PSU
PANGASINAN STATE UNIVERSITY
Region's Premier University of Choice

ELEC 4
SPECIAL TOPICS ON WEB AND MOBILE 2

PRESENTED TO

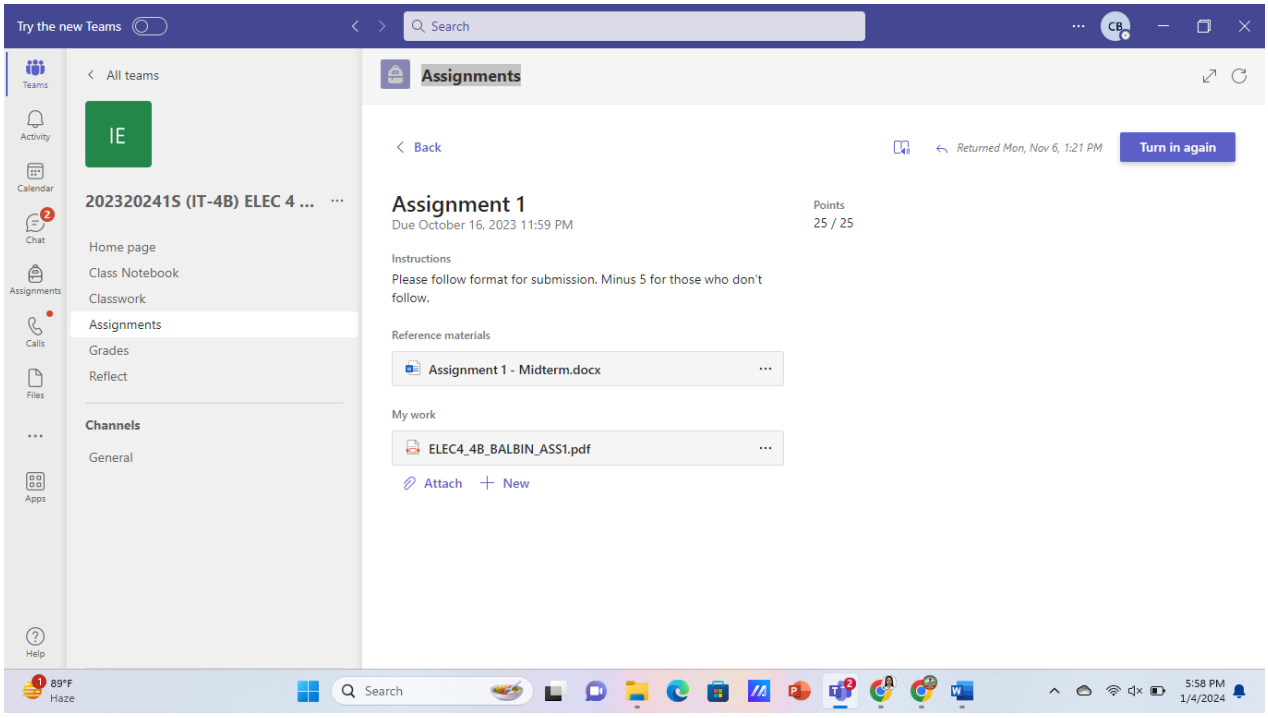
Mark Denver Adora

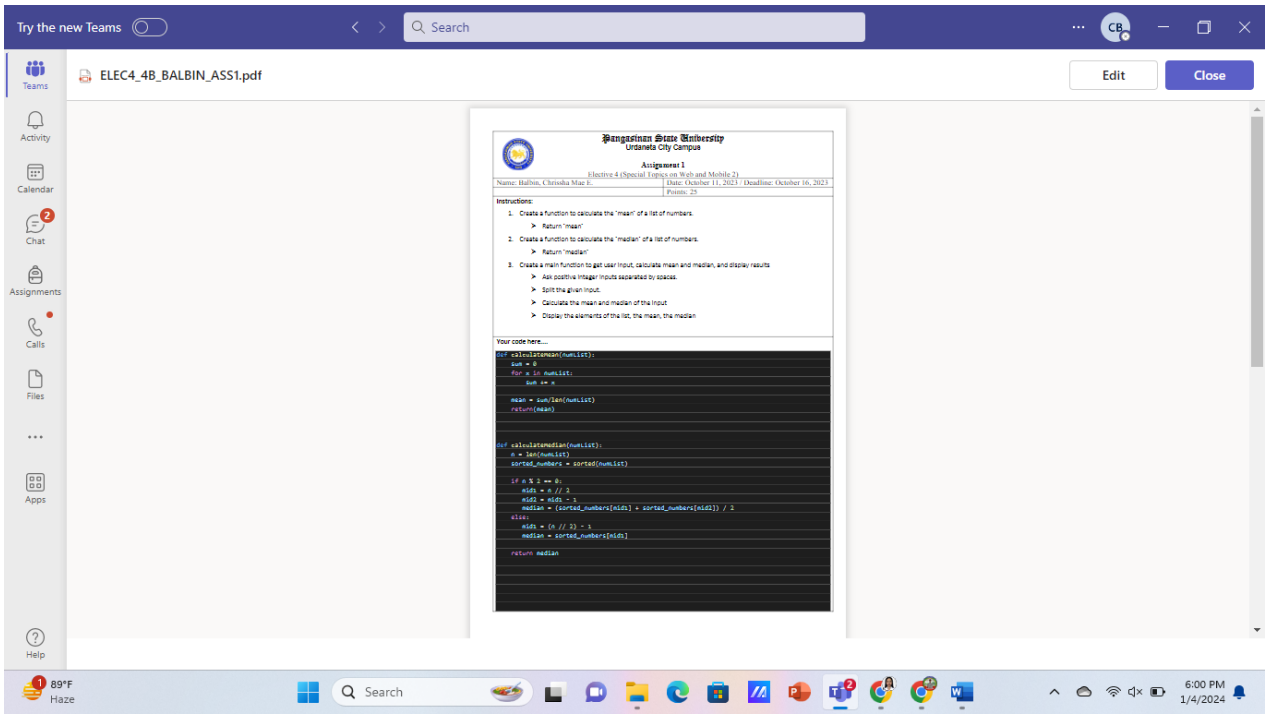
PRESENTED BY

Chrissha Mae E. Balbin

TABLE OF CONTENTS

ASSIGNMENT 1

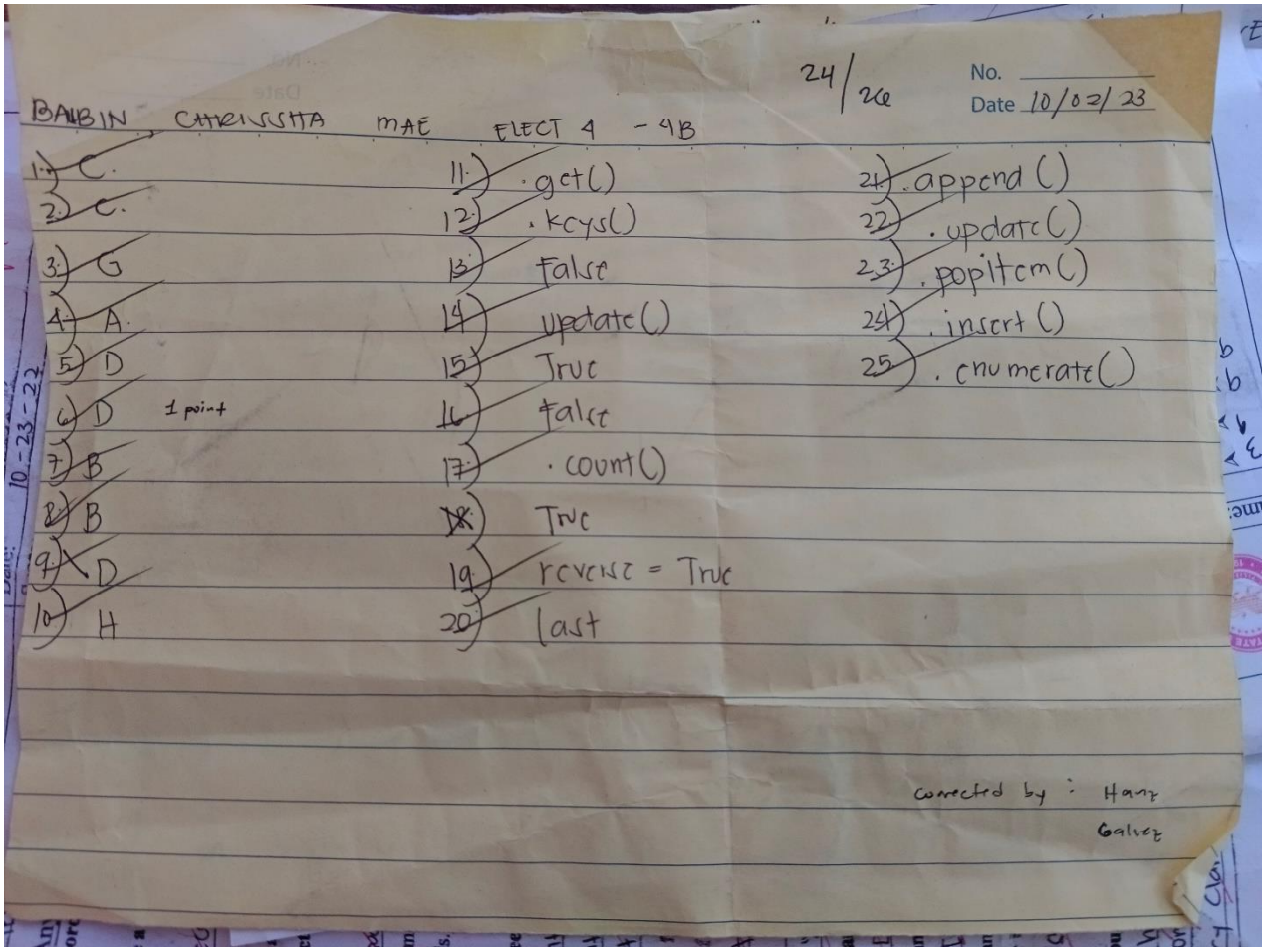




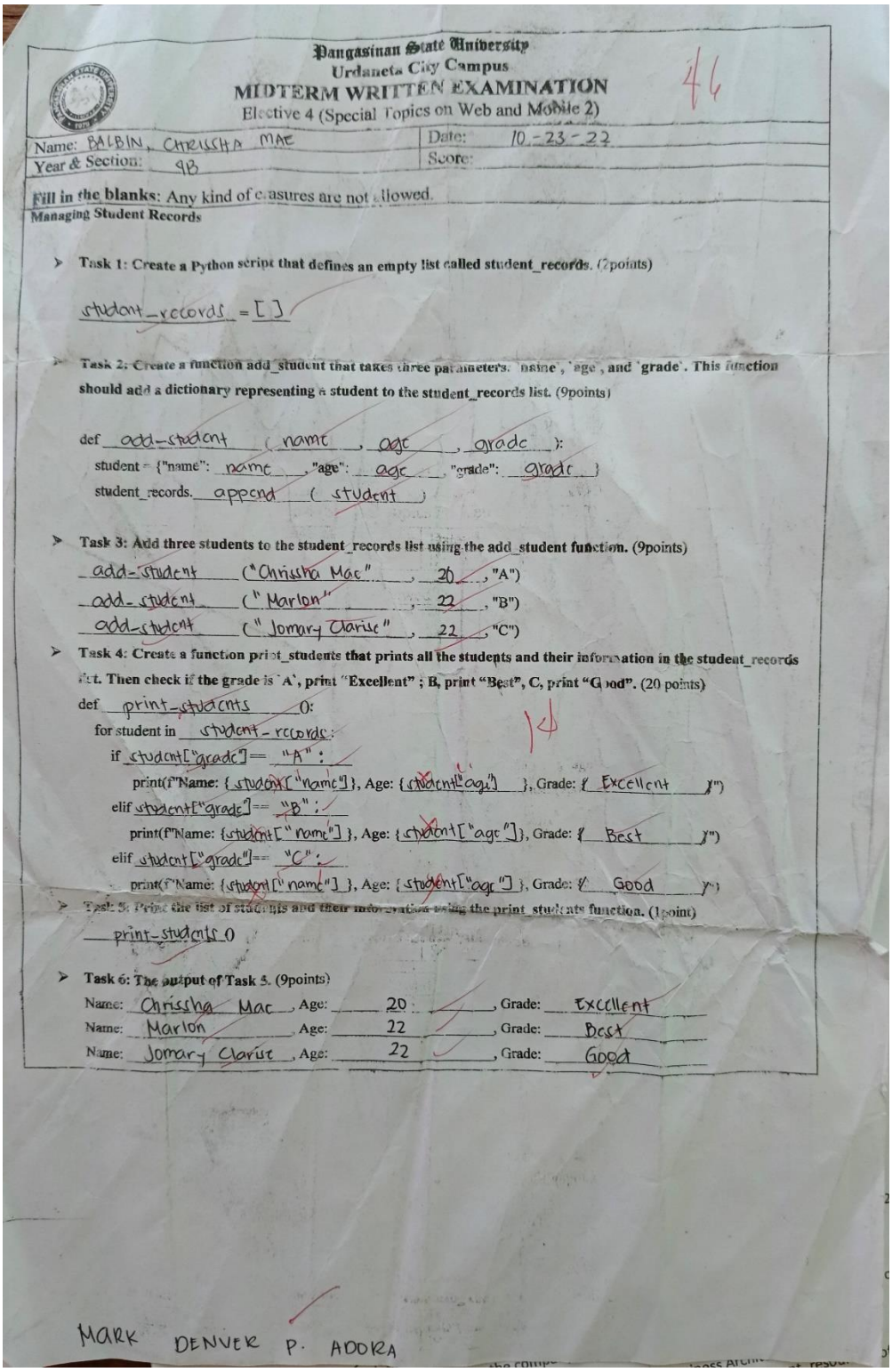
LABORATORY QUIZ 1

3	Balbin, Chrissha Mae Espenueva	24

WRITTEN QUIZ 1



MIDTERM WRITTEN EXAM



Pangasinan State University
Urdaneta City Campus
MIDTERM WRITTEN EXAMINATION
Elective 4 (Special Topics on Web and Mobile 2)

46

Name: BALBIN, CHRISSHA MAE | Date: 10-23-22
Year & Section: 4B | Score:

Fill in the blanks: Any kind of erasures are not allowed.

Managing Student Records

- **Task 1: Create a Python script that defines an empty list called student_records. (2points)**

```
student_records = []
```

- **Task 2: Create a function add_student that takes three parameters: `name`, `age`, and `grade`. This function should add a dictionary representing a student to the student_records list. (9points)**

```
def add_student(name, age, grade):
    student = {"name": name, "age": age, "grade": grade}
    student_records.append(student)
```

- **Task 3: Add three students to the student_records list using the add_student function. (9points)**

```
add_student("Chrissha Mae", 20, "A")
add_student("Marlon", 22, "B")
add_student("Jomary Clarise", 22, "C")
```

- **Task 4: Create a function print_students that prints all the students and their information in the student_records list. Then check if the grade is `A`, print "Excellent"; B, print "Best", C, print "Good". (20 points)**

```
def print_students():
    for student in student_records:
        if student["grade"] == "A":
            print(f"Name: {student['name']}, Age: {student['age']}, Grade: {Excellent}")
        elif student["grade"] == "B":
            print(f"Name: {student['name']}, Age: {student['age']}, Grade: {Best}")
        elif student["grade"] == "C":
            print(f"Name: {student['name']}, Age: {student['age']}, Grade: {Good}")
```

- **Task 5: Print the list of students and their information using the print_students function. (1point)**

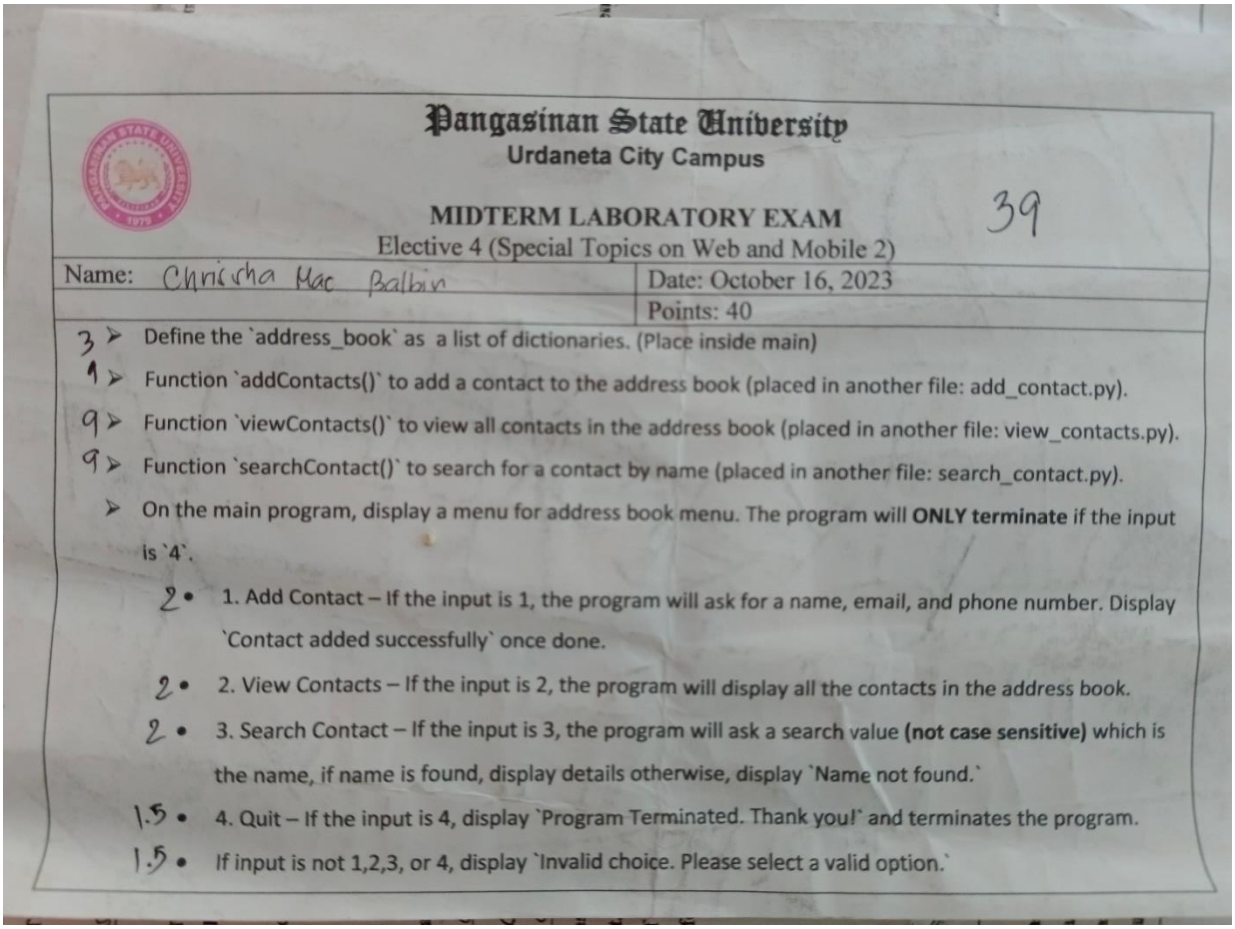
```
print_students()
```

- **Task 6: The output of Task 5. (9points)**

Name: Chrissha Mae, Age: 20, Grade: Excellent
Name: Marlon, Age: 22, Grade: Best
Name: Jomary Clarise, Age: 22, Grade: Good

MARK DENVER P. ADORA

MIDTERM LABORATORY EXAM



Pangasinan State University
Urdaneta City Campus

MIDTERM LABORATORY EXAM 39
Elective 4 (Special Topics on Web and Mobile 2)

Name: Chrischa Mae Balbin	Date: October 16, 2023
	Points: 40

3 ➢ Define the `address_book` as a list of dictionaries. (Place inside main)

1 ➢ Function `addContacts()` to add a contact to the address book (placed in another file: add_contact.py).

9 ➢ Function `viewContacts()` to view all contacts in the address book (placed in another file: view_contacts.py).

9 ➢ Function `searchContact()` to search for a contact by name (placed in another file: search_contact.py).

➢ On the main program, display a menu for address book menu. The program will **ONLY terminate** if the input is `4`.

2 • 1. Add Contact – If the input is 1, the program will ask for a name, email, and phone number. Display `Contact added successfully` once done.

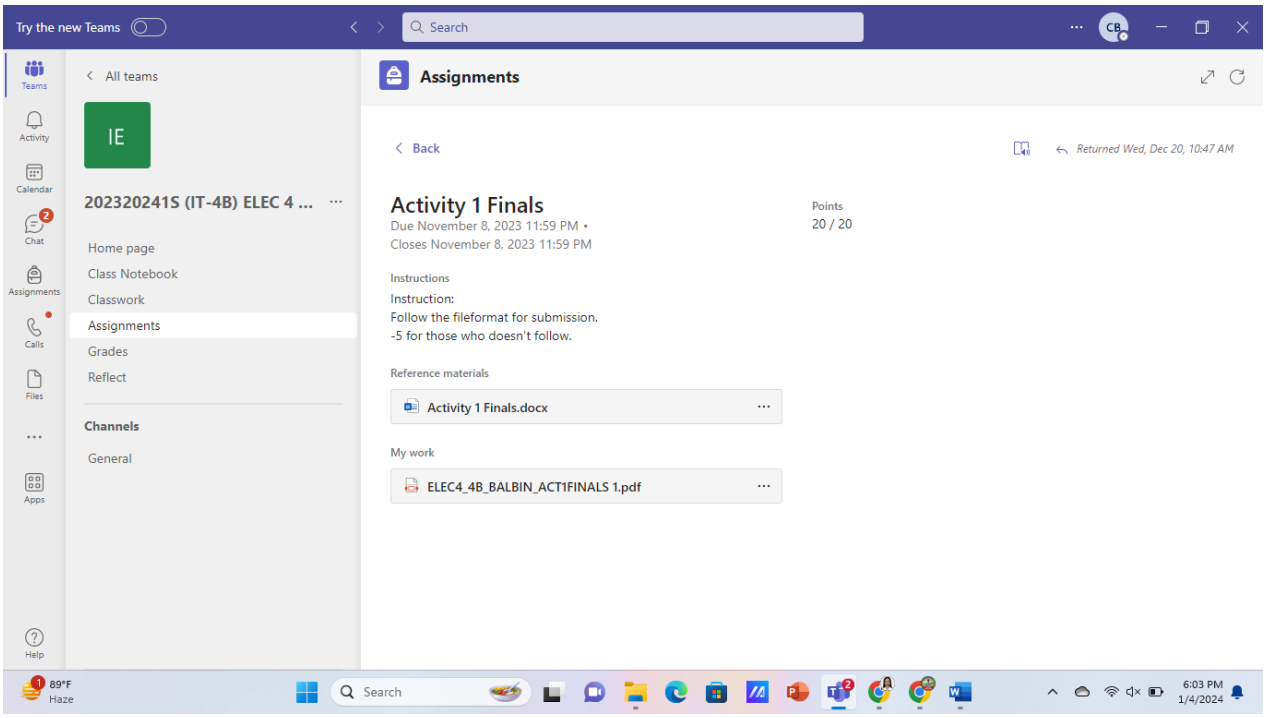
2 • 2. View Contacts – If the input is 2, the program will display all the contacts in the address book.

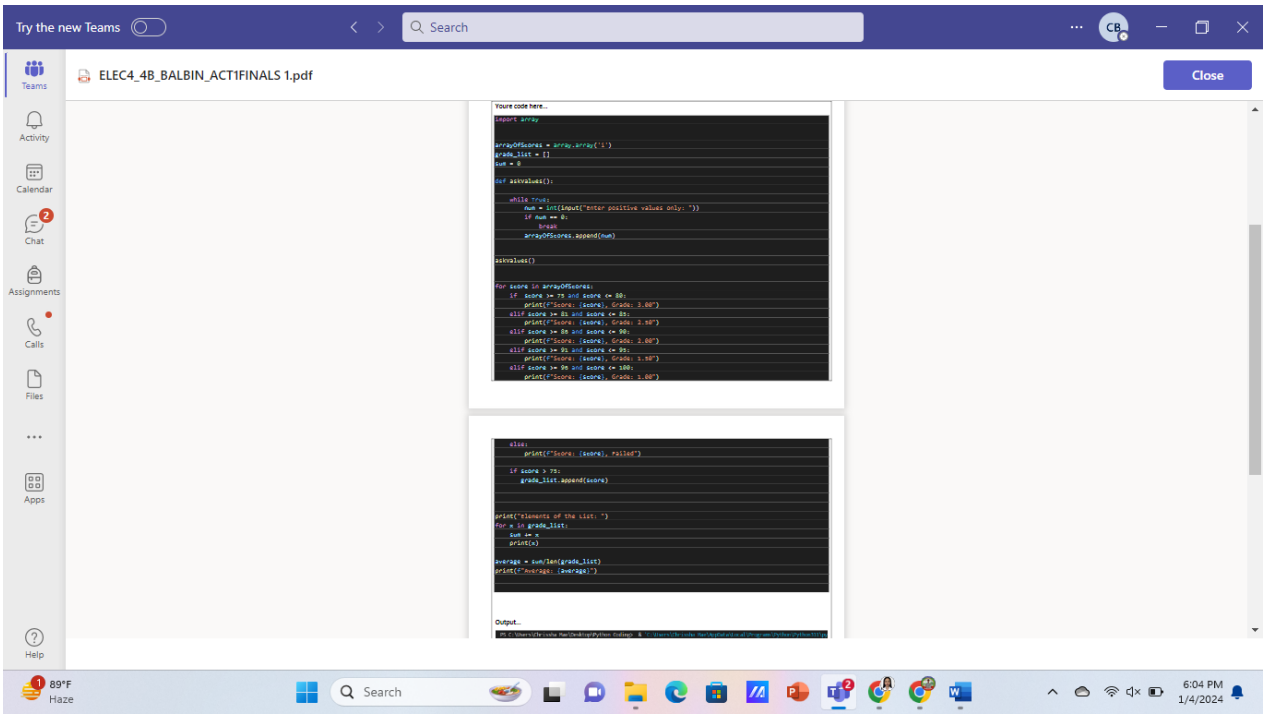
2 • 3. Search Contact – If the input is 3, the program will ask a search value **(not case sensitive)** which is the name, if name is found, display details otherwise, display `Name not found.`

1.5 • 4. Quit – If the input is 4, display `Program Terminated. Thank you!` and terminates the program.

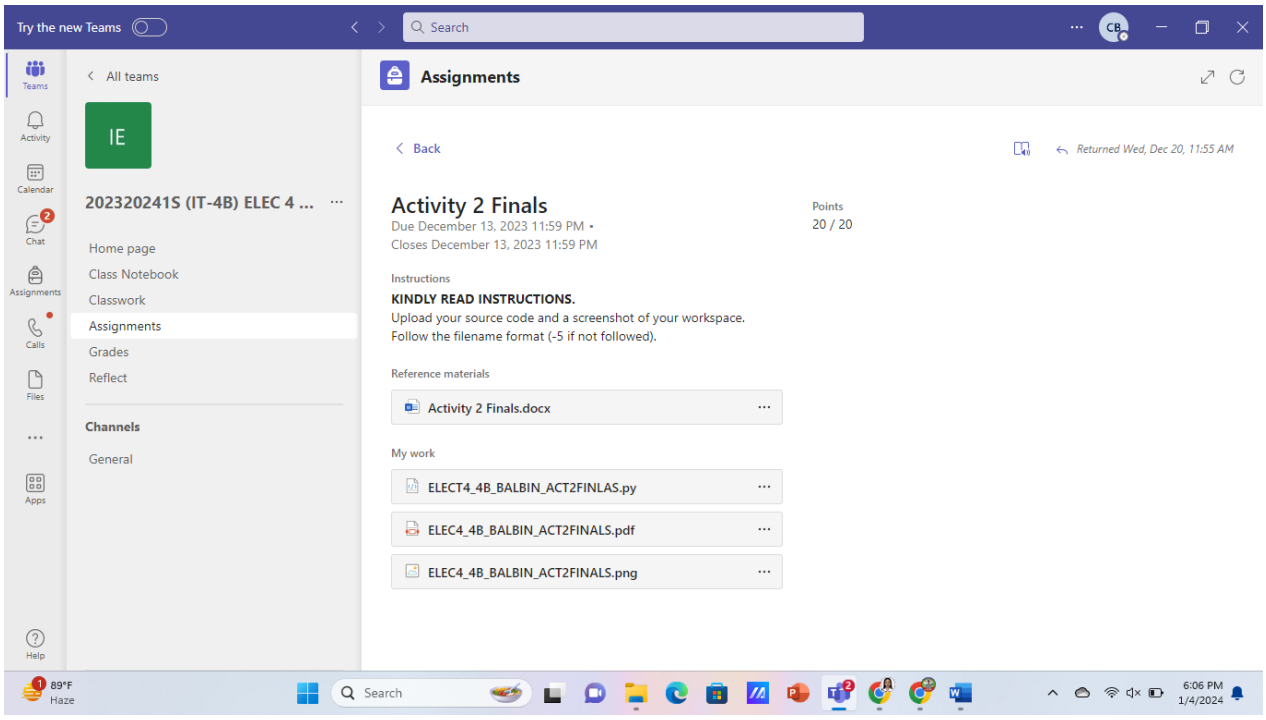
1.5 • If input is not 1,2,3, or 4, display `Invalid choice. Please select a valid option.`

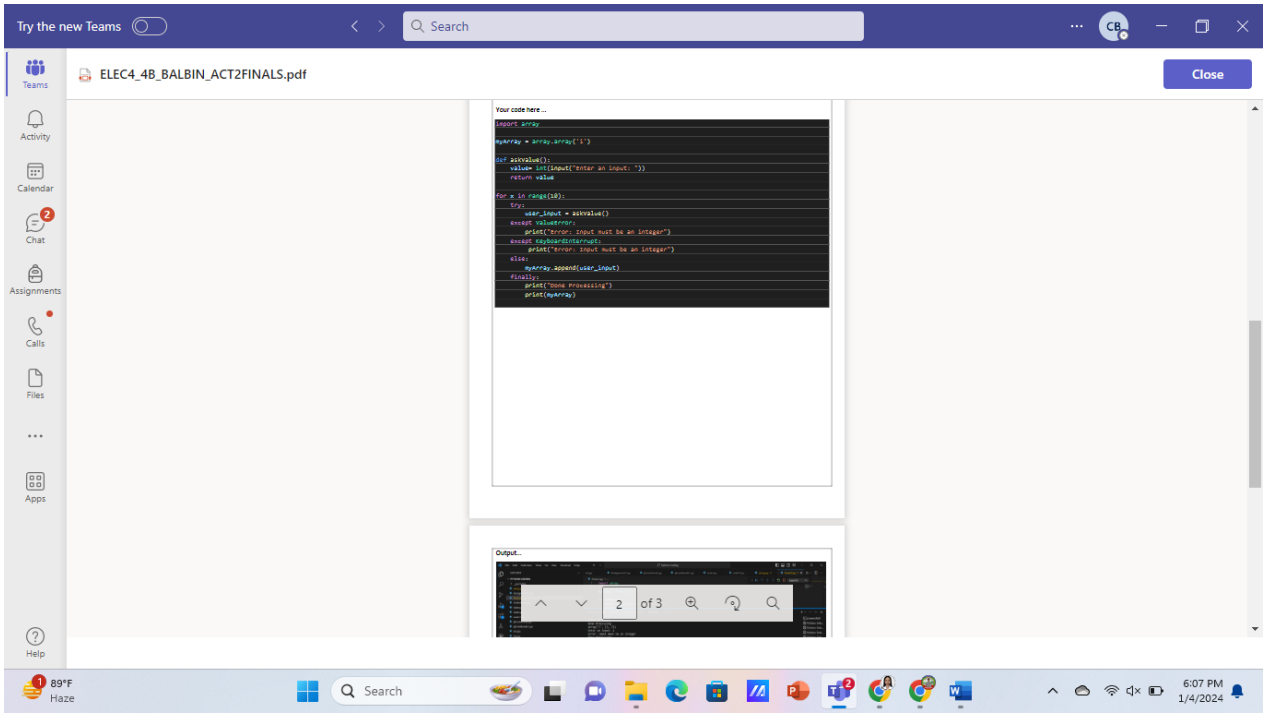
ACTIVITY 1 (FINALS)



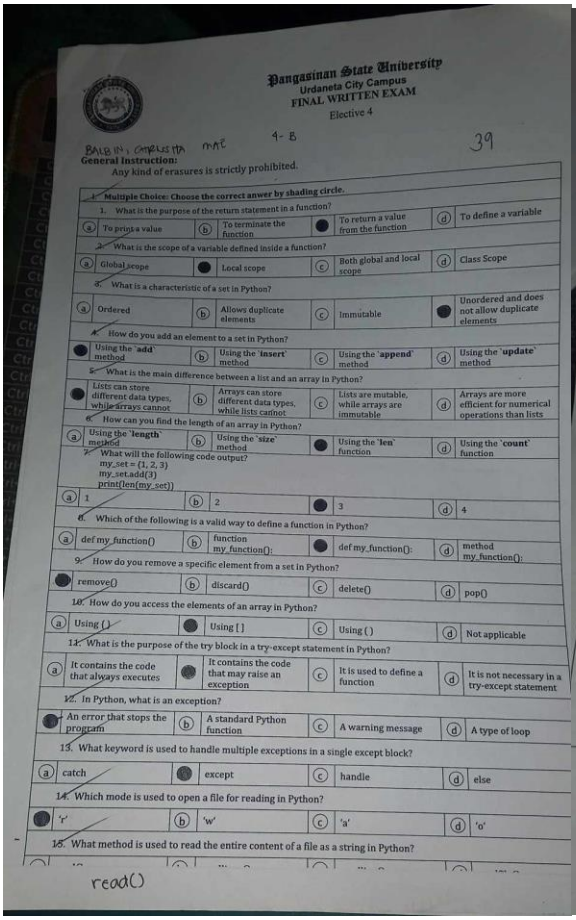


ACTIVITY 2 (FINALS)





FINAL WRITTEN EXAM



Pangasinan State University
Urdaneta City Campus
FINAL WRITTEN EXAM
Elective 4

4-B

39

General Instruction:
Any kind of erasures is strictly prohibited.

I. Multiple Choice: Choose the correct answer by shading circle.

1. What is the purpose of the return statement in a function?
(a) To print a value (b) To terminate the function (●) To return a value from the function (d) To define a variable

2. What is the scope of a variable defined inside a function?
(a) Global scope (●) Local scope (c) Both global and local scope (d) Class Scope

3. What is a characteristic of a set in Python?
(a) Ordered (b) Allows duplicate elements (c) Immutable (●) Unordered and does not allow duplicate elements

4. How do you add an element to a set in Python?
(●) Using the `add` method (b) Using the `insert` method (c) Using the `append` method (d) Using the `update` method

5. What is the main difference between a list and an array in Python?
(●) Lists can store different data types, while arrays cannot (b) Arrays can store different data types, while lists cannot (c) Lists are mutable, while arrays are immutable (d) Arrays are more efficient for numerical operations than lists

6. How can you find the length of an array in Python?
(a) Using the `length` method (b) Using the `size` method (●) Using the `len` function (d) Using the `count` function

7. What will the following code output?
```
my_set = {1, 2, 3}
my_set.add(3)
print(len(my_set))
```
(a) 1 (b) 2 (●) 3 (d) 4

8. Which of the following is a valid way to define a function in Python?
(a) def my_function() (b) function my_function(): (●) def my_function(): (d) method my_function():

9. How do you remove a specific element from a set in Python?
(●) remove() (b) discard() (c) delete() (d) pop()

10. How do you access the elements of an array in Python?
(a) Using () (●) Using [ ] (c) Using { } (d) Not applicable

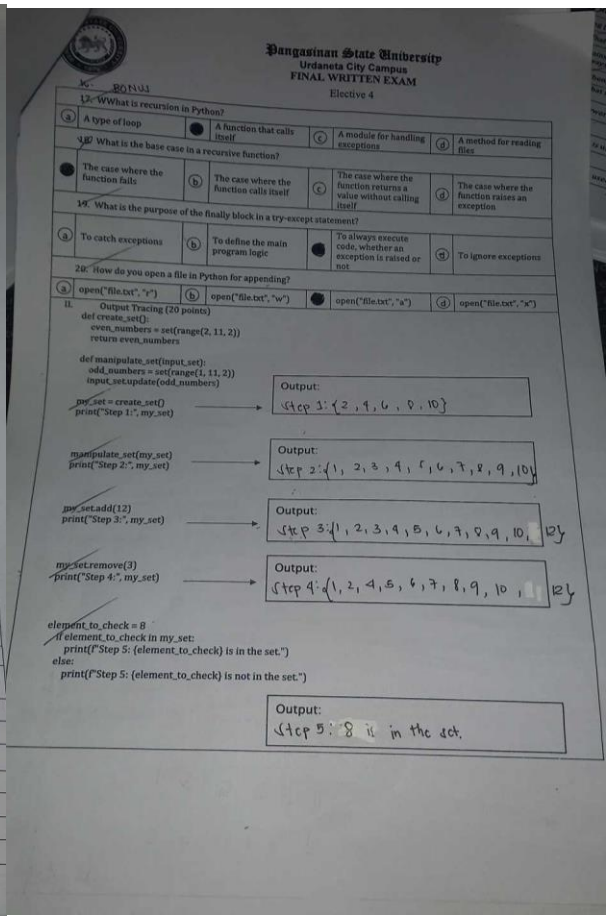
11. What is the purpose of the try block in a try-except statement in Python?
(a) It contains the code that always executes (●) It contains the code that may raise an exception (c) It is used to define a function (d) It is not necessary in a try-except statement

12. In Python, what is an exception?
(●) An error that stops the program (b) A standard Python function (c) A warning message (d) A type of loop

13. What keyword is used to handle multiple exceptions in a single except block?
(a) catch (●) except (c) handle (d) else

14. Which mode is used to open a file for reading in Python?
(●) 'r' (b) 'w' (c) 'a' (d) 'o'

15. What method is used to read the entire content of a file as a string in Python?
read()



Pangasinan State University
Urdaneta City Campus
FINAL WRITTEN EXAM
Elective 4

BONUS

12. What is recursion in Python?
(a) A type of loop (●) A function that calls itself (c) A module for handling exceptions (d) A method for reading files

13. What is the base case in a recursive function?
(●) The case where the function fails (b) The case where the function calls itself (c) The case where the function returns a value without calling itself (d) The case where the function raises an exception

14. What is the purpose of the finally block in a try-except statement?
(a) To catch exceptions (b) To define the main program logic (●) To always execute code, whether an exception is raised or not (d) To ignore exceptions

20. How do you open a file in Python for appending?
(a) open("file.txt", "r") (b) open("file.txt", "w") (●) open("file.txt", "a") (d) open("file.txt", "x")

II. Output Tracing (20 points)

```
def create_set():
    even_numbers = set(range(2, 11, 2))
    return even_numbers

def manipulate_set(input_set):
    odd_numbers = set(range(1, 11, 2))
    input_set.update(odd_numbers)

my_set = create_set()
print("Step 1:", my_set)
```
Output: Step 1: {2, 4, 6, 8, 10}

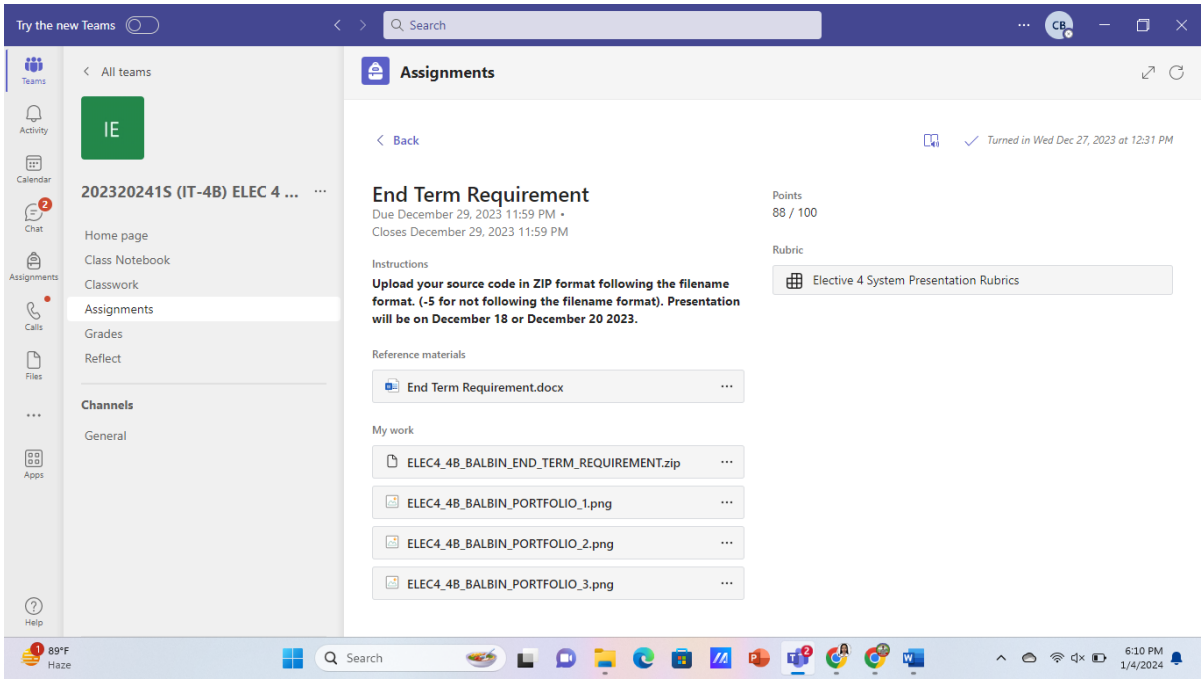
```
manipulate_set(my_set)
print("Step 2:", my_set)
```
Output: Step 2: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

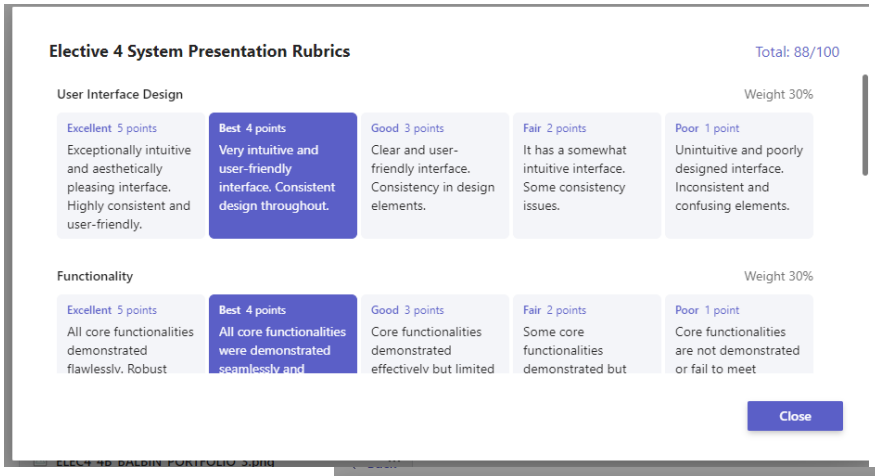
```
my_set.add(12)
print("Step 3:", my_set)
```
Output: Step 3: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12}

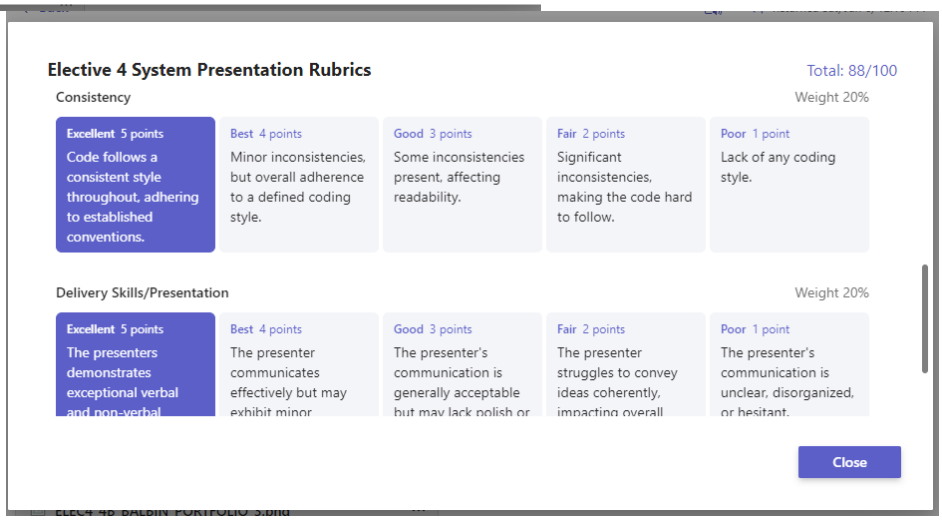
```
my_set.remove(3)
print("Step 4:", my_set)
```
Output: Step 4: {1, 2, 4, 5, 6, 7, 8, 9, 10, 12}

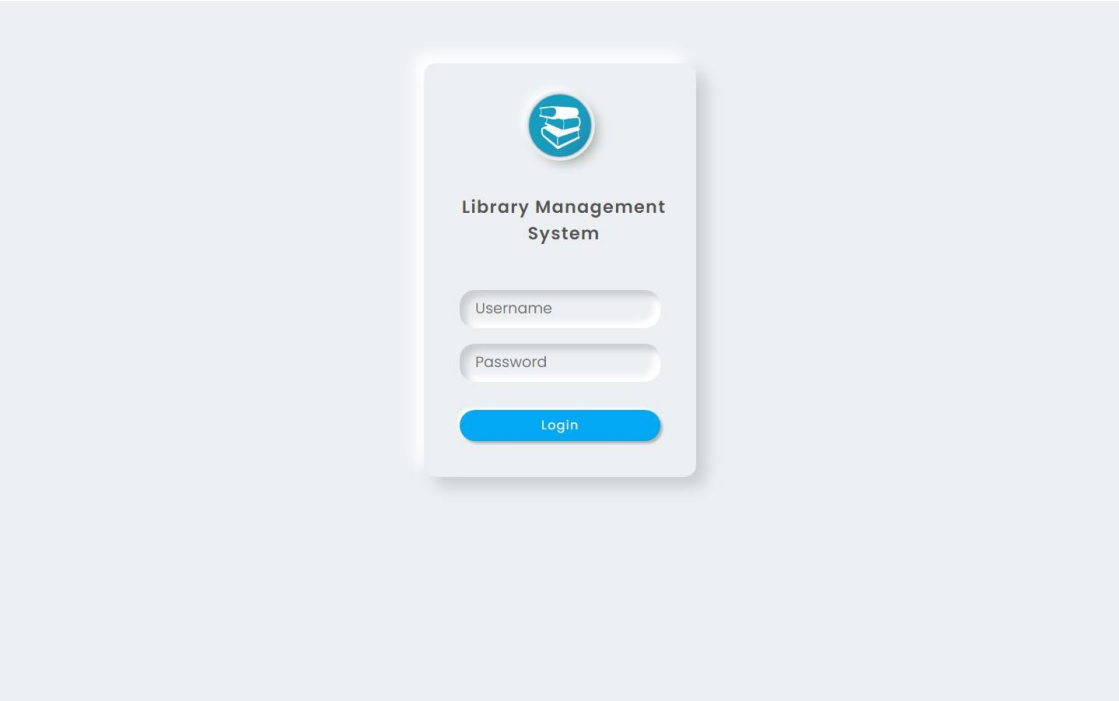
```
element_to_check = 8
if element_to_check in my_set:
    print(f"Step 5: {element_to_check} is in the set.")
else:
    print(f"Step 5: {element_to_check} is not in the set.")
```
Output: Step 5: 8 is in the set.

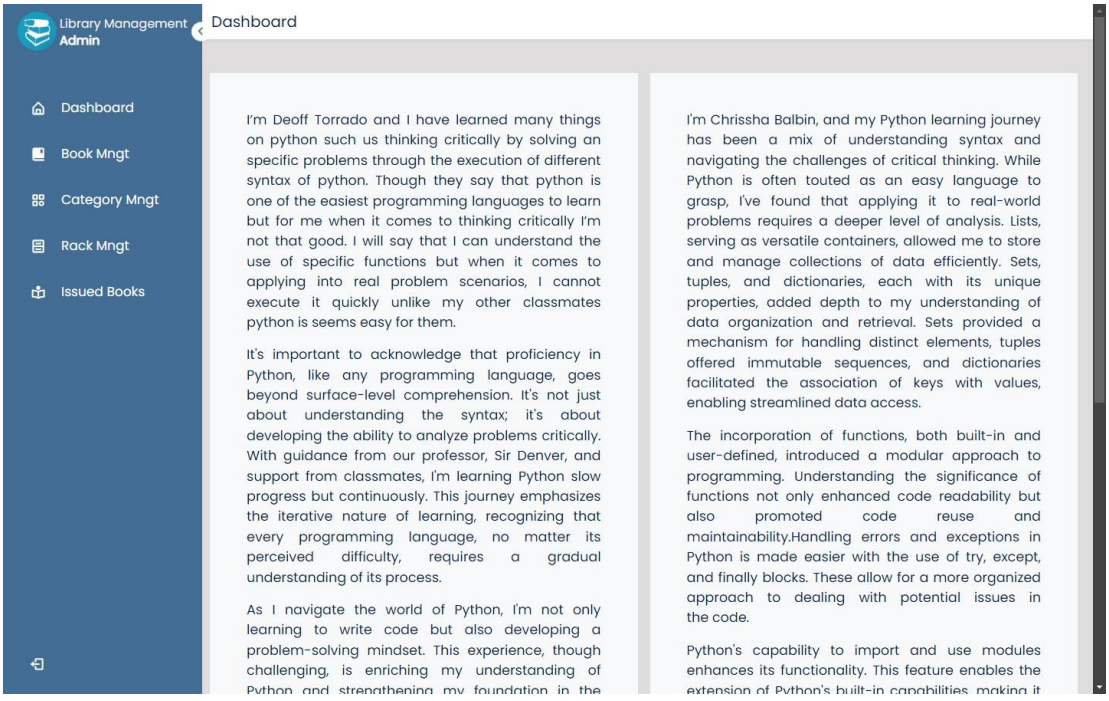
END-TERM REQUIREMENT








Library Management System
Username
Password
Login


Library Management
Admin
Dashboard
Dashboard
Book Mngt
Category Mngt
Rack Mngt
Issued Books
I'm Deoff Torrado and I have learned many things on python such us thinking critically by solving an specific problems through the execution of different syntax of python. Though they say that python is one of the easiest programming languages to learn but for me when it comes to thinking critically I'm not that good. I will say that I can understand the use of specific functions but when it comes to applying into real problem scenarios, I cannot execute it quickly unlike my other classmates python is seems easy for them.
It's important to acknowledge that proficiency in Python, like any programming language, goes beyond surface-level comprehension. It's not just about understanding the syntax; it's about developing the ability to analyze problems critically. With guidance from our professor, Sir Denver, and support from classmates, I'm learning Python slow progress but continuously. This journey emphasizes the iterative nature of learning, recognizing that every programming language, no matter its perceived difficulty, requires a gradual understanding of its process.
As I navigate the world of Python, I'm not only learning to write code but also developing a problem-solving mindset. This experience, though challenging, is enriching my understanding of
I'm Chrissha Balbin, and my Python learning journey has been a mix of understanding syntax and navigating the challenges of critical thinking. While Python is often touted as an easy language to grasp, I've found that applying it to real-world problems requires a deeper level of analysis. Lists, serving as versatile containers, allowed me to store and manage collections of data efficiently. Sets, tuples, and dictionaries, each with its unique properties, added depth to my understanding of data organization and retrieval. Sets provided a mechanism for handling distinct elements, tuples offered immutable sequences, and dictionaries facilitated the association of keys with values, enabling streamlined data access.
The incorporation of functions, both built-in and user-defined, introduced a modular approach to programming. Understanding the significance of functions not only enhanced code readability but also promoted code reuse and maintainability.Handling errors and exceptions in Python is made easier with the use of try, except, and finally blocks. These allow for a more organized approach to dealing with potential issues in the code.
Python's capability to import and use modules enhances its functionality. This feature enables the