DAM. UNIT 1. ACCESS TO FILES. ASSESSABLE TASK 1

DAM. Acceso a Datos (ADA) (a distancia en inglés)

Unit 1. ACCESS TO FILES

Assessable Task 1

Abelardo Martínez

Year 2024-2025

Aspects to bear in mind

Important

> **If you look for the solutions surfing the Internet or asking the oracle of ChatGPT you will be fooling yourself**. Keep in mind that **ChatGPT is not infallible or all-powerful**.
>
> It is a great tool to speed up your work once you have mastered a subject, but using it as a shortcut when acquiring basic skills and knowledge seriously undermines your learning. If you use it to get solutions or advice on your own, check the proposed solutions carefully as well. Try to solve the activities using the resources we have seen and the extended documentation you will find in the "Virtual Classroom".

Tips for programming

We advice to follow the next coding standards:

- One instruction per line.
- Add comments to make your code clearer and more readable.
- Use the Hungarian notation to recognise the type of variables at first sight.
- If necessary, we strongly recommend using buffer-based solutions.
- Remember that there are several ways to implement a solution, so choose the one you like best.

A. Instructions and guidelines

> **The project MUST be carried out in Java**. **Other technologies -such as Spring Boot- will not be supported**. Any of the IDEs proposed in unit 1 can be used for its development, although **Eclipse is strongly recommended**.

1. OVERVIEW

You are required to create a Java application **on your own** that utilises concepts taught during UNIT 1 to meet a provided specification.

2.TIMELINE AND EXPECTATIONS

- **Percentages within the TERM**: 50% of TERM total (AT2 would make the other 50%)
- **Percentages within the TASK**: 100% ADA skills (English skills must be PASSED).
- **Due/Deadline**: **11:59pm on Sunday, 3rd November, 2024** (3 WEEKS)

3. GRADING

You must get 5 marks out of 10 in ADA and a PASSED in English to pass this ASSESSABLE TASK.

A detailed grading scale will be providing with this document (check LEARNING RUBRIC).

4.RESOURCES

You should make a comprehensive reading of all the materials provided by your teacher as well as the non-assessable tasks, but also dive the Internet to find examples which provide similar outcomes to the ones required by this task.

Feel free to copy & paste code from ANY resource as long as you understand every piece of it since you will be required to defend your work in an individual meeting.

5. PLAGIARISM

You must not allow other students to copy your work and must take care to safeguard against this happening.

In case of suspected plagiarism, an additional oral interview might be required.

6. HANDING AND FEEDBACK

- **The task will be delivered ONLY in a ZIP format file, compressing the project folder from your IDE** (i.e. Eclipse). The ZIP file should be named "**at1_yourname.zip**", where the suffix refers to your name.
- Afterwards, **you WILL BE REQUIRED to attend an oral interview** with your teacher to

discuss certain aspects of your task in English for a maximum of 15 minutes.

- You will receive your marks broken down by each criteria, and the total, together with any comments giving suggestions on how you could have done better.

B. Assessment details

> **ONLY ENGLISH IS ALLOWED** for the implementation of the assessable task, both comments and explanatory/clarifying texts.

1. **EVERY METHOD MUST BE PROPERLY DESCRIBED IN YOUR OWN WORDS**. At the beginning of each method you must add comments to explain in your own words how it works.
2. **ALSO, YOU MUST ADD A TEXT EXPLAINING IN YOUR OWN WORDS, YOUR EXPERIENCE IMPLEMENTING THIS SOLUTION**.

Create a text file called "**comments**" (in txt or pdf format) and copy it into the project folder or create the text file within the project itself in the Eclipse IDE.

- **PARAGRAPH 1**. Describe briefly the solution provided.
- **PARAGRAPH 2**. Describe briefly the difficulties found.
- **PARAGRAPH 3**. Describe briefly several possible extensions you recommended.

B.1. Mandatory features

Activity (ASSESSABLE)

Create a program in Java to manage TAX INSPECTORS in a tax management by printing and using a specific menu. After each option, the user should see the same menu until option zero is pressed.

ATTENTION: Use the proper exceptions when accessing to files.

Menu options:

- **Press 0 to "Exit"**
- **Press 1 to "Get tax inspectors and commissions (to CSV & XML)"**
  - For every TAX INSPECTOR we need inspector ID (String without spaces), firstname (String with spaces), lastname (String with spaces), birthday (LocalDate in format **dd/MM/yyyy**), commission (Float) and cheated taxpayers (Integer), added to an ArrayList of TAX INSPECTORS. Every tax inspector has a commission, which means what it charges for fleecing the long-suffering taxpayers.
  - **Check if the tax inspector ID already exists in the array list**. If yes, you must display a message on the screen. You must ask for each value (in loop) until the user enters a valid ID.
  - Once zero is entered as ID, all TAX INSPECTORS will be saved both in a CSV and XML file, with a proper format, overwriting all the whole files if exist. Before saving, you should check if the ArrayList of TAX INSPECTORS is empty to avoid executing unnecessary code.
  - **ATTENTION**: For every TAX INSPECTOR, you must ask for each value (in loop) until the user enters a valid data.
  - **ATTENTION**: For every TAX INSPECTOR, the **commission percentage** shall be **between 0 and 30**.
  - Call the **CSV file**: taxinspectors.csv
  - Call the **XML file**: taxinspectors.xml

- **Press 2 to "List all tax inspectors stored (using DOM)"**
  - Just read the XML file and print every TAX INSPECTOR information. **You may use DOM**.
- **Press 3 to "Generate HTML with all tax inspectors via XSL"**
  - Editing and adapting the XSL template you can find in the section "Suggested XSL pattern" or any other XSL you can find on the Internet, you have to read the XML file and generate an HTML using the XSL.
  - **ATTENTION**: The header table must have a background color different from white.
  - **Call the HTML file**: taxinspectors.html
  - **Call the XSL file**: taxinspectors.xsl

Menu example:

```
*****
MENU
*****
================================================================================
  0. Exit
  1. Get tax inspectors and commissions (to CSV & XML)
  2. List all tax inspectors
  3. Generate HTML with all tax inspectors via XSL
  4. [optional] Modify the commission of a tax inspector
  5. [optional] Generate HTML with the meanest tax inspectors via XSL
================================================================================
    Select an option:
```

Here you have a SUGGESTION of how your CSV could look like:

```
TaxinspectorID, First name, Last name, Birthday, Comm percentage, Commission, Cheated taxpayers
insp01, Rafa, Teloquitotodo, 05/05/1976, 10.0, 4425.67, 125
insp02, Pepe, Pirata, 19/01/1970, 15.7, 7899.0, 250
insp03, Paco, Montoro, 04/07/1965, 18.0, 12432.21, 335
```

Here you have a SUGGESTION of how your XML could look like:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Taxinspectors>
     <Taxinspector>
          <id>insp01</id>
          <firstname>Rafa</firstname>
          <lastname>Tequitotodo</lastname>
          <birthday>05/05/1976</birthday>
          <commission percentage="10.0">4435.67</commission>
          <cheatedtaxpayers>125</cheatedtaxpayers>
     </Taxinspector>
     <Taxinspector>
          <id>insp02</id>
          <firstname>Pepe</firstname>
          <lastname>Pirata</lastname>
          <birthday>19/01/1970</birthday>
          <commission percentage="15.7">7899.0</commission>
          <cheatedtaxpayers>250</cheatedtaxpayers>
     </Taxinspector>
     <Taxinspector>
          <id>insp03</id>
          <firstname>Paco</firstname>
          <lastname>Montoro</lastname>
          <birthday>04/07/1965</birthday>
          <commission percentage="18.0">12432.21</commission>
          <cheatedtaxpayers>335</cheatedtaxpayers>
     </Taxinspector>
</Taxinspectors>
```

B.2. Optional features

Activity (ASSESSABLE)

Optionally, you can implement these following entries within the menu to reach more than 8 marks out of 10 at this ASSESSABLE TASK.

ATTENTION: Use the proper exceptions when accessing to files.

Menu options:

- **Press 4 to "[optional] Modify the commission of a tax inspector"**
  - The Ministry of Finance periodically reviews the effectiveness of its inspectors to see whether they have been successful in achieving their goals of fleecing as many taxpayers as possible. If they have been effective, their commission rate will be increased, but if they have not, then it will be decreased to motivate them to steal from the honest citizen.
  - After asking for the tax inspector ID and the new commission percentage, we walk through the DOM and change the commission percentage for the tax inspector whose ID matches the former introduced ID. In addition, we will update the commission percentage.
  - Then we (over)write the XML again.
- **Press 5 to "[optional] Generate HTML with the meanest tax inspectors via XSL"**
  - A tax inspector shall be considered mean if he/she has cheated 200 taxpayers at least.
  - Read the CSV and **create another XML** with the meanest tax inspectors. Call the XML file: **despicables.xml**
  - Create another XSL from the existing one, changing the font foreground colour to orange when listing the tax inspectors.
  - Then, apply this new XSL to get another HTML:
    - **Call the HTML file**: despicables.html
    - **Call the XSL file**: despicables.xsl

Suggested XSL pattern

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    version="1.0">
    <xsl:template match="/">
        <html>
            <head>
                <style type="text/css">
                    table.tfmt {
                    border: 2px ;
                    }

                    td.colfmt {
                    border: 1px ;
                    background-color: white;
                    color: black;
                    text-align:center;
                    }

                    th {
                    background-color: white;
                    color: black;
                    }
                </style>
            </head>
            <body>
                <table class="tfmt">
                    <tr>
                        <th style="width:100px">Tax inspector</th>
                        <th style="width:110px">First name</th>
                        <th style="width:175px">Last name</th>
                        <th style="width:100px">Birthday</th>
                        <th style="width:100px">% Comm</th>
                        <th style="width:110px">Commission</th>
                        <th style="width:100px">Cheated taxpayers</th>
                    </tr>
                    <xsl:for-each select="Taxinspectors/Taxinspector">
                    <tr>
                            <td class="colfmt">
```

```
                            <xsl:value-of select="id" />
                        </td>
                        <td class="colfmt">
                            <xsl:value-of select="firstname" />
                        </td>
                        <td class="colfmt">
                            <xsl:value-of select="lastname" />
                        </td>
                        <td class="colfmt">
                            <xsl:value-of select="birthday" />
                        </td>
                        <td class="colfmt">
                            <xsl:value-of select="commission/@percentage" />
                        </td>
                        <td class="colfmt">
                            <xsl:value-of select="commission" />
                        </td>
                        <td class="colfmt">
                            <xsl:value-of select="cheatedtaxpayers" />
                        </td>
                    </tr>
                </xsl:for-each>
            </table>
        </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

C. Learning Rubric

C.1. ADA skills

Minimum of 5 out of 10 required for this part.

These marks will be invalidated (mark 4) if you fail to defend your work in an oral interview.

ASSESSMENT ITEMS	ASSESSMENT ITEM DETAILS	SCORE (POINTS)
Classes and methods	Classes and methods are structured properly	0.75
Menu	The menu complies with the specifications	0.75
Get tax inspectors and commissions (to CSV & XML)	The ArrayList is populated properly. The date format is correct	2.5
	The CSV & XML are generated properly. The date format is correct	
List tax inspectors	Reads the XML properly (using DOM)	1.5
	Prints the data in a proper way. The date format is correct	
Generate HTML with all tax inspectors via XSL	Reads the information from the XML	2.5
	Generates the HTML properly from a XSL. The date format is correct	
[optional] Modify the commission of a tax inspector		1
[optional] Generate HTML with the meanest tax inspectors via XSL. The date format is correct		1

C.2. English skills

Mandatory to be COMPETENT to pass this part.

ASSESSMENT ITEMS	ASSESSMENT ITEM DETAILS	SCORE
Writing skills	Every method is described properly	COMPETENT/NOT COMPETENT
	A proper text is provided (within the code or in a text file) to describe the AT using THREE PARAGRAPHS	COMPETENT/NOT COMPETENT
Oral skills	Uses a vocabulary appropriate for the purpose	COMPETENT/NOT COMPETENT
	Shows fluency and confidence	COMPETENT/NOT COMPETENT
Comprehension skills		Accomplished since all materials are in English
Reading skills		Accomplished since all materials are in English