Dublin Business School
Assessment Brief

Assessment Details

Module Title:	Advance Programming Techniques
Module Code:	B9CY108
Module Leader:	Swati Dongre
Stage (if relevant):	NA
Assessment Title:	CA_ONE_(30%)
Assessment Number (if relevant):	ONE
Assessment Type:	Continuous Assessment
Restrictions on Time/Length:	NA
Individual/Group:	Individual
Assessment Weighting:	30%
Issue Date:	Oct. 16, 2025
Hand In Date:	Dec. 13, 2025
Planned Feedback Date:	Dec. 18, 2025
Mode of Submission:	Moodle Only

Requirements:

- You are required to implement part-I in C# and part-II in Python.
- Que.1 & 2 - Create two separate projects for Que.1 and Que.2. Place them in separate folders. Have screenshots of the results in the report and respective folders.
- Que.3 & Que.4 - One File per problem is required – DO NOT SUBMIT ONE PYTHON FILE FOR ALL PROBLEMS. File names should be indicative e.g. Que4.py, Que3_server.py, Que3_client.py and so on. Have screenshots of database and results in the respective folders and the report.
- Prepare a report (max. 700 words) that explains your work and submit .pdf file. Make sure you attach the Electronic Assignment Cover Sheet (provided with this document, pls. refer to page.no. 9) Also ensure that all external help and resources are attributed in detail. General references to w3school, friends are not acceptable.
- Please keep your folder organized. Name your folder by your student-id. Have all subfolders and report in this main folder.
- Submission: Upload your work using the CA_ONE_(30%) link
- **IMPORTANT NOTE:** Only One submission is allowed so please ensure that you have included all required files in the zip before uploading to Moodle.
- Ensure you have all progress recorded on GitHub – multiple updates per day are expected. Please include the GitHub repository link in your report.

The following table illustrates the distribution of marks.

Total marks: 100

CA PARTS	PROBLEM	DETAILS	BREAKDOWN OF MARKS
Part I	Question1	C# Code	30 Marks
	Question2	C# Code	20 Marks
Part II	Question3	Python Code	30 Marks

		(server.py, client.py)	
	Question4	Python Code (code.py, info1.csv etc.)	20 Marks

Assessment Task

CA_ONE – part I : Programming in C#

Que. 1.

Develop a phone contact book that stores contact information. A sample contact information is shown as below:

First Name: Emily
Last Name: Blackwell
Company: Dublin Business School
Mobile Number: 0871111111
Email: emily.blackwell@dbs.ie
Birthdate: 1 Jan 1990

Users can create several contacts with their details, update the information if required and delete a contact when it's no longer needed.

Write a menu driven console application in C# to implement the contact book.

- -

Main Menu
1: Add Contact
2: Show All Contacts
3: Show Contact Details
4: Update Contact
5: Delete Contact
0: Exit

- -

Create a class ContactBook that contains data and methods providing the functionalities as mentioned above. The contact book should have min.20 contacts. (min.20 objects of the class 'ContactBook' to store 20 distinct persons). Make use of suitable data structure. The code should accept a non-zero, 9-digit, positive number as mobile number and reject all invalid data gracefully.
Your C# code should exhibit the following Object-Oriented principles:
Classes & Objects, Encapsulation, Properties, Access Modifiers, Method Overloading, Object Relationships.

(Marks 30)

Que. 2. Write a C# program that implements a system, which assists users by providing information about file extension. (e.g. .mp4, .mov, .avi, .mkv, .webm etc.) Your program should support inquiries for min 20 file extensions of your choice. Make use of suitable data structures in your code. Program should handle unexpected enquiries and respond gracefully.

(Marks 20)

Que. 3. Consider the following scenario, design and implement a solution to complete the tasks:

Students apply for admission to DBS for the following courses: MSc in Cyber Security, MSc Information Systems & computing, MSc Data Analytics.

Applicants are required to submit the following information:

- Name
- Address
- Educational qualifications
- The course they wish to enroll in
- Intended start year and month

Upon receiving the application, DBS assigns a unique application number for reference in all future correspondence.

Tasks:

i. Develop a console-based client application that accepts information from applicant.

ii. Implement a server that receives the applicant's information from client and stores it in a database. (Make use of disk persistent relational database, or document database)

iii. The server then generates a unique registration number for the user and sends it back to the client to provide to the user.

Implement the client-server application using connectionless protocol- UDP or connection-oriented protocol - TCP. Ensure that security factors are taken into consideration.

(Submit Que3_server.py, Que3_client.py, screenshots showing successful connection established between the client and the server and data transmission)

(Marks 30)

Que. 4.

Develop and implement a system that extracts and consolidates hotel room price data from two chosen establishments in a popular holiday destination during a specified seasonal period (e.g., 20–30 December). The system should gather information for at least ten rooms and present the data in a structured format, enabling prospective customers to perform comparative analysis and make informed accommodation decisions. It is essential to ensure that the data sources selected explicitly authorize the legal extraction of the required content.

a. Write a Python code to scrape the webpages for the required data.
b. Store the data in csv file.
c. At the end of the program, retrieve all data from csv file and display in the terminal window.

Use the appropriate Python modules to scrape the webpage and work with csv file. Submit the resulting csv file.

(Marks 20)

Assessment criteria

Minimum Intended Module Learning Outcomes – 2 and 3

Assessment criteria

Criteria/ Mark	< 40	40 – 49	50 - 59	60 – 69	70 +
Programming Project	Insufficient code or incomplete project	Some but barely sufficient and poorly written code with which partially solve the problem	Sufficient solves problem but lack of attention to best programming practices	Well-structured code with good interface design and functions	Excellent solution to problem – very well written code and interface with several functions

Group Management[1]

Refer to the College's Group Work Policy, QAH B.5.2.2.2.

Group Sizes:	-NA-
Group allocation:	SELECT 1
Grading approach:	SELECT 1
Alternative assessment:	Learners with PMCs that require an alternative assessment must log a ticket for consideration.

Resources

Software or tools required/ useful: -
Templates, files, data tables provided by the lecturer: -

Generative Artificial Intelligence Assessment Scale[2]

Can generative AI be utilised in this assignment? 02

2
AI-ASSISTED IDEA GENERATION AND

STRUCTURING
AI can be used in the assessment for generating input data for Que1, Que2 and Que3 **No AI content is allowed in the final submission.**

General Requirements for Students:

1. A proportion of assessment marks is allocated to presentation. All assignments must be word-processed, with word count noted unless otherwise stated by the lecturer.

2. Where a hardcopy submission is required, an Assignment Submission Form must be securely attached to each submission.

3. All assignments must be submitted no later than the stated deadline.

   Any CA must be submitted through Moodle by the specified submission deadline, unless an IT issue has precluded this. Any IT issue must be documented and notified to the Programme Coordinator in advance of the submission date.
   A penalty of two marks per day will be applied per day or part thereof for an ongoing failure to submit beyond the submission deadline. For the purposes of these penalties, a day is defined as any day of the week, including weekends and public holidays when the College may be closed. An examiner has the right to refuse to mark the assessment if the submission instructions have not been observed.
   - Where a late assessment, submitted within 14 days of the deadline, is of a passing standard (i.e. would achieve more than 40% in the normal scheme of things), the late penalty is capped such that the minimum grade that can be awarded is 40% for the late submission.
   - Where a late assessment is submitted more than 14 days after the deadline, it will receive 0%. The lecturer may, at their discretion, review the submission for feedback.

   Where an assessment is undertaken in a group, the piece of work should be submitted in its entirety, and any penalty for late submission incurred applies to all group members. Any learner who becomes aware that a group deadline will not be achieved through a lack of participation of another group member, should make this clearly known to the examiner in advance of the deadline.
4. All relevant provisions of the Assessment Regulations must be complied with.

5. Extensions to assignment submission deadlines will be not be granted, other than in exceptional circumstances. To apply for an extension please go to http://www.dbs-students.com/Registrar/ and download the Assignment Extension Request Form. *Once completed this form should then be returned to your Programme Leader, for approval.*

6. Students are required to retain a copy of each assignment submitted, and the submission receipt (If a physical submission is required) until the issuing of a transcript indicating the mark awarded.

7. Assignments must be appropriately packaged and presented.

8. Where a submission involves digital media, it is the submitting students' responsibility to ensure the media is appropriately labelled, fully working and they must retain a copy.

9. Assignments that *breach* the word count requirements will be penalised. *There is a 10% discretion, either way, applicable in terms of word count.*

10. Students are required to refer to the assessment regulations in their Student Guides and on the Student Website.

11. Dublin Business School penalises students who engage in academic impropriety (i.e. plagiarism, collusion and/or copying).

12. To prevent plagiarism please follow this link to the Harvard/OSCALA (delete as appropriate) Style Referencing Guide - all referencing is required in this format.

http://issuu.com/dbslibrary/docs/harvard-referencing-guide/1?mode=a_p

(Guide on referencing is also available under DBS library guides at www.library.dbs.ie)

13. In relation to electronic submissions:

A. All assignments should be submitted to your subject/course page on Moodle by the deadline date.
B. It is the student's responsibility to ensure their file is uploaded correctly.
C. When an assignment is submitted, it is the student's responsibility to ensure that the file is in the correct format and opens correctly.
D. When you submit your assignment you will be asked to click on a button which will declare the following:

By submitting this assignment I confirm that I am aware of DBS's policy regarding cheating, plagiarism and all other forms of academic impropriety. The coursework submitted is my own or my group's work, and all other sources consulted have been appropriately acknowledged. I am aware that in the case of doubt an investigation will be held.

E. Include an electronic **cover sheet** with the following details to the front of the assignment:

Electronic Assignment Cover sheet

Please fill out and attach as the first page of Assignment.

Student (s) Number as per your student card:

_____ _____

_____ _____

_____ _____

Course Title: M.Sc. Cyber Security

Lecturer Name: Swati Dongre

Module/Subject Title: Advanced Programming Techniques

Assignment Title: CA_ONE_(30%)

No of Words:

Note: Technical support is available to student between **09.30- 17.00 hrs only**. There is no technical support after 1700 hrs. It is your responsibility to ensure that you allow time to troubleshoot any technical difficulties by uploading early on the due date.