

**SCHOOL OF COMPUTER SCIENCE AND INFORMATICS
COURSEWORK ASSESSMENT PROFORMA**

MODULE & LECTURER: DR Y LAI and DR M MORGAN

DATE SET: Thursday 1st March 2018

SUBMISSION DATE: Friday 27th April 2018 @ 9:30am

SUBMISSION ARRANGEMENTS:

Hand in Date: Submission of all material (Short Report and Java code) by 9:30am on Friday 27th April 2018 (Week 10, Semester 2) via Learning Central.

TITLE: Simple Weather Data Viewer using JavaFX

This coursework is worth 30% of the total marks available for this module. The penalty for late or non-submission is an award of zero marks. You are reminded of the need to comply with Cardiff University's Student Guide to Academic Integrity. Your work should be submitted using the official Coursework Submission Cover sheet.

INSTRUCTIONS

The individual project assignment involves implementing a Simple Weather Data Viewer using JavaFX. You must supply a typeset report in PDF format. A short (about 4-6 pages) written description conveying all the appropriate information to demonstrate its operation and explain your programming philosophy is all that is required. You should also submit all Java source code, which should be zipped into a single file for submission. For complete details see the following sheets.

SUBMISSION INSTRUCTIONS

Your solution must include:

- A (standard) cover page that details: Student No., Module Code, Module Title, Coursework Title, Lecturer(s), Hours Spent on this Exercise, (Special Provision if applicable).

- A typeset PDF report (approximately 1000 words/roughly 3-4 pages of text; diagrams, screenshots etc. are encouraged) detailing the following:
 - An overview of your program design and implementation.
 - A basic algorithmic description of the main elements of your solution and how they satisfy the basic requirement listed below. You can use any suitable way to describe your solution, for example using pseudocode, UML diagrams, text description and/or screen snapshots etc.
 - A brief user guide describing how major features can be used.

You should explicitly highlight any novel features — i.e. those features developed beyond the basic requirements.

- A copy of all Java source code.
 - Include all Java source files, and related files (e.g. dependent third-party libraries and graphics)
 - Java files should be submitted as a single zip file collection.

Ensure that your student number is in each file that makes up your submission.

CRITERIA FOR ASSESSMENT

Credit will be awarded against the following criteria.

- A Pass mark (50-60%) will be awarded for solutions that satisfy most of the main points of the basic requirements (specified below).
- Good marks (60-70%) will be awarded for solutions that satisfy all the main points of the basic requirements thoroughly, with some justification of design philosophy (specified below).
- High marks (over 70%) will be awarded for solutions that provide thorough solutions to addressing the basic requirements and also some novel extensions/additional features, with a good justification of design philosophy.

- Solutions that do not satisfy all the requirements given in the coursework will get lower marks.

Specifically, the marks will be apportioned as follow:

- 50 marks are available for your solution to a working functional system with suitable JavaFX implementations of basic exercise criteria.
- 20 marks are available for a clear write up. The write up should clearly state your design and implementation strategy.
- 30 marks are available for your design and incorporation of other features beyond the basic coursework specification.

FURTHER DETAILS

- Feedback on your coursework will address the above criteria.
- The Individual project work will be returned via Grade Centre within 3 working weeks of submission. Individual feedback will be given via Grade Centre.
- This will be supplemented with oral feedback in the revision lecture in Week 12.

Individual Project: Simple Weather Data Viewer using JavaFX

You will need to download the file CMT205CWDATA.zip, from the CMT205 module on Learning Central, which you will find under the 'Assessment' link on the left-hand menu. This zip archive file contains historical monthly weather data, covering each month in the period from 2009 to 2017, for 34 UK meteorological stations (NOTE: You will need to use appropriate software to extract these files from the zip archive). The uncompressed archive contains 34 CSV (Comma-Separated Values) files, each relating to a specific meteorological station. The list of meteorological stations is as follows:

Aberporth, Armagh, Ballypatrick Forest, Bradford, Braemar, Camborne, Cambridge NIAB, Cardiff Bute Park, Chivenor, Dunstaffnage, Durham, Eastbourne, Eskdalemuir, Heathrow, Hurn, Lerwick, Leuchars, Lowestoft, Manston, Nairn, Newton Rigg, Oxford, Paisley, Ross-on-Wye, Shawbury, Sheffield, Stornoway Airport, Sutton Bonington, Tiree, Valley, Waddington, Whitby, Wick Airport, Yeovilton

The relevant weather data is stored in [Meteorological Station].csv. For instance, Aberporth.csv stores the weather data for the Aberporth station.

Each CSV file is structured with the following fields, separated by commas:

year,month,tmax,tmin,af,rain

The following table provides an explanation of each field:

year	The year for the weather data
month	The month for the weather data
tmax	The mean maximum temperature in the month
tmin	The mean minimum temperature in the month
af	The number of days of air frost in the month
rain	The total rainfall in the month

Coursework Tasks

You are required to create a Java application for interactively exploring the historical weather data contained in the given data files using JavaFX.

The following basic requirements should be met in order to gain average to good marks:

1. The application should read the data stored within all CSV files and present the data in a suitable format, using a grid, with the following information:

Meteorological Station

as well as statistics from the last year (2017):

Highest monthly mean maximum temperature (tmax)

Lowest monthly mean minimum temperature (tmin)

Total air frost days

Total rainfall

As a basic requirement, the list of meteorological stations and CSV files can be hard-coded in your program.

2. The application should allow an individual meteorological station to be selected, which provides the user with a more detailed view in a separate tab. This view should present the user with historical data pertaining to that station and include suitable basic charts to present the data, e.g. bar graph.
3. The application should also allow users to generate a report containing a summary of key facts for all the meteorological stations. The report should be a single text file containing the following information for each station:

Number: <sequence number>

Station: <station name>

Highest: <month/year with the highest tmax>

Lowest: <month/year with the lowest tmin>

Average annual af: <average days of air frost per year>

Average annual rainfall: <average annual rainfall>

In the above, texts in the angular brackets (<>) are placeholders and should be replaced with the actual statistics calculated based on all the available data of the station. The sequence number is a natural number assigned to each station in the order they appear in the report (i.e. 1, 2, 3, 4, ...).

4. You should develop an appropriate GUI for your application, using JavaFX.

In order to gain higher marks you need to add some novel extensions or additional features. You need only provide two further different novel extensions. Here are a few suggestions.

- Advanced GUI features
- Advanced formatting of the report
- Advanced visualisation of weather data

There are endless possibilities here and you are encouraged to think of your own extensions.