**A231 SQIT 3073:**

**Assignment 1: Housing Loan Eligibility and DSR Calculator (10%)**

Develop a Python program that calculates the monthly instalment for a housing loan and determines the loan applicant's eligibility based on their Debt Service Ratio (DSR).

**Requirements:**

* Your program should be able to prompt the user for the principal loan amount, annual interest rate, loan term (in years), applicant's monthly income, and any other monthly financial commitments. – Key in
* Implement a function to calculate the monthly instalment for the housing loan. – Formula (Fix + Flow)
* Implement a function to calculate the total amount payable over the term of the loan. – Formula (Fix + Flow)
* Implement a function to calculate the Debt Service Ratio (DSR) for the applicant. Assume the threshold for DSR is 70%. The DSR is calculated as the sum of all monthly debt commitments (including the housing loan) divided by the applicant's monthly income.
* The program should inform the applicant of their monthly instalment, total payment over the loan term, and whether they are eligible for the loan based on their DSR.
* Include a menu system with options to calculate a new loan, display all previous loan calculations, and exit the program.
* Store each loan calculation's details in a list so that the user can view previous calculations.

**Optional Enhancements:**

* Implement data validation to ensure that numerical inputs are valid and within reasonable bounds.
* Allow the user to delete a previous calculation or modify the DSR threshold.

**Deliverables:**

* The source code for the program that was uploaded to Github.
* A report documenting your code process flow, and the link to the Source code in PDF format. Maximum page is 4 pages. Name file : *name*\_*matrixno*\_SQIT3073\_A231.pdf

**Assessment Criteria:**

* Correctness and completeness of the program.
* Code readability and use of comments.
* Quality of the report.

**Submission Deadline:**

21 November 2023 (23:59)  **(No extension will be given)**

**Notes to Students:**

* Remember to test your program with a variety of inputs to ensure it behaves as expected.
* Use functions to organise your code and avoid redundancy.
* Ensure your program adheres to the principles of user-friendly design, providing clear prompts and handling user input errors gracefully.