



اَوْنُوْ سِيْطِيْ تِيْكَوْلُوْجِيْ مَارَا
UNIVERSITI
TEKNOLOGI
MARA

Cawangan Kedah
Kampus Sungai Petani

**SCHOOL OF INFORMATION SCIENCE,
COLLEGE OF COMPUTING, INFORMATICS AND MATHEMATICS
UNIVERSITI TEKNOLOGI MARA**

CDIM 144 - DIPLOMA IN INFORMATICS LIBRARY

IML 208 PROGRAMMING FOR LIBRARIES

**INDIVIDUAL ASSIGNMENT:
MANGA BOOKING SYSTEM**

**PREPARED BY
MUHAMMAD ZAHIRUDDIN BIN IZAM AZALI (2023804798)**

KCDIM1443 E

**PREPARED FOR
SIR MOHD FIRDAUS BIN MOHD HELMI**

**SUBMISSION DATE
WEEK 10**

Project Name: Manga Booking System

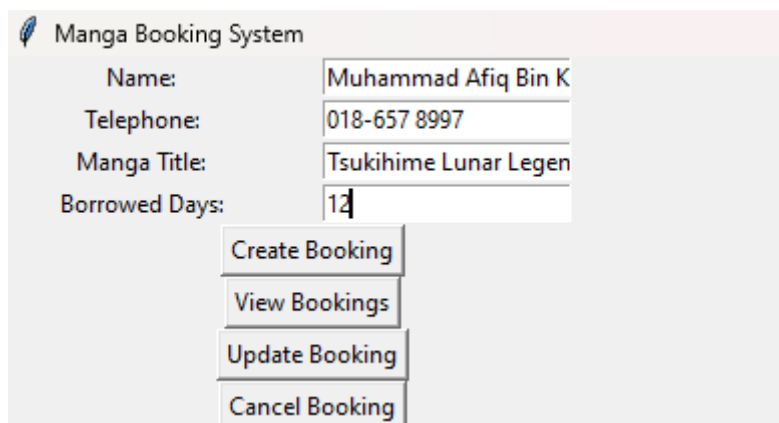
File name: Mangabookingsystem.py

Prompt Data:

- i) Name
- ii) Telephone number
- iii) Manga title
- iv) Number of borrowed days

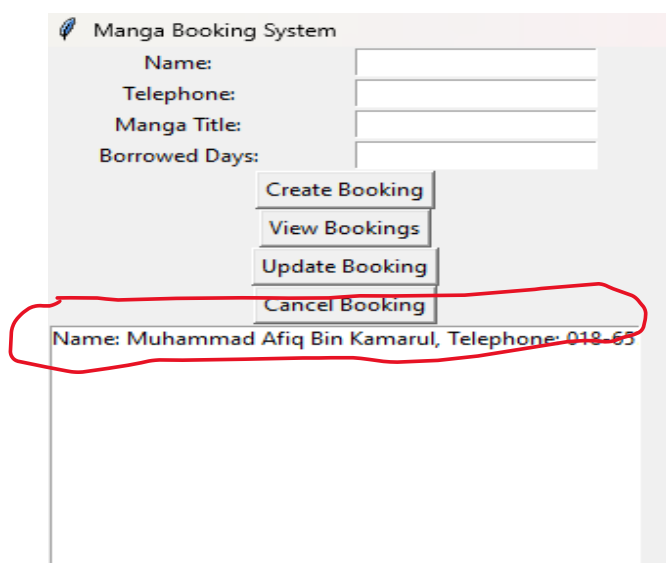
Function:

- i. Create the data



The screenshot shows a window titled "Manga Booking System". Inside, there are four input fields with labels: "Name:", "Telephone:", "Manga Title:", and "Borrowed Days:". The fields contain the following text: "Muhammad Afiq Bin K", "018-657 8997", "Tsukihime Lunar Legen", and "14". Below the input fields are four buttons stacked vertically: "Create Booking", "View Bookings", "Update Booking", and "Cancel Booking".

- ii. Read the data



The screenshot shows the same "Manga Booking System" window. The input fields are now empty. The buttons "Create Booking", "View Bookings", "Update Booking", and "Cancel Booking" are still present. Below the buttons, a red oval highlights a text label that reads: "Name: Muhammad Afiq Bin Kamarul, Telephone: 018-657 8997".

iii. Update the data

Before update the data of customer borrowed days

Manga Booking System

Name:	mad Afiq Bin Kamarul
Telephone:	018-657 8997
Manga Title:	ukihime Lunar Legend
Borrowed Days:	14

Create Booking

View Bookings

Update Booking

Cancel Booking

997, Manga: Tsukihime Lunar Legend, Borrowed Days: 12

Booking update successfully

Manga Booking System

Name:	mad Afiq Bin Kamarul
Telephone:	018-657 8997
Manga Title:	ukihime Lunar Legend
Borrowed Days:	14

Create Booking

View Bookings

Update Booking

Cancel Booking

997, Manga: Tsukihime Lunar Legend, Borrowed Days: 12

Success

Booking updated successfully!

OK

After update the data of customer borrowed days

The screenshot shows the 'Manga Booking System' interface. On the left, there are input fields for 'Name:', 'Telephone:', 'Manga Title:', and 'Borrowed Days:'. Below these fields are four buttons: 'Create Booking', 'View Bookings', 'Update Booking', and 'Cancel Booking'. The 'Update Booking' button is highlighted with a red circle. Below the buttons, there is a list of bookings. The first item in the list is '997, Manga: Tsukihime Lunar Legend, Borrowed Days: 14', which is also highlighted with a red circle. Below the list is a large empty rectangular area.

iv. Delete the data

Before delete the data

The screenshot shows the 'Manga Booking System' interface. On the left, there are input fields for 'Name:', 'Telephone:', 'Manga Title:', and 'Borrowed Days:'. Below these fields are four buttons: 'Create Booking', 'View Bookings', 'Update Booking', and 'Cancel Booking'. The 'Delete Booking' button is highlighted with a red circle. Below the buttons, there is a list of bookings. The first item in the list is '997, Manga: Tsukihime Lunar Legend, Borrowed Days: 14', which is also highlighted with a red circle. Below the list is a large empty rectangular area.

Order cancels successfully

Manga Booking System

Name:	mad Afiq Bin Kamarul
Telephone:	018-657 8997
Manga Title:	tsukihime Lunar Legend
Borrowed Days:	12

Create Booking

View Bookings

Update Booking

Cancel Booking

997, Manga: Tsukihime Lunar Legend, Borrowed Days: 12

Success

i

Order cancelled successfully!

OK

No booking is available

Manga Booking System

Name:	
Telephone:	
Manga Title:	
Borrowed Days:	

Create Booking

View Bookings

Update Booking

Cancel Booking

Conditional Statement: Yes

```
53 def main():
54     bookings = []
55     while True:
56         show_menu()
57         choice = input("Choose an option (1-5): ")
58         if choice == "1":
59             create_booking(bookings)
60         elif choice == "2":
61             view_bookings(bookings)
62         elif choice == "3":
63             update_booking(bookings)
64         elif choice == "4":
65             delete_booking(bookings)
66         elif choice == "5":
67             print("Exiting Manga Booking System. Goodbye!")
68             break
69         else:
70             print("Invalid choice. Please try again.")
71
72 if __name__ == "__main__":
73     main()
```

GUI: Yes

Manga Booking System

Name:

Telephone:

Manga Title:

Borrowed Days:

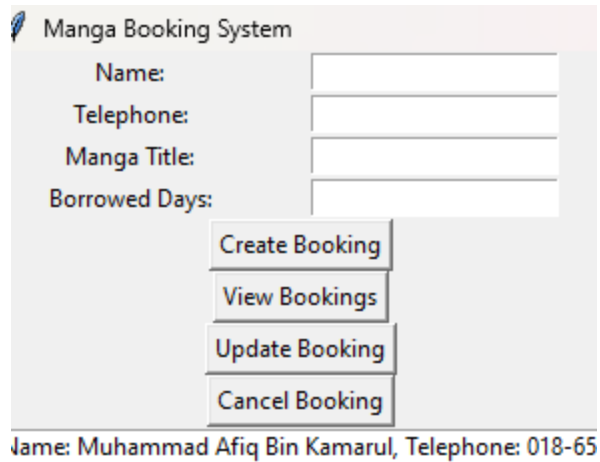
Create Booking

View Bookings

Update Booking

Cancel Booking

Result: Screenshot



The screenshot displays the 'Manga Booking System' window. It features a title bar with a pencil icon and the text 'Manga Booking System'. Below the title bar, there are four input fields with labels: 'Name:', 'Telephone:', 'Manga Title:', and 'Borrowed Days:'. To the right of these fields are four empty text boxes. Below the input fields, there are four buttons stacked vertically: 'Create Booking', 'View Bookings', 'Update Booking', and 'Cancel Booking'. At the bottom of the window, there is a status bar that reads 'Name: Muhammad Afiq Bin Kamarul, Telephone: 018-65'.

Strengths:

Comprehensive Functionality:

- The application supports essential CRUD (Create, Read, Update, Delete) operations, making it versatile and functional for managing bookings.
- Each action (create, view, update, delete) is well-defined and clearly implemented.

User Feedback:

- Utilizes messagebox that providing immediate feedback to users, such as success or error messages and it also enhancing user experience.
- This ensures users are informed about the outcome of their actions and reducing their confusion.

Clear Layout:

- The GUI layout is clean and straightforward, with labels, entry fields, buttons, and a listbox organized in a logical manner.

Add input validation for users:

Currently, there's minimal input validation. Users might enter invalid data. For example, letters in "Borrowed Days" or an incorrect telephone format.

User-friendly error messages:

The error messages are functional but could be more descriptive and user-focused.

Improved UI design:

the GUI layout works but could be more visually appealing.