email>  <phone>016-80902454</phone>  <id>1141205055</id>  </returnBy>  </item>    </lostReport> |

**FoundItemReport.xml**

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" standalone="no" ?>  <!-- This report is to record the items that are reported FOUND to the Lost and Found Station -->  <!-- Link to XSL -->  <?xml-stylesheet type="text/xsl" href="ItemTobeClaimed.xsl" ?>  <!-- Internal DTD -->  <!DOCTYPE foundReport [  <!ELEMENT foundReport (item\*) >  <!ELEMENT item (status, description, report:datetime, found:datetime, claim:datetime, location, detail, reportBy, claimBy) >  <!ELEMENT status EMPTY >  <!ELEMENT description (#PCDATA) >  <!ELEMENT report:datetime (report:date, report:time) >  <!ELEMENT report:date EMPTY >  <!ELEMENT report:time EMPTY >  <!ELEMENT found:datetime (found:date, found:time) >  <!ELEMENT found:date EMPTY >  <!ELEMENT found:time EMPTY >  <!ELEMENT claim:datetime (claim:date, claim:time) >  <!ELEMENT claim:date EMPTY >  <!ELEMENT claim:time EMPTY >  <!ELEMENT location (#PCDATA) >  <!ELEMENT detail (#PCDATA) >  <!ELEMENT reportBy (name, email, phone, id) >  <!ELEMENT claimBy (name, email, phone, id) >  <!ELEMENT name (firstName, lastName) >  <!ELEMENT firstName (#PCDATA) >  <!ELEMENT lastName (#PCDATA) >  <!ELEMENT email (#PCDATA) >  <!ELEMENT phone (#PCDATA) >  <!ELEMENT id (#PCDATA) >  <!ATTLIST item id ID #REQUIRED >  <!ATTLIST item include IDREFS #IMPLIED>  <!ATTLIST status choice (Open | Closed) #REQUIRED >  <!ATTLIST report:datetime xmlns:report CDATA #FIXED "http://www.w3.org/xml/report/">  <!ATTLIST report:date  year CDATA #REQUIRED  month CDATA #REQUIRED  date CDATA #REQUIRED  >  <!ATTLIST report:time  hour CDATA #REQUIRED  minute CDATA #REQUIRED  >  <!ATTLIST found:datetime xmlns:found CDATA #FIXED "http://www.w3.org/xml/found/">  <!ATTLIST found:date  year CDATA #REQUIRED  month CDATA #REQUIRED  date CDATA #REQUIRED  >  <!ATTLIST found:time  hour CDATA #REQUIRED  minute CDATA #REQUIRED  >  <!ATTLIST claim:datetime xmlns:claim CDATA #FIXED "http://www.w3.org/xml/claim/">  <!ATTLIST claim:date  year CDATA #REQUIRED  month CDATA #REQUIRED  date CDATA #REQUIRED  >  <!ATTLIST claim:time  hour CDATA #REQUIRED  minute CDATA #REQUIRED  >  ]>  <foundReport>  <item id="F2001"> <!-- indicate an item that have been found; id incidate the primary key -->  <status choice="Open" /> <!-- Open: Item is awaiting to be claimed; Closed: Item have been claimed -->  <description>Watch</description> <!-- short description of the item found -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="03" />  <report:time hour="17" minute="26" />  </report:datetime>  <!-- when the item is being found by the reporting person -->  <found:datetime xmlns:found="http://www.w3.org/xml/found/">  <found:date year="2019" month="07" date="03" />  <found:time hour="17" minute="00" />  </found:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="" month="" date="" />  <claim:time hour="" minute="" />  </claim:datetime>  <!-- where the item is being found by the reporting person -->  <location>MSMX2003</location>  <!-- detail description of the item found -->  <detail> <![CDATA[  Model: GA-2000S-1A  Colour: Black  ]]></detail>  <!-- item found is reported by this person -->  <reportBy>  <name>  <firstName>Chang He</firstName>  <lastName>Wong</lastName>  </name>  <email>1151105293@student.mmu.edu.my</email>  <phone>012-9161580</phone>  <id>1151105293</id>  </reportBy>  <!-- item is being claimed by this person -->  <claimBy>  <name>  <firstName></firstName>  <lastName></lastName>  </name>  <email></email>  <phone></phone>  <id></id>  </claimBy>  </item>  <item id="F2002" include="F2003"> <!-- indicate an item that have been found; id incidate the primary key -->  <status choice="Closed" /> <!-- Open: Item is awaiting to be claimed; Closed: Item have been claimed -->  <description>Book - "Sapiens"</description> <!-- short description of the item found -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="04" />  <report:time hour="13" minute="21" />  </report:datetime>  <!-- when the item is being found by the reporting person -->  <found:datetime xmlns:found="http://www.w3.org/xml/found/">  <found:date year="2019" month="07" date="04" />  <found:time hour="13" minute="00" />  </found:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="05" />  <claim:time hour="11" minute="12" />  </claim:datetime>  <!-- where the item is being found by the reporting person -->  <location>Library, Second Floor</location>  <!-- detail description of the item found -->  <detail> <![CDATA[  A book by Yuval Noah Harari.  The owner's name is written on the cover of the book -- "Enzo Cheng".  -Besides a wireless keyboard  -One of the tables on second floor of the library  ]]></detail>  <!-- item found is reported by this person -->  <reportBy>  <name>  <firstName>Daphne</firstName>  <lastName>Flower</lastName>  </name>  <email>Dflower@hotmail.com</email>  <phone>0129450002</phone>  <id>1151103434</id>  </reportBy>  <!-- item is being claimed by this person -->  <claimBy>  <name>  <firstName>Enzo</firstName>  <lastName>Cheng</lastName>  </name>  <email>enzocheng977@hotmail.com</email>  <phone>013-3333333</phone>  <id>1151010001</id>  </claimBy>  </item>  <item id="F2003"> <!-- indicate an item that have been found; id incidate the primary key -->  <status choice="Closed" /> <!-- Open: Item is awaiting to be claimed; Closed: Item have been claimed -->  <description>Wireless Keyboard</description> <!-- short description of the item found -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="04" />  <report:time hour="13" minute="21" />  </report:datetime>  <!-- when the item is being found by the reporting person -->  <found:datetime xmlns:found="http://www.w3.org/xml/found/">  <found:date year="2019" month="07" date="04" />  <found:time hour="13" minute="00" />  </found:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="05" />  <claim:time hour="11" minute="12" />  </claim:datetime>  <!-- where the item is being found by the reporting person -->  <location>Library, Second Floor</location>  <!-- detail description of the item found -->  <detail> <![CDATA[  An Alcatroz 2.4G wireless keyboard  Model: AirPad1  Colour: Black  -Has a built-in touch pad  -A book was lying beside this keyboard.  -One of the tables on second floor of the library.  ]]></detail>  <!-- item found is reported by this person -->  <reportBy>  <name>  <firstName>Daphne</firstName>  <lastName>Flower</lastName>  </name>  <email>Dflower@hotmail.com</email>  <phone>0129450002</phone>  <id>1151103434</id>  </reportBy>  <!-- item is being claimed by this person -->  <claimBy>  <name>  <firstName>Enzo</firstName>  <lastName>Cheng</lastName>  </name>  <email>enzocheng977@hotmail.com</email>  <phone>013-3333333</phone>  <id>1151010001</id>  </claimBy>  </item>  <item id="F2004"> <!-- indicate an item that have been found; id incidate the primary key -->  <status choice="Closed" /> <!-- Open: Item is awaiting to be claimed; Closed: Item have been claimed -->  <description>Scientific Calculator</description> <!-- short description of the item found -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="05" />  <report:time hour="14" minute="21" />  </report:datetime>  <!-- when the item is being found by the reporting person -->  <found:datetime xmlns:found="http://www.w3.org/xml/found/">  <found:date year="2019" month="07" date="05" />  <found:time hour="14" minute="13" />  </found:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="05" />  <claim:time hour="18" minute="00" />  </claim:datetime>  <!-- where the item is being found by the reporting person -->  <location>New LP</location>  <!-- detail description of the item found -->  <detail> <![CDATA[  A new scientic calculator with a name tag writting, "Andrew Wong 1131140099"  -Found on one of the tables in new lp  ]]></detail>  <!-- item found is reported by this person -->  <reportBy>  <name>  <firstName>Kah Siang</firstName>  <lastName>Koh</lastName>  </name>  <email>KS123@hotmail.com</email>  <phone>012-9330002</phone>  <id>1120010022</id>  </reportBy>  <!-- item is being claimed by this person -->  <claimBy>  <name>  <firstName>Andrew</firstName>  <lastName>Wang</lastName>  </name>  <email>wang8090@hotmail.com</email>  <phone>013-3334443</phone>  <id>1131140099</id>  </claimBy>  </item>  <item id="F2005"> <!-- indicate an item that have been found; id incidate the primary key -->  <status choice="Open" /> <!-- Open: Item is awaiting to be claimed; Closed: Item have been claimed -->  <description>Portable Harddrive</description> <!-- short description of the item found -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="11" />  <report:time hour="15" minute="30" />  </report:datetime>  <!-- when the item is being found by the reporting person -->  <found:datetime xmlns:found="http://www.w3.org/xml/found/">  <found:date year="2019" month="07" date="11" />  <found:time hour="13" minute="15" />  </found:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="" month="" date="" />  <claim:time hour="" minute="" />  </claim:datetime>  <!-- where the item is being found by the reporting person -->  <location>MSMX3016</location>  <!-- detail description of the item found -->  <detail> <![CDATA[  An blue color Seagate portable harddirve was left on the lecturer's table in the lecture hall.  ]]></detail>  <!-- item found is reported by this person -->  <reportBy>  <name>  <firstName>Elaine</firstName>  <lastName>Cheng</lastName>  </name>  <email>1191191199@hotmail.com</email>  <phone>014-3339983</phone>  <id>1191191199</id>  </reportBy>  <!-- item is being claimed by this person -->  <claimBy>  <name>  <firstName></firstName>  <lastName></lastName>  </name>  <email></email>  <phone></phone>  <id></id>  </claimBy>  </item>  <item id="F2006"> <!-- indicate an item that have been found; id incidate the primary key -->  <status choice="Closed" /> <!-- Open: Item is awaiting to be claimed; Closed: Item have been claimed -->  <description>Keys</description> <!-- short description of the item found -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="15" />  <report:time hour="11" minute="30" />  </report:datetime>  <!-- when the item is being found by the reporting person -->  <found:datetime xmlns:found="http://www.w3.org/xml/found/">  <found:date year="2019" month="07" date="15" />  <found:time hour="10" minute="15" />  </found:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="15" />  <claim:time hour="12" minute="00" />  </claim:datetime>  <!-- where the item is being found by the reporting person -->  <location>Gym room</location>  <!-- detail description of the item found -->  <detail> <![CDATA[  6 six keys chained by a pink key rings was left on the floor of the gym room.  No name tag is chained with the keys.  A tag stating --"B-11-03-03", is chained with the keychain ring.  I suppose these are house keys.  On the floor, beside a chair in the gym room  ]]></detail>  <!-- item found is reported by this person -->  <reportBy>  <name>  <firstName>Lillian</firstName>  <lastName>Wong</lastName>  </name>  <email>1191220091@gmail.com</email>  <phone>018-1113303</phone>  <id>1191220091</id>  </reportBy>  <!-- item is being claimed by this person -->  <claimBy>  <name>  <firstName>Joanna</firstName>  <lastName>Johnson</lastName>  </name>  <email>joannaJohn344@outlook.com</email>  <phone>011-9623335</phone>  <id>1161107869</id>  </claimBy>  </item>  <item id="F2007"> <!-- indicate an item that have been found; id incidate the primary key -->  <status choice="Closed" /> <!-- Open: Item is awaiting to be claimed; Closed: Item have been claimed -->  <description>Black frame glasses</description> <!-- short description of the item found -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="15" />  <report:time hour="16" minute="20" />  </report:datetime>  <!-- when the item is being found by the reporting person -->  <found:datetime xmlns:found="http://www.w3.org/xml/found/">  <found:date year="2019" month="07" date="15" />  <found:time hour="15" minute="00" />  </found:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="2019" month="07" date="16" />  <claim:time hour="09" minute="15" />  </claim:datetime>  <!-- where the item is being found by the reporting person -->  <location>Software Enginnering Lab, MNBR2007</location>  <!-- detail description of the item found -->  <detail> <![CDATA[  A black frame, stylish pair of glasses.  The brand of the frame is called, "Whoosh".  A pair of stylish glasses with black color frame.  Found on a table.  ]]></detail>  <!-- item found is reported by this person -->  <reportBy>  <name>  <firstName>Terrence</firstName>  <lastName>Tam</lastName>  </name>  <email>1131188801@outlook.com</email>  <phone>012-7747731</phone>  <id>1131188801</id>  </reportBy>  <!-- item is being claimed by this person -->  <claimBy>  <name>  <firstName>Daphne</firstName>  <lastName>Flower</lastName>  </name>  <email>Dflower@hotmail.com</email>  <phone>012-9450002</phone>  <id>1151103434</id>  </claimBy>  </item>  <item id="F2008"> <!-- indicate an item that have been found; id incidate the primary key -->  <status choice="Open" /> <!-- Open: Item is awaiting to be claimed; Closed: Item have been claimed -->  <description>Heart shaped necklace</description> <!-- short description of the item found -->  <!-- when the item is reported to the lost and found station -->  <report:datetime xmlns:report="http://www.w3.org/xml/report/">  <report:date year="2019" month="07" date="17" />  <report:time hour="12" minute="40" />  </report:datetime>  <!-- when the item is being found by the reporting person -->  <found:datetime xmlns:found="http://www.w3.org/xml/found/">  <found:date year="2019" month="07" date="17" />  <found:time hour="12" minute="20" />  </found:datetime>  <!-- when the item is being claimed by its owner -->  <claim:datetime xmlns:claim="http://www.w3.org/xml/claim/">  <claim:date year="" month="" date="" />  <claim:time hour="" minute="" />  </claim:datetime>  <!-- where the item is being found by the reporting person -->  <location>ladies toilet, CLC ground floor</location>  <!-- detail description of the item found -->  <detail> <![CDATA[  A platinum, heart shaped necklace was left on the sink of the ladies toilet in CLC.  The necklace is about 20cm long.  ]]></detail>  <!-- item found is reported by this person -->  <reportBy>  <name>  <firstName>Annebelle</firstName>  <lastName>Ferryman</lastName>  </name>  <email>1172281100@gmail.com</email>  <phone>014-9982234</phone>  <id>1172281100</id>  </reportBy>  <!-- item is being claimed by this person -->  <claimBy>  <name>  <firstName></firstName>  <lastName></lastName>  </name>  <email></email>  <phone></phone>  <id></id>  </claimBy>  </item>  </foundReport> |

## Logical View

**Logical View for LostItemReport.xml**

Figure 1shows the logical view of first 2 child elements based on LostItemReport.xml. You may refer to softcopy suppose that the image is not clear.

**![A close up of text on a white background

Description automatically generated**

Figure 1 Logical View for LostItemReport.xml

**Logical View for FoundItemReport.xml**

Figure 2 shows the logical view of first 2 child elements based on FoundItemReport.xml. You may refer to softcopy suppose that the image is not clear.

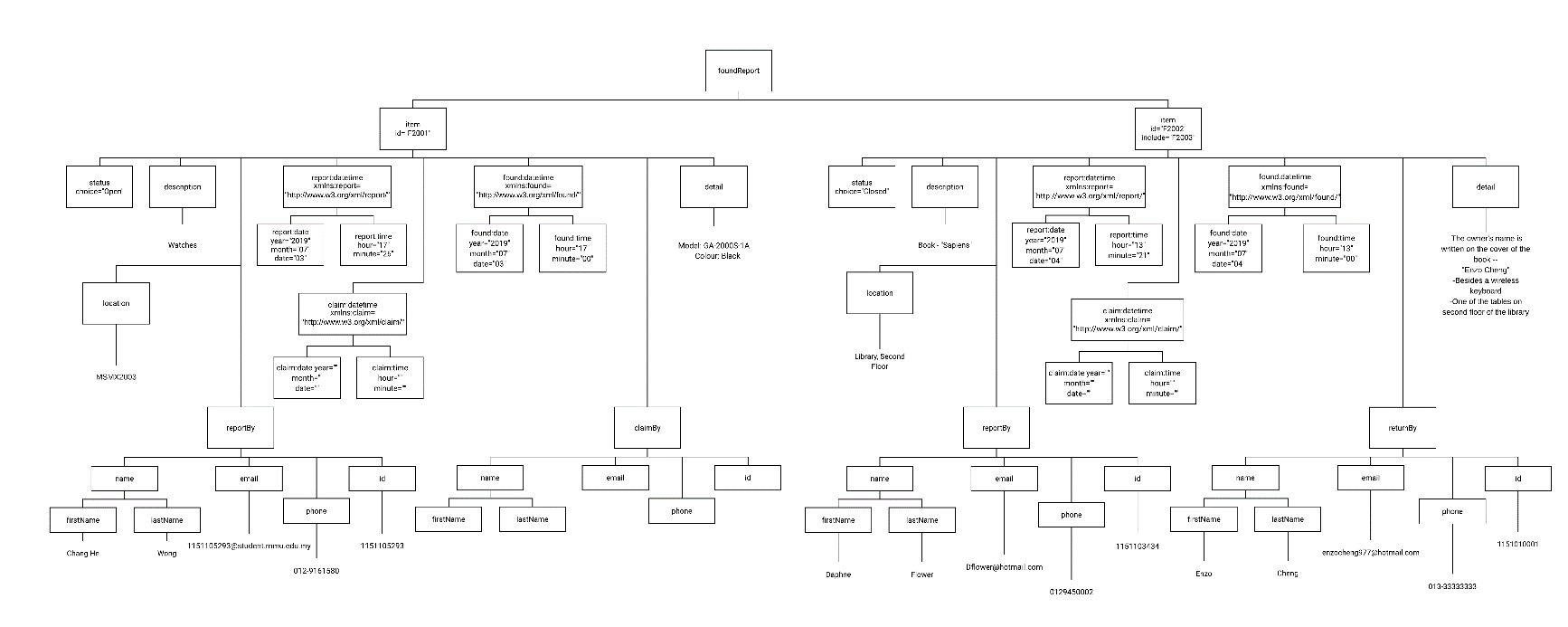


Figure 2 Logical View for FoundItemReport.xml

## DTD Documents

**LostItemReport.dtd for LostItemReport.xml**

|  |
| --- |
| <!ELEMENT lostReport (item\*) >  <!ELEMENT item (status, description, report:datetime, lost:datetime, claim:datetime, locations, detail, reportBy, returnBy) >  <!ELEMENT status EMPTY >  <!ELEMENT description (#PCDATA) >  <!ELEMENT report:datetime (report:date, report:time) >  <!ELEMENT report:date EMPTY >  <!ELEMENT report:time EMPTY >  <!ELEMENT lost:datetime (lost:dateFrom, lost:dateTo, lost:timeFrom, lost:timeTo) >  <!ELEMENT lost:dateFrom EMPTY >  <!ELEMENT lost:dateTo EMPTY >  <!ELEMENT lost:timeFrom EMPTY >  <!ELEMENT lost:timeTo EMPTY >  <!ELEMENT claim:datetime (claim:date, claim:time) >  <!ELEMENT claim:date EMPTY >  <!ELEMENT claim:time EMPTY >  <!ELEMENT locations (location+)>  <!ELEMENT location (#PCDATA) >  <!ELEMENT detail (#PCDATA) >  <!ELEMENT reportBy (name, email, phone, id) >  <!ELEMENT returnBy (name, email, phone, id) >  <!ELEMENT name (firstName, lastName) >  <!ELEMENT firstName (#PCDATA) >  <!ELEMENT lastName (#PCDATA) >  <!ELEMENT email (#PCDATA) >  <!ELEMENT phone (#PCDATA) >  <!ELEMENT id (#PCDATA) >  <!ATTLIST item id ID #REQUIRED >  <!ATTLIST item include IDREFS #IMPLIED>  <!ATTLIST status choice (Open | Found | Closed) #REQUIRED >  <!ATTLIST report:datetime xmlns:report CDATA #FIXED "http://www.w3.org/xml/report/">  <!ATTLIST report:date  year CDATA #REQUIRED  month CDATA #REQUIRED  date CDATA #REQUIRED  >  <!ATTLIST report:time  hour CDATA #REQUIRED  minute CDATA #REQUIRED  >  <!ATTLIST lost:datetime xmlns:lost CDATA #FIXED "http://www.w3.org/xml/lost/">  <!ATTLIST lost:dateFrom  year CDATA #REQUIRED  month CDATA #REQUIRED  date CDATA #REQUIRED  >  <!ATTLIST lost:dateTo  year CDATA #REQUIRED  month CDATA #REQUIRED  date CDATA #REQUIRED  >  <!ATTLIST lost:timeFrom  hour CDATA #REQUIRED  minute CDATA #REQUIRED  >  <!ATTLIST lost:timeTo  hour CDATA #REQUIRED  minute CDATA #REQUIRED  >  <!ATTLIST claim:datetime xmlns:claim CDATA #FIXED "http://www.w3.org/xml/claim/">  <!ATTLIST claim:date  year CDATA #REQUIRED  month CDATA #REQUIRED  date CDATA #REQUIRED  >  <!ATTLIST claim:time  hour CDATA #REQUIRED  minute CDATA #REQUIRED  > |

**DTD for FoundItemReport.xml**

It is an internal DTD included together with FoundItemReport.xml.

## XSLT Codes

**ListOfLostItem.xsl for LostItemReport.xml**

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" ?>  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"  xmlns:report="http://www.w3.org/xml/report/"  xmlns:lost="http://www.w3.org/xml/lost/"  xmlns:claim="http://www.w3.org/xml/claim/">  <xsl:template match="/">    <html>  <head>  <title>List of Lost Item</title>  <style>  #lostItem {  font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;  border-collapse: collapse;  width: 100%;  }  #lostItem td, #lostItem th {  border: 1px solid #ddd;  padding: 8px;  }  #lostItem tr:nth-child(even){background-color: #f2f2f2;}  #lostItem tr:hover {background-color: #ddd;}  #lostItem th {  padding-top: 12px;  padding-bottom: 12px;  text-align: left;  background-color: #64b5f6;  color: white;  }  .open {  color: #66bb6a;  }  .closed {  color: #ef5350;  }  .found {  color: #ffa726;  }  </style>  </head>      <body>        <h2>List of Lost Item</h2>  <table id="lostItem">  <tr>  <th>Item ID</th>  <th>Item</th>  <th>Status</th>  <th colspan="2">Lost between</th>  <th>Where its lost</th>  <th>Detail</th>  <th>Report By</th>  <th>Email</th>  <th>Phone Number</th>  <th>ID</th>  </tr>  <xsl:for-each select="lostReport/item">  <!--<xsl:if test="status/@choice='Open'">-->  <tr>  <td><xsl:value-of select="@id" /></td>  <td><xsl:value-of select="description" /></td>  <xsl:choose>  <xsl:when test="status/@choice='Open'">  <td class="open"><xsl:value-of select="status/@choice" /></td>  </xsl:when>  <xsl:when test="status/@choice='Found'">  <td class="found"><xsl:value-of select="status/@choice" /></td>  </xsl:when>  <xsl:otherwise>  <td class="closed"><xsl:value-of select="status/@choice" /></td>  </xsl:otherwise>  </xsl:choose>  <td>  <xsl:value-of select="lost:datetime/lost:dateFrom/@year" />-<xsl:value-of select="lost:datetime/lost:dateFrom/@month" />-<xsl:value-of select="lost:datetime/lost:dateFrom/@date" />@<xsl:value-of select="lost:datetime/lost:timeFrom/@hour" />:<xsl:value-of select="lost:datetime/lost:timeFrom/@minute" />  </td>  <td>  <xsl:value-of select="lost:datetime/lost:dateTo/@year" />-<xsl:value-of select="lost:datetime/lost:dateTo/@month" />-<xsl:value-of select="lost:datetime/lost:dateTo/@date" />@<xsl:value-of select="lost:datetime/lost:timeTo/@hour" />:<xsl:value-of select="lost:datetime/lost:timeTo/@minute" />  </td>  <td>  <xsl:for-each select="locations/location">  -<xsl:value-of select="." /><br/>  </xsl:for-each>  </td>  <td><xsl:value-of select="detail" /></td>  <td><xsl:value-of select="reportBy/name/firstName" /></td>  <td><xsl:value-of select="reportBy/email" /></td>  <td><xsl:value-of select="reportBy/phone" /></td>  <td><xsl:value-of select="reportBy/id" /></td>  </tr>  <!--</xsl:if>-->  </xsl:for-each>  </table>      </body>    </html>  </xsl:template>  </xsl:stylesheet> |

**ItemTobeClaimed.xsl for FoundItemReport.xml**

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" ?>  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"  xmlns:report="http://www.w3.org/xml/report/"  xmlns:found="http://www.w3.org/xml/found/"  xmlns:claim="http://www.w3.org/xml/claim/">  <xsl:template match="/">  <html>  <head>  <title>Items to be Claimed</title>  <style>  .foundItem {  font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;  border-collapse: collapse;  width: 100%;  }  .foundItem td, .foundItem th {  border: 1px solid #ddd;  padding: 8px;  }  .foundItem tr:nth-child(even){background-color: #f2f2f2;}  .foundItem tr:hover {background-color: #ddd;}  .foundItem th {  padding-top: 12px;  padding-bottom: 12px;  text-align: left;  background-color: #c63f17;  color: white;  }  </style>  </head>    <body>  <h2>Items to be Claimed</h2>  <xsl:apply-templates/>  </body>  </html>  </xsl:template>  <xsl:template match="item">  <xsl:if test="status/@choice='Open'">  <table class="foundItem">  <tr>  <th colspan="2"><xsl:apply-templates select="@id" /> - <xsl:apply-templates select="description" /></th>  </tr>  <tr><xsl:apply-templates select="found:datetime" /></tr>  <tr><xsl:apply-templates select="location" /></tr>  <tr><xsl:apply-templates select="detail" /></tr>  <tr style="height:20px;"></tr>  </table>  </xsl:if>  </xsl:template>  <xsl:template match="@id">  <xsl:value-of select="."/>  </xsl:template>  <xsl:template match="description">  <xsl:value-of select="."/>  </xsl:template>  <xsl:template match="found:datetime">  <td style="width:130px">When its found: </td>  <td><xsl:apply-templates select="found:date" /> @ <xsl:apply-templates select="found:time" /></td>  </xsl:template>  <xsl:template match="found:date">  <xsl:value-of select="@year" />-<xsl:value-of select="@month" />-<xsl:value-of select="@date" />  </xsl:template>  <xsl:template match="found:time">  <xsl:value-of select="@hour" />:<xsl:value-of select="@minute" />  </xsl:template>  <xsl:template match="location">  <td>Location found: </td>  <td><xsl:value-of select="."/></td>  </xsl:template>  <xsl:template match="detail">  <td>Detail: </td>  <td><xsl:value-of select="." /></td>  </xsl:template>  </xsl:stylesheet> |

## HTML Outputs

**ListOfLostItem.xsl for LostItemReport.xml**

Figure 3 shows the print screen of the LostItemReport.xml output after transform with ListOfLostItem.xsl. The XSL file extract some useful information from the XML file and display it in a table form. Besides that, the status of the items is coloured with respect to its status, to ease the end user to find the relevant items.

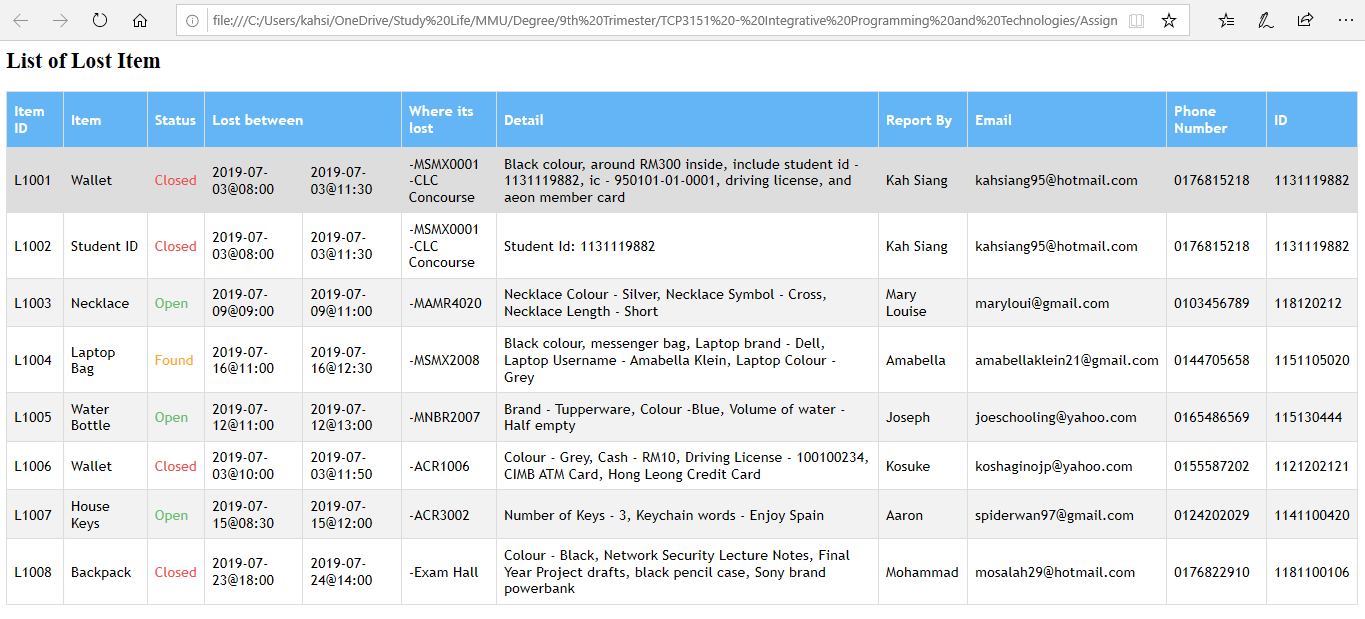


Figure 3 Print Screen of XHTML Output of LostItemReport.xml

**ItemTobeClaimed.xsl for FoundItemReport.xml**

Figure 4 shows the print screen of FoundItemReport.xml output after transform with ItemTobeClaimed.xsl. The XSL file extract some important information from the XML file and display it in a tabular form. It will only show the items where its status is “Open” and hide other items which have been claimed. This may ease the end user to find the items which are awaiting to be claimed by its owner.

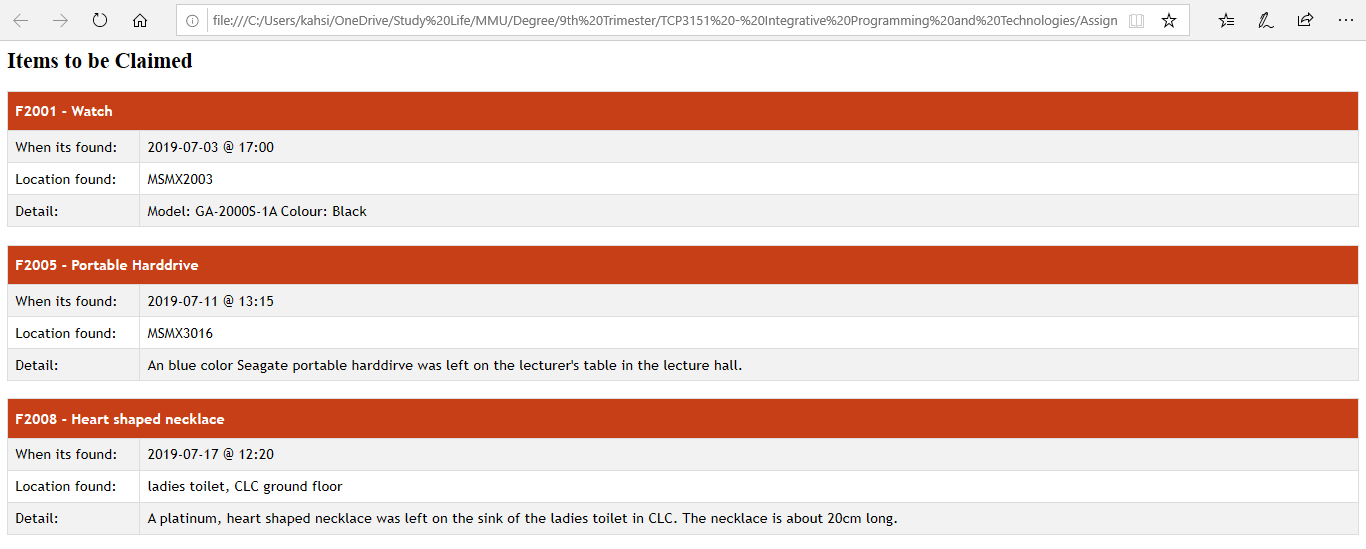


Figure 4 Print Screen of XHTML Output of FoundItemReport.xml

# Part 2 – JNI Program



## JNI Program Codes

### Question 1

**ClassifyHonours.java**

|  |
| --- |
| import java.util.Scanner;  import java.io.IOException;  class ClassifyHonours{  // Native methods  private native String classifyDiploma(double cgpa);  private native String classifyDegree();  private native void estimateNextGPA(double cgpa);    private double jCgpa;  // To clear the terminal  public static void clearScreen(){  try {  new ProcessBuilder("cmd", "/c", "cls").inheritIO().start().waitFor();  } catch (InterruptedException | IOException e) {  e.printStackTrace();  }  }  // Header line for the CLI program  private void displayTitle(){  System.out.println("Classification of Honours");  System.out.println("----------------------------------------------------");  }  // get selection of qualification from user  private int getSelection(Scanner scanner){  int selection;  boolean error;    // clear the terminal  ClassifyHonours.clearScreen();    displayTitle();  System.out.println("1. Diploma");  System.out.println("2. Degree");  System.out.print("Please select your qualification: ");  do{  error = false;  // get the user's input as selection  try{  selection = Integer.parseInt(scanner.nextLine());  }  // if user input is not a number  catch(NumberFormatException e){  selection = 0;  }  // if the selection is not 1 or 2, ask the user to re-enter it again until a valid selection is made  if(selection != 1 && selection != 2){  System.out.println();  System.out.println("Invalid selection!");  System.out.print("Please select your qualification (1|2): ");  error = true;  }  } while(error);  return selection;  }  // Get CGPA from user input  private double getCgpa(Scanner scanner){  double cgpa;  boolean error;  String decimalError;  System.out.print("Please enter your current CGPA: ");  do{  error = false;  decimalError = "";  // get user's input as cgpa  try{  cgpa = Double.parseDouble(scanner.nextLine());  }  // if user input is not a number  catch(NumberFormatException e){  cgpa = -1;  }  // if the cgpa is not in the valid range, ask the user to re-enter again, until a valid input is inserted  if(cgpa < 0 || cgpa > 4){  error = true;  }  // if the cgpa inserted is not 2 decimal places  else if(!validDecimalPlaces(cgpa, 2)){  decimalError = " CGPA must be within 2 decimal places";  error = true;  }    // display error message  if(error){  System.out.println();  System.out.println("Invalid input!" + decimalError);  System.out.print("Please enter your current CGPA (0.00 - 4.00): ");  }  } while(error);  return cgpa;  }  private boolean validDecimalPlaces(double number, int places){  String text = Double.toString(Math.abs(number));  int integerPlaces = text.indexOf('.');  int decimalPlaces = text.length() - integerPlaces - 1;  return (decimalPlaces <= places);  }  // To calculate the GPA to get for current trimester, given the target CGPA  private boolean calculateNextGPA(Scanner scanner){  String input;  System.out.println();  System.out.print("Do you wish to calculate the target GPA to get for current trimester? (Y|N): ");  // get user's input  input = scanner.nextLine();  input = input.toUpperCase();  // if input is not "Y" or "YES", will be evaluate as "N" or false  if (input.equals("Y") || input.equals("YES")){  return true;  }  else{  return false;  }  }  public static void main(String[] args){  String honour = "";  boolean calculateNext = false;  ClassifyHonours classification = new ClassifyHonours();  Scanner scanner = new Scanner(System.in);  // Get user input for qualification  int selection = classification.getSelection(scanner);  // Get user inpur as CGPA  double cgpa = classification.getCgpa(scanner);  // if the cgpa is fail  if(cgpa < 2){  System.out.println("Your are not qualified to get any certificate with your current CGPA of " + cgpa + ".");  }  else{  switch(selection){  case 1:  // return the classification of honour  honour = classification.classifyDiploma(cgpa);  break;  case 2:  // just to demostrate another way to get cgpa'value  // assign cgpa to instance variable 'jCgpa'  classification.jCgpa = cgpa;  // return the classification of honour  honour = classification.classifyDegree();  break;  }  // Display the result in command prompt  System.out.println("Your are qualified to get \"" + honour + "\" certificate with your current CGPA of " + cgpa + ".");  }  // To calculate the GPA to get for current trimester, given the target CGPA  calculateNext = classification.calculateNextGPA(scanner);  if(calculateNext){  classification.estimateNextGPA(cgpa);  }  // avoid resource leak  scanner.close();  }  // Load library files  static{  System.loadLibrary("DiplomaClassification");  System.loadLibrary("DegreeClassification");  System.loadLibrary("EstimateNextGPA");  }  } |

**DegreeClassification.c**

|  |
| --- |
| #include <stdio.h>  #include <jni.h>  #include <string.h>  JNIEXPORT jstring JNICALL Java\_ClassifyHonours\_classifyDegree(JNIEnv \*env, jobject object){  char honour[15] = "Fail";  // just to demostrate another way to get cgpa'value  // To get cgpa's value from Java  jclass clazz = (\*env)->GetObjectClass(env, object);  jfieldID fid = (\*env)->GetFieldID(env, clazz, "jCgpa", "D");  jdouble cgpa = (\*env)->GetDoubleField(env, object, fid);  if(cgpa >= 3.67){  strncpy(honour, "First", 15);  }  else if(cgpa >= 3.33){  strncpy(honour, "Second Upper", 15);  }  else if(cgpa >= 2.67){  strncpy(honour, "Second Lower", 15);  }  else if(cgpa >= 2.0){  strncpy(honour, "Third", 15);  }  // convert char[] to java.lang.String  return (\*env)->NewStringUTF(env, honour);  } |

**DiplomaClassification.c**

|  |
| --- |
| #include <stdio.h>  #include <jni.h>  #include <string.h>  JNIEXPORT jstring JNICALL Java\_ClassifyHonours\_classifyDiploma(JNIEnv \*env, jobject object, jdouble cgpa){  char honour[15] = "Fail";  if(cgpa >= 3.5){  strncpy(honour, "Distinction", 15);  }  else if(cgpa >= 3.0){  strncpy(honour, "Credit", 15);  }  else if(cgpa >= 2.0){  strncpy(honour, "Pass", 15);  }  // convert char[] to java.lang.String  return (\*env)->NewStringUTF(env, honour);  } |

**EstimateNextGPA.c**

|  |
| --- |
| #include <stdio.h>  #include <jni.h>  #include <string.h>  #include <stdlib.h>  #include <stdbool.h>  #include <ctype.h>  // evaluate whether the string is integer  int isInteger(char credit[]){  bool valid = false;  int len = strlen(credit);    // strip trailing newline or other white space  while (len > 0 && isspace(credit[len - 1])){  len--;  }    if (len > 0)  {  valid = true;  for (int i = 0; i < len; i++)  {  if (!isdigit(credit[i]))  {  valid = false;  break;  }  }  }    return valid;  }  // get the credit hours earned  int getCompletedCredit(){  char completedCreditString[15];  bool error;  do{  error = false;  // get input from user  printf("Your completed credit hours: ");  scanf("%s", completedCreditString);  // if the input is not positive integer, prompt error message, and get the user input again  if(!isInteger(completedCreditString)){  printf("\nInvalid input! Positive integer only!\n");  error = true;  }  } while(error);  // convert string to int  return atoi(completedCreditString);  }  // get the currently taking credit hours by the users  int getCurrentCredit(){  char currentCreditString[15];  bool error;  do{  error = false;  // get input from user  printf("Your current credit hours: ");  scanf("%s", currentCreditString);  // if the input is not positive integer, prompt error message, and get the user input again  if(!isInteger(currentCreditString)){  printf("\nInvalid input! Positive integer only!\n");  error = true;  }  } while(error);  // convert string to int  return atoi(currentCreditString);  }  // get the target cgpa user wish to achieve  float getTargetCgpa(JNIEnv \*env, jobject object, jdouble currentCgpa){  jdouble targetCgpa;  char targetCgpaString[15];  bool error;    // connect the java's method  jclass clazz = (\*env)->GetObjectClass(env, object);  jmethodID mid = (\*env)->GetMethodID(env, clazz, "validDecimalPlaces", "(DI)Z");  do{  error = false;  // get user input  printf("Your target CGPA for this trimester: ");  scanf("%s", targetCgpaString);  // convert user input to double  targetCgpa = atof(targetCgpaString);    // if the input is lesser than current cgpa or greater than 4, prompt error, and get user input again  if(targetCgpa < currentCgpa || targetCgpa > 4){  printf("\nInvalid input! CGPA should be within (%.2f - 4.00)!\n", currentCgpa);  error = true;  }  // if the cgpa inserted is not 2 decimal places  else if(!(\*env)->CallBooleanMethod(env, object, mid, targetCgpa, 2)){  printf("\nInvalid input! CGPA must be within 2 decimal places!\n");  error = true;  }  } while(error);  return targetCgpa;  }  // calculate the gpa to get in this trimester in order to get the target cgpa  float calculateGpaToGet(float currentCgpa, int completedCredit, int currentCredit, float targetCgpa){  return ((targetCgpa \* (currentCredit + completedCredit)) - (currentCgpa \* completedCredit)) / currentCredit;  }  JNIEXPORT void JNICALL Java\_ClassifyHonours\_estimateNextGPA(JNIEnv \*env, jobject object, jdouble currentCgpa){  int completedCredit;  int currentCredit;  float targetCgpa;  float gpaToGet;  // display the current cgpa  printf("Your current CGPA: %.2f\n", currentCgpa);  // get the credit hours earned  completedCredit = getCompletedCredit();  // get the currently taking credit hours by the users  currentCredit = getCurrentCredit();  // get the target cgpa user wish to achieve  targetCgpa = getTargetCgpa(env, object, currentCgpa);  // calculate the gpa to get in this trimester in order to get the target cgpa  gpaToGet = calculateGpaToGet(currentCgpa, completedCredit, currentCredit, targetCgpa);  // display to result  printf("You should get a GPA of %.2f to raise your CGPA to %.2f", gpaToGet, targetCgpa);  } |

### Question 2

**ArrayGenerate.java**

|  |
| --- |
| import java.util.Scanner;  import java.io.IOException;  class ArrayGenerate{  private native int[] arrayGenerator(int length);  private native int arrayMinMaxDiff(int[] arrayNumber);  private native void arrayMiscellaneous(int[] arrayNumber);  // To clear the terminal  public static void clearScreen(){  try {  new ProcessBuilder("cmd", "/c", "cls").inheritIO().start().waitFor();  }  catch (InterruptedException | IOException e) {  e.printStackTrace();  }  }  // Header line for the CLI program  private void displayTitle(){  // clear the terminal  ArrayGenerate.clearScreen();  System.out.println("Array Generation");  System.out.println("----------------------------------------------------");  }  // get user input as length for array  private int inputArrayLength(Scanner scanner){  int length;  boolean error;  do{  error = false;    // get user input as length  System.out.print("Size of array to generate: ");  try{  length = Integer.parseInt(scanner.nextLine());  }  // if user input is not a number  catch(NumberFormatException e){  length = 0;  }  // if the length is not valid, ask the user to re-enter again, until a valid input is inserted  if(length < 1){  System.out.println();  System.out.println("Invalid input! Size of array must be equal or larger than 1");  error = true;  }  } while(error);  return length;  }  public static void main(String[] args){  int arrayLength;  int diff;  int arrayNum[];  ArrayGenerate arrayGenerate = new ArrayGenerate();    Scanner scanner = new Scanner(System.in);  // display the header line  arrayGenerate.displayTitle();  // get the length of the array from user  arrayLength = arrayGenerate.inputArrayLength(scanner);  // generate the array with random numbers  arrayNum = arrayGenerate.arrayGenerator(arrayLength);  // calculate the different between the largest and smallest number  diff = arrayGenerate.arrayMinMaxDiff(arrayNum);  System.out.println("The difference between min and max number is " + diff + ".");  System.out.println();  // extra features for the array  arrayGenerate.arrayMiscellaneous(arrayNum);  System.out.println();  System.out.print("Press any key to continue...");  scanner.nextLine();  // avoid resource leak  scanner.close();  }  static{  System.loadLibrary("ArrayGenerator");  System.loadLibrary("ArrayMinMaxDiff");  System.loadLibrary("ArrayMiscellaneous");  }  } |

**ArrayGenerator.c**

|  |
| --- |
| #include <stdio.h>  #include <jni.h>  #include <stdlib.h>  #include <stdbool.h>  #include <time.h>  // generate the array with random number [1-100] with given length  // amendment to the assignment question @ 20190904  // display all elements in the array instead of only less than 50  void GenerateArrayDisplayL50(jint \*theArray, int length){  bool numLessThan50 = false;  printf("Numbers less than 50 in the array: ");  for(int i = 0; i < length; i++){  // generate the random numbers  // (rand() % (MAX - MIN + 1)) + MIN  theArray[i] = (rand() % (100 - 1 + 1)) + 1;  // display the numbers that are less than 50  if(theArray[i] < 50){  numLessThan50 = true;  printf("%ld ", theArray[i]);  }  }    // if no number less than 50  if(!numLessThan50){  printf("not found");  }  printf("\n");  }  // generate the array with random number [1-100] with given length  // amendment to the assignment question @ 20190904  // display all elements in the array instead of only less than 50  void GenerateArray(jint \*theArray, int length){  int countLess50 = 0;  printf("Numbers in the array: ");  for(int i = 0; i < length; i++){  // generate the random numbers  // (rand() % (MAX - MIN + 1)) + MIN  theArray[i] = (rand() % (100 - 1 + 1)) + 1;  printf("%ld ", theArray[i]);  // count the number of elements that are less than 50  if(theArray[i] < 50){  countLess50++;  }  }  printf("\n");  // display the number of elements less than 50  printf("Number of elements less than 50: %d\n", countLess50);  }  JNIEXPORT jintArray JNICALL Java\_ArrayGenerate\_arrayGenerator(JNIEnv \*env, jobject object, jint length){  // c array  jint randomNumber[length];  // create a new java array  jintArray javaArray;  javaArray = (\*env)->NewIntArray(env, length);  // Use current time as seed for random generator  srand(time(0));  // generate the array  GenerateArray(randomNumber, length);  // copy the native array to java array  (\*env)->SetIntArrayRegion(env, javaArray, 0, length, randomNumber);  return javaArray;  } |

**ArrayMinMaxDiff.c**

|  |
| --- |
| #include <stdio.h>  #include <jni.h>  #include <math.h>  int findMaxNumber(jint theArray[], int length){  int max;  int index;  int counter;    max = theArray[0];  index = 0;  counter = 1;    for (int i = 1; i < length; i++) {  if (theArray[i] > max) {  index = i;  counter = 1;  max = theArray[i];  }  else if(theArray[i] == max){  counter++;  }  }  printf("The Maximum is %d, at Index %d, Occurs %d times\n", max, index, counter);  return max;  }  int findMinNumber(jint theArray[], int length){  int min;  int index;  int counter;    min = theArray[0];  index = 0;  counter = 1;    for (int i = 1; i < length; i++) {  if (theArray[i] < min) {  index = i;  counter = 1;  min = theArray[i];  }  else if(theArray[i] == min){  counter++;  }  }  printf("The Minimum is %d, at Index %d, Occurs %d times\n", min, index, counter);  return min;  }  JNIEXPORT int JNICALL Java\_ArrayGenerate\_arrayMinMaxDiff(JNIEnv \*env, jobject object, jintArray numArray){  int max;  int min;  // link the java array field, and get the array  jsize len = (\*env)->GetArrayLength(env, numArray);  jint \*numArr = (\*env)->GetIntArrayElements(env, numArray, NULL);  max = findMaxNumber(numArr, len);  min = findMinNumber(numArr, len);  (\*env)->ReleaseIntArrayElements(env, numArray, numArr, JNI\_ABORT);  return max - min;  } |

**ArrayMiscellaneous.c**

|  |
| --- |
| #include <stdio.h>  #include <jni.h>  void showAllNumber(jint numArr[], int len){  printf("All number in the array: ");  for(int i = 0; i < len; i++){  printf("%ld ", numArr[i]);  }  printf("\n");  }  void sortTheNumber(jint numArr[], int len){  int temp;  for(int i = 0; i < len; i++){  for(int j = i + 1; j < len; j++){  if(numArr[i] > numArr[j]){  temp = numArr[i];  numArr[i] = numArr[j];  numArr[j] =temp;  }  }  }  printf("After sort: ");  for(int i = 0; i < len; i++){  printf("%ld ", numArr[i]);  }  printf("\n");  }  void calculateTotalMean(jint numArr[], int len){  int total = 0;  for(int i = 0; i < len; i++){  total += numArr[i];  }  printf("Total of the array is %d\n", total);  printf("Mean of the array is %.2f\n", (float)total / (float)len);  }  JNIEXPORT void JNICALL Java\_ArrayGenerate\_arrayMiscellaneous(JNIEnv \*env, jobject object, jintArray numArray){  // link the java array field, and get the array  jsize len = (\*env)->GetArrayLength(env, numArray);  jint \*numArr = (\*env)->GetIntArrayElements(env, numArray, NULL);  showAllNumber(numArr, len);  sortTheNumber(numArr, len);  calculateTotalMean(numArr, len);  (\*env)->ReleaseIntArrayElements(env, numArray, numArr, JNI\_ABORT);  } |

## Print Screen of Outputs

### Question 1

Figure 5 shows the output of the ClassifyHonours.java. The program will first ask user to select their qualification, before asking for the CGPA. The program will then use these two data to calculate the classification of honours for the users. The classification of honours for diploma and degree is based on the Table 1 and Table 2 respectively.

After calculated the classification, the program will prompt user if they wish to continue to calculate the GPA will achieve to get the target CGPA. If user proceed, the program will request the user’s completed credit hours, current taking credit hours, and target CGPA to get for this trimester. After getting this information, the program will then calculate the GPA to get in this trimester in order to achieve the target CGPA.

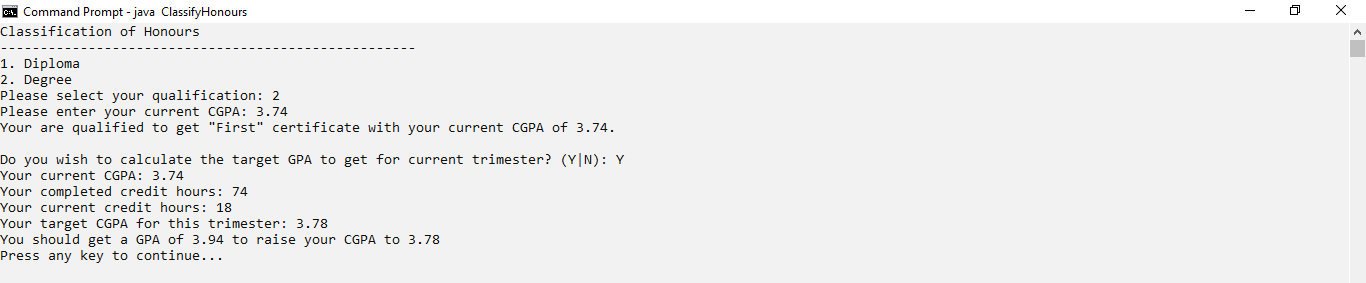


Figure 5 Output for ClassifyHonours.java

Table 1 Diploma Classification of Honours

|  |  |
| --- | --- |
| Classification of Honours | CGPA |
| Distinction | 3.50 – 4.00 |
| Credit | 3.00 – 3.49 |
| Pass | 2.00 – 2.99 |

Table 2 Degree Classification of Honours

|  |  |
| --- | --- |
| Classification of Honours | CGPA |
| First | 3.67 – 4.00 |
| Second Upper | 3.33 – 3.66 |
| Second Lower | 2.67 – 3.32 |
| Third | 2.00 – 2.66 |

### Question 2

Figure 6 shows the output for the ArrayGenerate.java. Once the program starts, it will ask the user to enter the size of the array to generate. After that, an array with the size of the input will be fill with random numbers from 1 to 100. The program will count if any of the elements in the array is less than 50 and display to the user. Besides that, it will find the maximum number and minimum number in the array, and find the index of its first occurrence, as well as the number of times that the numbers appear in the array. Hereafter, it will calculate the difference between maximum number and minimum number.

In additions, the program will also show all the elements in the array and then sort it accordingly. It will also calculate the sum of all elements in the array, as well as its mean number.

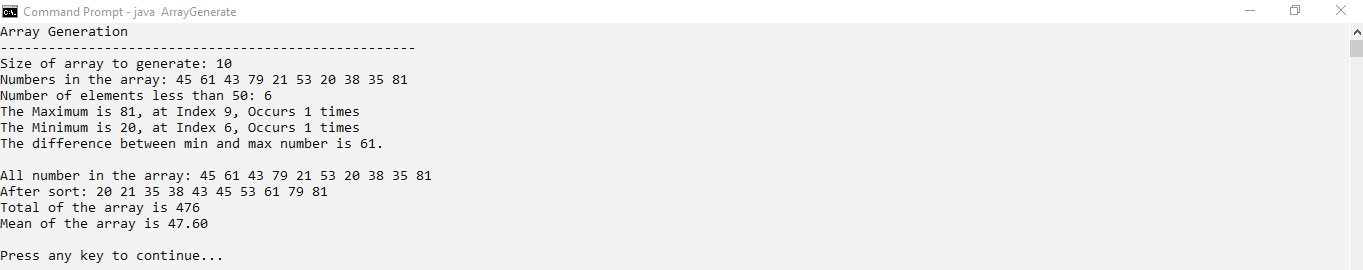


Figure 6 Output for ArrayGenerate.java