CS489 – Applied Software Development

Lab1a

(April 2024)

1. The estimated time allotted for completing the tasks is about 2 hours.
2. You are expected to use your Computer with an IDE or any Code Editor tool of your choice to implement your solution for the coding task(s).
3. *For the tasks/operations in the question where applicable, you are expected to take screenshot(s) of your work/result(s), save each into a .png or .jpg image file, placed inside a folder named, screenshots and include these in your submission, making sure to include all your project source code. When you have completed your solution, you must take each of the set of 3 evidential sample screenshots, included in the question/tasks.*
4. Upon completion, to submit your work for review and grading, simply commit and push your entire Project folder (including the screenshots folder) into your corresponding repository for this lab.

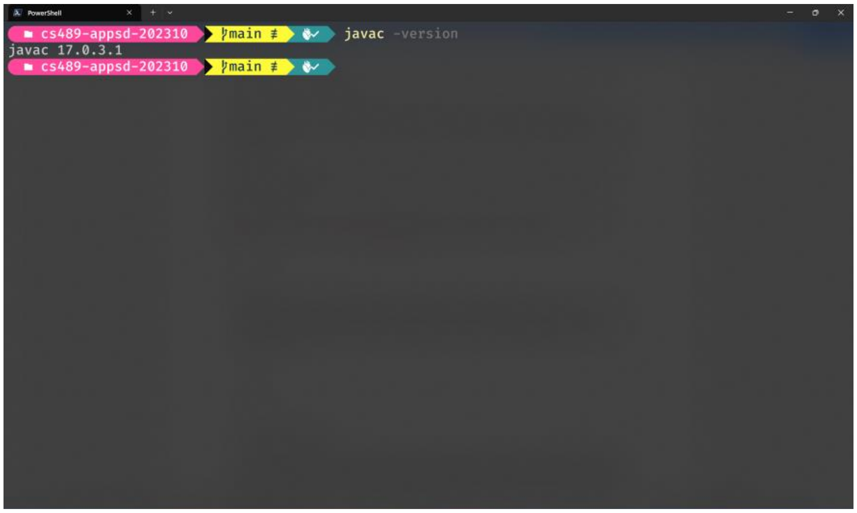
Make sure to include the screenshots of your work/results, as required.

Software Devt Environment Setup Tasks and Coding (10 points)

1. (10 points) **Setup and exercise your Dev Environment/Tools by implementing a basic CLI Application Project, and publish to Github**

   TASK 1:

   Obtain, install and configure the Java JDK for your OS. The minimum required version is JDK17. When done (or if already done), open a command terminal/shell and execute the command, javac -version. Take a screenshot of your computer, showing the result like the one below:



```
cs489-appsd-202310  main ≢  javac -version
javac 17.0.3.1
cs489-appsd-202310  main ≢
```

TASK 2:

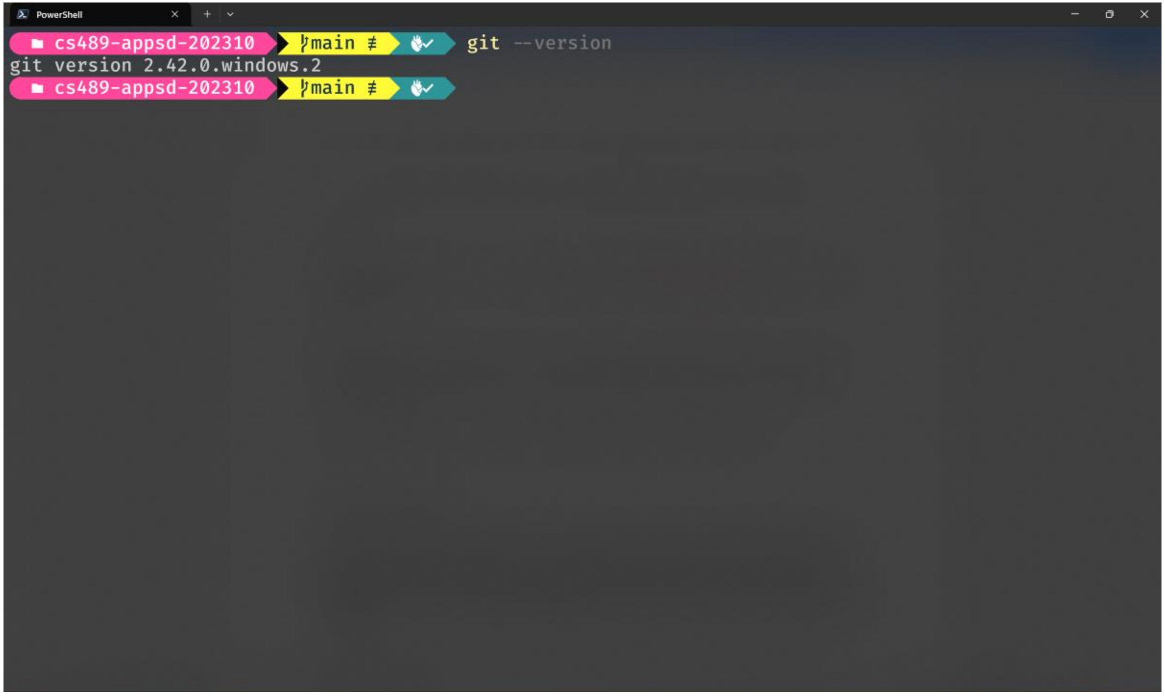
Obtain and install an IDE for Java application development. JetBrains IntelliJ IDEA is highly recommended.

TASK 3:

Obtain, install and configure Git for your OS. The minimum required version is 2.40.0. When done (or if already done), open a command terminal/shell and execute the command,

```
git --version.
```

Take a screenshot of your computer, showing the result like the one below:



```
cs489-appsd-202310  main ≢  ✓  git --version
git version 2.42.0.windows.2
cs489-appsd-202310  main ≢  ✓
```

TASK 4:

Problem Statement:

Assume that a National Groceries supply company has hired you to develop a Command-Line Interface (CLI) application for the Products Management System (PMS), which they will be using to manage data about the Products they supply.

Here are the attributes for the **Product** entity, including some useful descriptions and/or sample data values:

Product:

productId, (e.g. 3128874119, 2927458265, 9189927460 etc.)
Note: These productIds are numbers

name, (e.g. Banana, Apple, Carrot etc.)

dateSupplied, (e.g. 2023-01-24, 2022-12-09, 2023-03-31 etc.)

quantityInStock, (e.g. 124, 18, 89 etc.)

unitPrice, This is money (in dollars and cents) (e.g. $0.55, $1.09, $2.99 etc.)

Data:

Here is the company's existing data, which you are expected to load/use it your Program and print as output upon its execution:

Products data:

Product Id	Name	Date Supplied	Quantity In Stock	Unit Price (in US$)
3128874119	Banana	2023-01-24	124	0.55
2927458265	Apple	2022-12-09	18	1.09
9189927460	Carrot	2023-03-31	89	2.99

For this question, do the following:

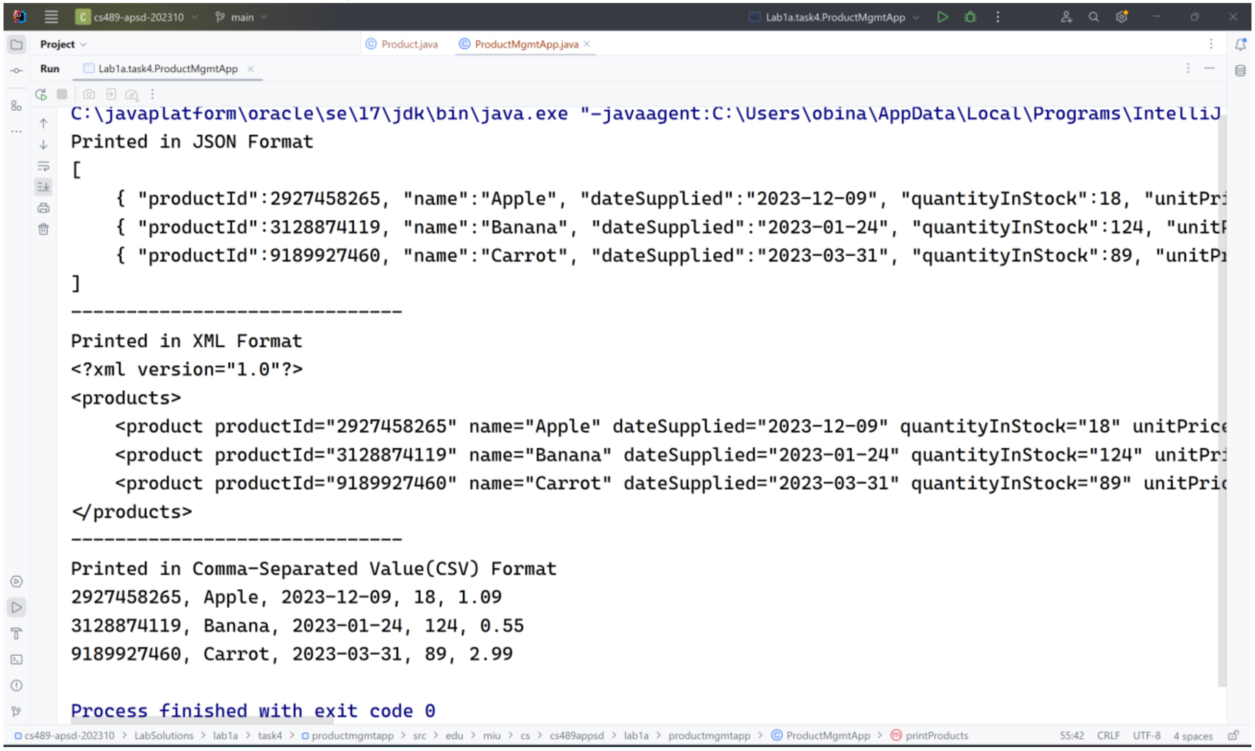
1. Create a new Command-Line Interface application project.
2. In the project, implement code for the class named, **Product**, including each of the data fields required by the specification above, add 3 constructors including the default constructor, and getter (accessor) and setter (mutator) methods. Make the class be inside a package/namespace named, "edu.miu.cs.cs489appsd.lab1a.productmgmtapp.model".
3. Add and code an executable class named, ProductMgmtApp, and in it include the main method. And, in the main method, create an array of Products using the company's data, as given above. Also, implement a method named, printProducts, that takes as input the array of products data and it prints them out to the console, sorted in ascending order by the product name, in each of the following formats: JSON, XML and CSV.

   Make the ProductMgmtApp class be inside a package named, "edu.miu.cs.cs489appsd.lab1a.productmgmtapp".

4. In the main method, add code to invoke your printProducts(…) method and execute your program, take a screenshot of your IDE showing the output/result.

Shown below is a sample screenshot and data presentation for the above requirements.

JSON-formatted list of all Products:
XML-formatted list of all Products:
and Comma-Separated Values (CSV)-formatted list of all Products
(Note: Sorted in ascending order of the Product Names)


cs489-apsd-202310
main
Lab1a.task4.ProductMgmtApp
Project
Product.java
ProductMgmtApp.java
Run
Lab1a.task4.ProductMgmtApp
cs489-apsd-202310 > LabSolutions > lab1a > task4 > productmgmtapp > src > edu > miu > cs > cs489appsd > lab1a > productmgmtapp > ProductMgmtApp > printProducts
55:42
CRLF
UTF-8
4 spaces

```
C:\javaplatform\oracle\se\17\jdk\bin\java.exe "-javaagent:C:\Users\obina\AppData\Local\Programs\IntelliJ
Printed in JSON Format
[
    { "productId":2927458265, "name":"Apple", "dateSupplied":"2023-12-09", "quantityInStock":18, "unitPri
    { "productId":3128874119, "name":"Banana", "dateSupplied":"2023-01-24", "quantityInStock":124, "unitP
    { "productId":9189927460, "name":"Carrot", "dateSupplied":"2023-03-31", "quantityInStock":89, "unitPr
]
------------------------------
Printed in XML Format
<?xml version="1.0"?>
<products>
    <product productId="2927458265" name="Apple" dateSupplied="2023-12-09" quantityInStock="18" unitPrice
    <product productId="3128874119" name="Banana" dateSupplied="2023-01-24" quantityInStock="124" unitPri
    <product productId="9189927460" name="Carrot" dateSupplied="2023-03-31" quantityInStock="89" unitPric
</products>
------------------------------
Printed in Comma-Separated Value(CSV) Format
2927458265, Apple, 2023-12-09, 18, 1.09
3128874119, Banana, 2023-01-24, 124, 0.55
9189927460, Carrot, 2023-03-31, 89, 2.99

Process finished with exit code 0
```

//-- The End --//