LONDON METROPOLITAN UNIVERSITY

ITAHARI INTERNATIONAL COLLEGE

Module Code & Module Title:

CS4051NI

Fundamental of computing

Assessment Weightage & Type: **60%**

Individual Coursework

Year and Semester: **2023**

Autumn

[redacted]

I confirm that I understand my coursework needs to be submitted online via MySecondTeacher under the relevant module page before the deadline for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded

Tables of contents:

Tables of Images:

Table of tables:

1 Introduction

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. It was created by Guido van Rossum in 19991 and further developed by the python Software foundation. (Geeks for Geeks, 2023). Python's syntax is designed to be clear and concise, utilizing indentation to denote block structure instead of traditional braces or keywords. This feature, known as the "off-side-rule", contributes to Python's clean and easy to understand code.

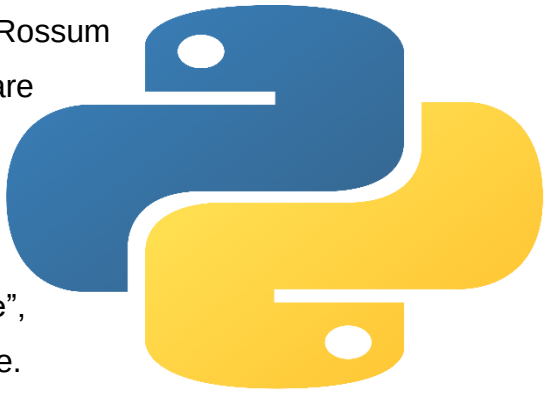


Figure 1: Python

1.1 Introduction to project

The project is about creating a land rental system that helps to keeps the basic records of land rent and returns by the costumer. This project was created as a coursework for the module "Fundamental of computing". The project was created by using various tools like IDLE, draw.io and MS-Words. In the project, there is scenario of Techno Property Nepal, a company managing land on a contract basis in Nepal. The program is tasked with reading a text file containing land availability information, facilitating transactions such as renting and returning land, and generating invoices for each transaction. We are required to develop the program based on detailed specifications and present it in a structured report.

1.2 Goals and Objectives

- To learn about computing and programming languages.
- To learn about the various built-in keywords in python.
- To explore python's essential data structures.
- To know about the conditional statements for decision making in code.
- To develop strong skills in algorithms design, flowcharting, and pseudocode creation.
- To understand error and exception handling techniques.
- To develop a land rental system.
- To test the program and see if it is functional as required.

1.3 Discussion and Analysis

In this coursework, the task is to develop The Land Rental system using python based on the given specifications. The system must be capable of managing land availability, processing rental, return, transactions and generating invoices for each transaction. Some of the key aspects for this coursework are discussed below.

1.4 Tools used.

- **MS-Word**

Microsoft word is a word processor software developed by Microsoft in 1993. (Geeks for Geeks, 2021). It helps us to create, edit, format and share documents. With its friendly interface, we can easily create documents. The software provides extensive text editing features like copy, cut, paste, and undo, along with formatting options like font styles, sizes, color and alignment. So, I used MS word for making this documentation.

Figure 2: MS-Word

- **Python Idle**

IDLE stands for Integrated Development and Learning Environment which is an integrated development environment for python that comes along with python installer for windows. (HIgssoftware, n.d.). it provides a user-friendly development interface for writing, executing, and debugging python code. Some of the key features and functionalities of python idle are it comes with built-in code editor which offers features

like syntax highlighting, indentation support, multiple color. Some of the other features are interactive shell, debugging tools, integrated documentation, customizable and cross platform support.

Figure 3: Python IDLE

- **Draw.io**

Draw.io is a web-based drawing tool that allows users to create wide variety of diagrams, flowcharts, mind maps, class diagrams and more. It is very user friendly with a comprehensive set of features for creating, editing and sharing diagram so I use draw.io for making flowcharts.

Figure 4: Draw.io

2 Algorithm

The words Algorithm means “A set of finite rules or instructions to be followed in calculation or other problem-solving operations” or “A procedure for solving a mathematical problem in a finite number of steps that frequently involves recursive operations”. (Geeks for Geeks, 2023). Algorithms play a significant role in multiple fields and have multiple application. Some of the common areas where algorithms are used are computer science, mathematics, operation research, artificial intelligence, and data sciences.

The algorithm used for developing program is described below:

STEP1: START

STEP2: Import all the necessary modules.

STEP3: Define all the necessary function for:

- Printing header, menu, and footer.
- Define validating function for kitta number input.
- Define function for validating duration input.
- Define function for checking if the land is rented or not.
- Define main method to control the program flow.

STEP4: Print a welcome message and header for land rental system.

STEP5: Print menu for the user choice:

- Display land information.
- Rent land.
- Return land.
- Exit

STEP6: if user choice equals 1.

STEP7: Call DisplayInfo function

- Start try block.
- Initialize variables like gap, heading.
- Print heading.
- Open the land file illiterates over each line.
- Split the line data by comma.
- Print the record.
- If file not found print the error message

STEP8: if user choice equals 2.

STEP7: Call rent function.

- Get the current date and time.
- Read the contents in the land file.

- Iterate through each line in the file and check if the land with specified kitta number is available for rent.
- If the land is available. Update its status to 'Rented' in the file.
- Generates invoice.
- Print the invoice.
- If file not found display an error message as "file not found"

STEP9: If user choice equals 3.
STEP10: Call return function.

- Get the current date and time.
- Read the contents in the land file.
- Iterate through each line in the file and the status of land:
- if the land is rented. Update its status to 'Available' in the file.
- calculate the fine if return duration exceeds the rent duration.
- Generates invoice in text file.
- Print the invoice in console.
- If file not found display an error message as "file not found"

STEP10: if user choice equals 4:

- Exits from the main loop.

STEP11: if user choice is any thigs except option

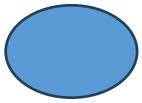
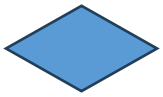
- Display an error message.

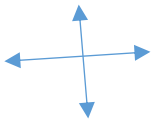
STEP11: Stop

3 Flowchart:

A flowchart is the graphical representation of program flow. We use it as a program planning tool to develop the logic for program. Each step in the process is represented by a different symbol and contains a short description of the process step. The flow chart symbol is linked together with arrows showing the process flow direction. (Hebb, n.d.). **The symbols used in flowcharts are given below with their purpose and Descriptions.**

Table 1: Symbols used in flowcharts.

Symbol Name:	Symbol:	Description:
Oval		It represents the beginning and end point of a process or algorithm and usually represented by oval shape.
Parallelogram		It represents inputs from the user or output to the user. Input or output operations involve reading data from user or displaying results.
Rectangle		It represents processing and is used for arithmetic operation and data manipulation.
Diamonds		It represents decision making and uses for the operation in which there are multiple alternatives like true, false, if etc.

Arrows		It is used to indicate the flow logic by connecting symbols.

The flow chart used while developing program:
For main.py

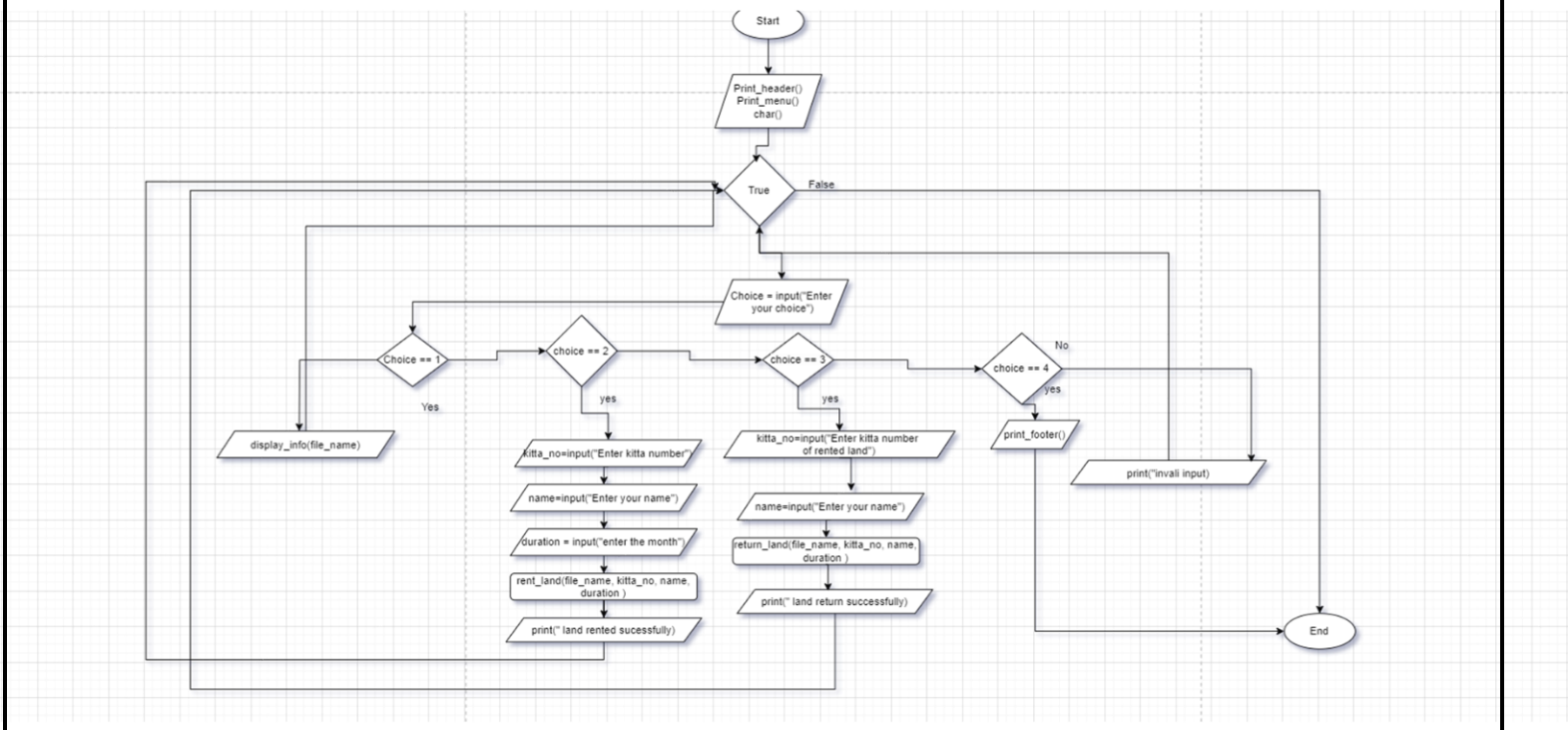


Figure 5: Flowcharts of main.py.

For rent.py

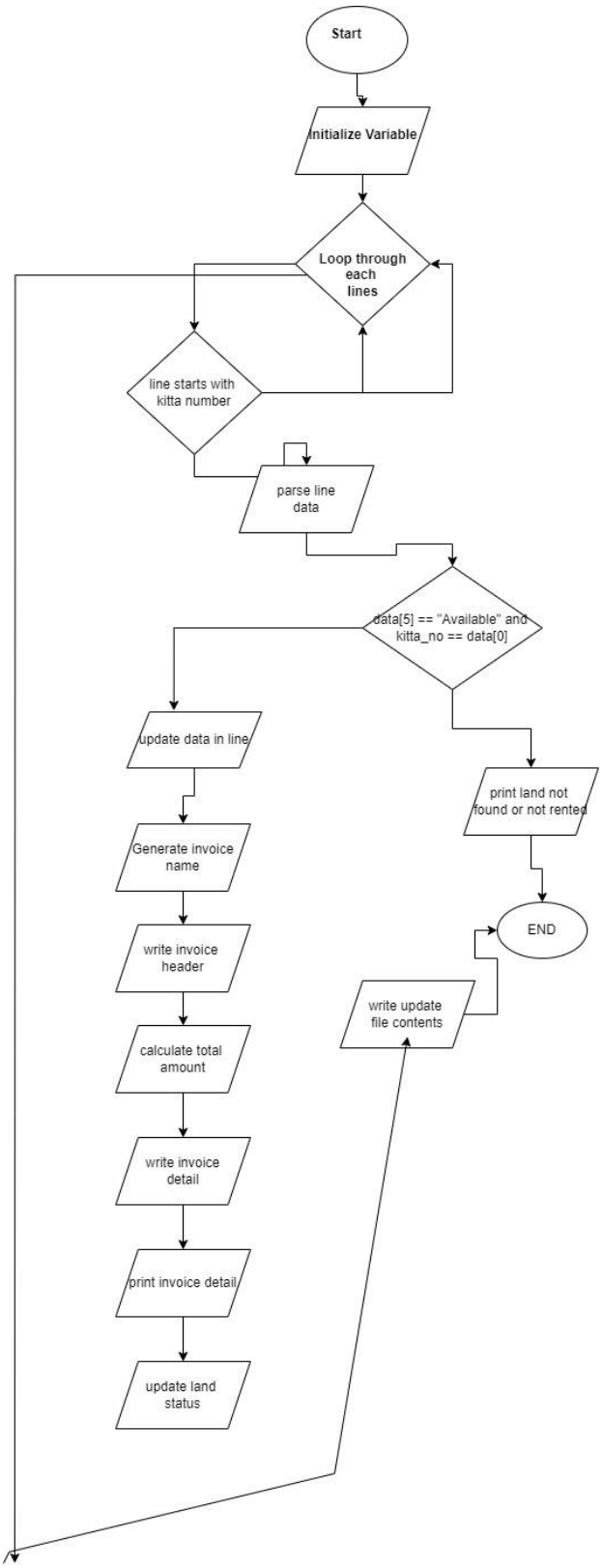


Figure 6: Flow chart of rent.py.

For return.py

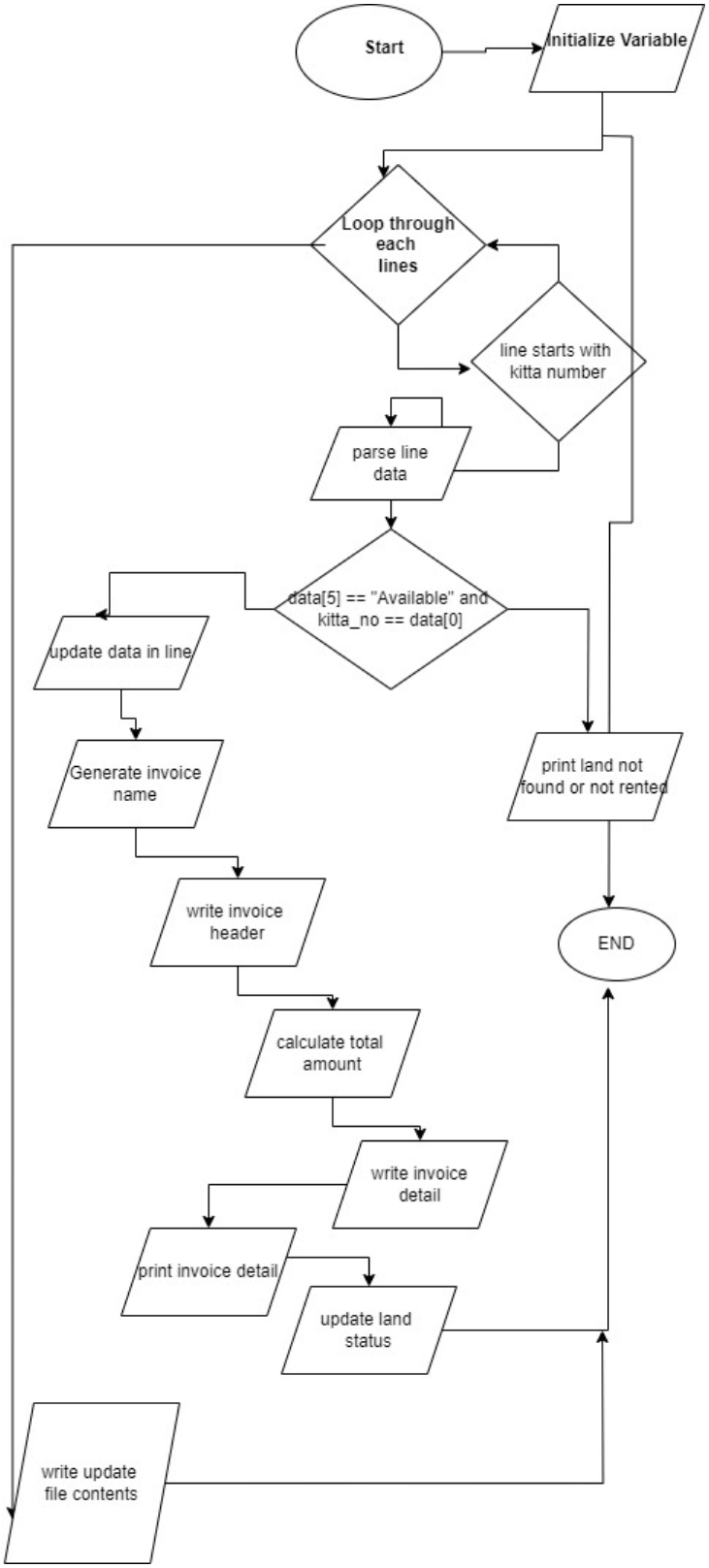


Figure 7: return.py

For DisplayInfo.py

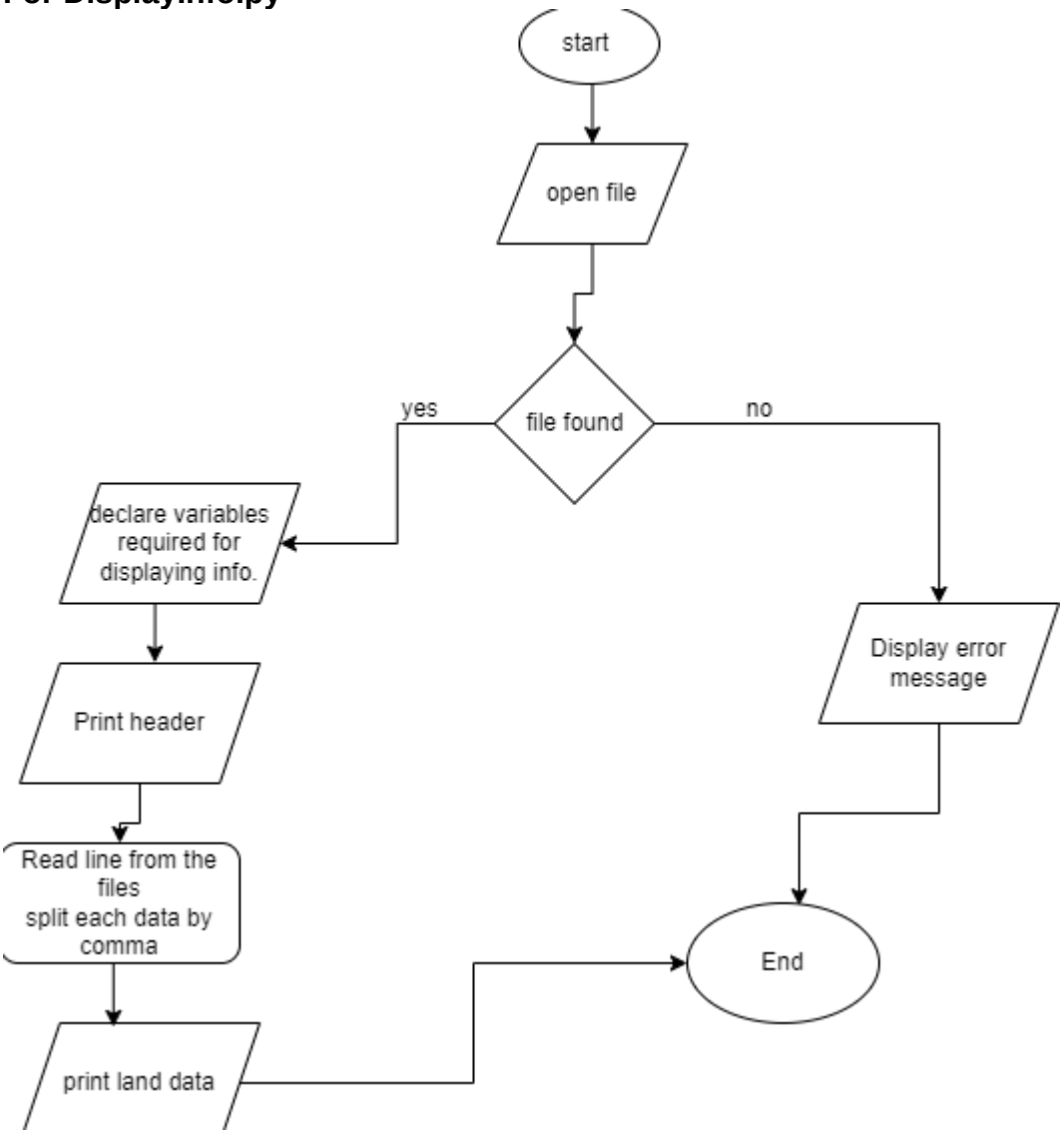


Figure 8: DisplayInfo.py flow charts

4 Pseudocode

A pseudocode is an unofficial way of coding description which does not need any programming language syntax. It is used for developing an outline of a program to understand the methods used in it. It is not an actual programming language so it cannot be compiled**.** (The economic times, 2021)

The pseudocode for the program is given below:

- **For main.py**

IMPORT time.
FROM display_land_info IMPORT displayInfo
FROM rent_land IMPORT rent_land
FROM return_land IMPORT return_land
DEFINE print_header function:
DEFINE print_menu: function:
DEFINE print_footer function:
DEFINE validate_kitta_number with parameters kitta_no, and lans_data:
    FOR each line in land_data
        SPLIT line by comma
        IF kitta_no is in data
            RETURN true
    RETURN false
DEFINE validate_duration with parameter duration
  WHILE true
    TRY
      CONVERT duration to integer
      IF duration > 0
        RETURN duration
      ELSE
        DISPLAY "Duration must be a positive integer."
    EXCEPT ValueError
      DISPLAY "Duration must be a positive integer."
    SET duration to input "Enter the duration of rent (in months): "

DEFINE is_land_rented with parameters kitta_no, file_name
OPEN file_name as file
FOR each line in file
SPLIT line by ','
IF kitta_no equals data[0]
IF last element of data equals "Rented"
RETURN true
ELSE
RETURN false
RETURN false
DEFINE main function:
SET file_name to land_info.txt
CALL print_header function
WHILE true
CALL print_menu function
DECLARE variables ‘choice’ for user input
IF choice equals “1”
CALL DisplayInfo with parameter file_nam
ELSE IF choice equals “2”
WHILE true
DECLARE kitta_no to input kitaa number of the land to rent
IF CALL validate_kitta_number with parameters kitta_no, open(file_name)
BREAK
DECLARE customer_name to input “Enter your name:”
DECLARE duration and call validate_ duration for input “Enter thr duration of rent:”
DECLARE ph_number to input phone number
CALL rent_land with parameters file_name, kitta_no, customer_name, duration, ph_number
ELSE IF choice equals “3”
WHILE true
DECLARE kitta_no to input "Enter the kitta number of the rented land to return: "

IF CALL validate_kitta_number with parameters kitta_no and open(file_name)

BREAK

DECLARE customer_name to input "Enter your name: "

DECLARE rent_dur to CALL validate_duration with parameter input "Enter the duration of rent (in months): "

DECLARE return_dur to CALL validate_duration with parameter input "Enter the duration of return (in months): "

DECLARE ph_number to input "Enter your phone number: "

CALL return_land with parameters file_name, kitta_no, customer_name, rent_dur, return_dur, ph_number

ELSE IF choice equals "4"

CALL print_footer

BREAK

ELSE

DISPLAY "Invalid choice. Please enter a valid option."

CALL mainFunction IF __name__ equals "__main__"

- **For rent.py**

IMPORT datetime.

DEFINE rent_land function with parameters file_name, kitta_no, customer_name, duration, ph_number

now = **GET** current date and time

formatted_datetime = FORMAT now to string using format " %H:%M:%S"

date = **GET** date from now

TRY

OPEN file_name for reading as file

READ lines from file

FOR each line and index in lines
IF line starts with kitta_no
data = SPLIT line by ', '
IF data[5] equals "Available" **AND** kitta_no equals data[0]
UPDATE lines[i] to "{data[0]}, {data[1]}, {data[2]}, {data[3]}, {data[4]}, Rented\n"
invoice_name = **F(** "Invoice_", customer_name, "_", **FORMAT** now to string using format '%m-%d', ".txt")
OPEN invoice_name for writing as invoice
WRITE invoice header and details
PRINT "Printing Invoice!!!!!!!!!!!!!"
PRINT "Please Wait!!!!!!!!!!"
WAIT for 5 seconds
PRINT invoice details
PRINT "Land rented successfully!"
BREAK
ELSE
PRINT "Land not found or already rented."
OPEN file_name for writing as file
WRITE lines to file
EXCEPT FileNotFoundError
PRINT "File not found!"
EXCEPT Exception AS e
PRINT "An error occurred:", CONVERT e to string

- **For return.py**

IMPORT datetime
DEFINE return_land with parameters file_name, kitta_no, customer_name, rent_dur, return_dur, ph_number
now = **GET** current date and time
formatted_datetime = **FORMAT** now to string using format " %H:%M:%S"

```
    date = GET date from now
    TRY
        OPEN file_name for reading as file.
        READ lines from file
        FOR each line and index in lines
            IF line starts with kitta_no
                data = SPLIT line by ', '
                IF kitta_no equals data[0] AND data[5] equals "Rented"
                    UPDATE lines[i] to "data[0], data[1], data[2], data[3], data[4], Available\n"
                    invoice_name = CONCATENATE "Return_", customer_name, "_",
FORMAT now to string using format '%Y-%m-%d_%H-%M-%S', ".txt"
                    OPEN invoice_name for writing as invoice.
                    WRITE invoice header and details.
                    PRINT "Printing Invoice!!!!!!!!!!!!!!"
                    PRINT "Please Wait!!!!!!!!!!"
                    WAIT for 5 seconds.
                    PRINT invoice details.
                    PRINT "Land returned successfully!"
                    BREAK
        ELSE
            PRINT "Land not found or not rented."
        OPEN file_name for writing as file
        WRITE lines to file
    EXCEPT FileNotFoundError
        PRINT "File not found!"
    EXCEPT Exception AS e
        PRINT "An error occurred:", CONVERT e to string
```

- **Display_Info.py**

```
    DEFINE displayInfo with parameter filename.
```

```
    TRY
        gap = ' ' * 3
        heading = F( "| {'Kitta No.':<10s} | ", gap, "{'City/District':<20s} | ", gap, "{'Direction':<10s} | ", gap, "{'Area(anna)':<10s} | ", gap, "{'Price':<10s} | ", gap, "{'Availability':<15s} |")
        PRINT "=" for 108 times
        PRINT heading.
        PRINT "-" for 108 times.
        OPEN filename for reading as file.
        FOR each line in file
            SPLIT line by ','
            rec = F( "| ", data[0]:<10s, " | ", gap, data[1]:<20s, " | ", gap, data[2]:<10s, " | ", gap, data[3]:<10s, " | ", gap, data[4]:<10s, " | ", gap, data[5]:<15s, " |")
            PRINT rec
        PRINT "=" for 108 times
    EXCEPT FileNotFoundError AS exception
        PRINT "File not found"
        PRINT " "
```

5 Data structures

A data structure is a specialized format for organizing, processing, retrieving, and storing data. Data structures provide a efficient way of organizing and storing data. It provides a means to manage and manipulate collection of data, enabling operation such as insertion, deletion, addition, subtraction, searching etc. Python offers several in-built data structures. Each data structure is designed for specific use. Some of them are primitive data structures which include integer, float, string, and Boolean. Likewise, some advanced data structures include list, tuples, dictionary, and sets. Some of the data structures that are widely used in python are briefly described below:

- **Integer**

Integers represent whole numbers without decimal or fractional parts. It can be positive, negative, or null value. It represents a wide range of numerical values within the program. In python integer values are defined with “int” keywords. Python’s integers can handle very large and very small numbers without losing accuracy. Integers are commonly used in arithmetic operation, counting and indexing within data structures like lists and tuples. In this project I have used integer for calculating total amount, fine amount and getting duration from the user.

This is how I use integer:

```
 total_amount = int(data[4]) * rent_dur
 invoice.write(f"Total Amount without fine: {total_amount}\n")
 fine_amount = 0.1 * total_amount*(return_dur-rent_dur)
 total_amount += fine_amount
 invoice.write(f"Total Amount with fine: RS.{total_amount}\n")
f return_dur <= rent_dur:
 total_amount = int(data[4]) * rent_dur
 invoice.write(f"Total Amount: RS.{total amount}\n")
```

Figure 9: Use of integer.

- **String**

Strings in python are sequences of characters enclosed within single quotes (‘’) or doubles quotes (“”). They are immutable, meaning once they are created, it cannot be changed. String is widely used in programming and in python strings are defined using the keywords ‘str’. In python it is used to represent textual data such as names, messages, and file contents. They support various operations including concatenation, slicing, formatting, and searching. Python provides extensive string manipulation methods, making it easy to work with and manipulate text data efficiently. String is widely used in this coursework such as for getting customer name, address and message.

```
            break  # Exit the loop if the kitta number is valid
    customer_name = input("Enter your name: ")
```

Figure 10: Use of String

- **Boolean**

Booleans are the truth values. Booleans in python are fundamental data types used to represent truth values. They can only have only two possible values which are true or false. Booleans are the crucial parts for conditional statements and logical expression. Booleans are usually the result of comparison operation like greater then, smaller than, equals to or logical operations like or, not, etc. They are important for making decisions and control flow in python program. Boolean is used for controlling the flow of program in this coursework.

```
            if kitta_no == data[0]:
                if data[-1].strip() == "Rented":
                    return True
                else:
                    return False
    return False
```

Figure 11: use of Boolean.

- **Dictionary**

Dictionaries in python is a data structure, used to store values in key:value format. This makes it different from lists tuples, and arrays as in a dictionary each key has an associated value. (Geeks for Geeks, 2024). A dictionary associates key with values. Each key maps with a specific value. They are versatile data structures in python and are used to store collections of items in key value pairs. Each key in a dictionary is unique and is associated with a corresponding value. Dictionaries are mutable, allowing for the addition, modification, and deletion of key value pair. It is commonly used for representing structured data, such as user profiles, database records etc. Here is a example for dictionary:

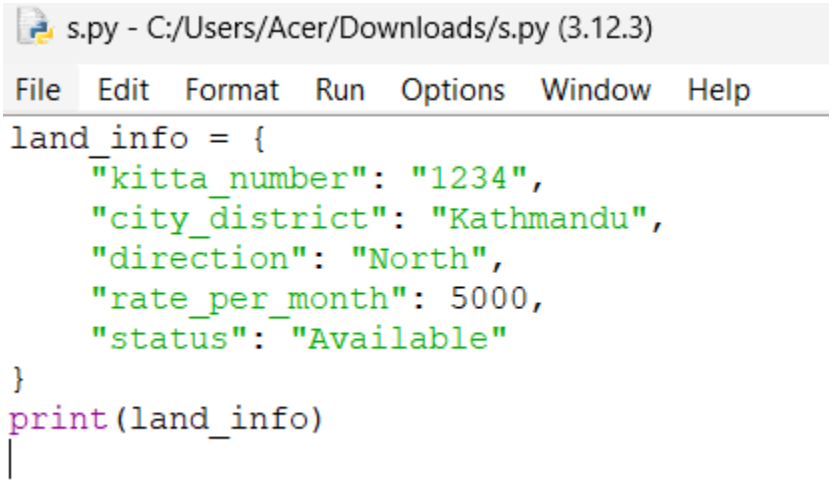


Figure 12: Example of Dictionary

- **List**

List in python is ordered collections of data. Lists are one of the built-in data types in python which are used to store collections of data which are created using square brackets. They are ordered, changeable and allow duplicate values. List items are indexed, the first items in the index have a value of zero and the second items in the index have a value of one and so on. List in python supports various operations such as appending, inserting, removing, and slicing elements. They are widely used for tasks like storing data, iterating over sequences, and implementing algorithms such as sorting and searching.

```
land_data = [
    ["1234", "Kathmandu", "North", 5000, "Available"],
    ["5678", "Pokhara", "South", 6000, "Rented"],
    ["91011", "Biratnagar", "East", 4500, "Available"]
]
print(land_data)
```

Figure 13: Example of list

6 Program

The program is designed for land renting and returning process. This type program helps to rent and return land. It consists of multiple functions. Some of the major functions are 'displayInfo' which is responsible for displaying land information stored in a file. The next function is 'rent_land' function which is responsible for handling the process of renting land. It takes input such as the kitta number, customer name, duration of rent, and phone number. It checks the requested land, if it is available the requested land will be rented and generates an invoice in text file and prints in console. Another function is return_land function which handles the process of returning rented land. It is similar to rent_land function. Fine is also calculated if the return duration exceeds the rent duration. The main function of the program is main. It is in a continuous loop until the user stops the loop. It repeatedly displays the menu, takes user input and calls the corresponding function based on the user's choice until the user chooses to exit. The program starts with welcome message, and then a menu message is displayed to the user instructing to input value as 1 for displaying available lands, 2 for renting the land, 3 for returning land and 4 for the existing the program.

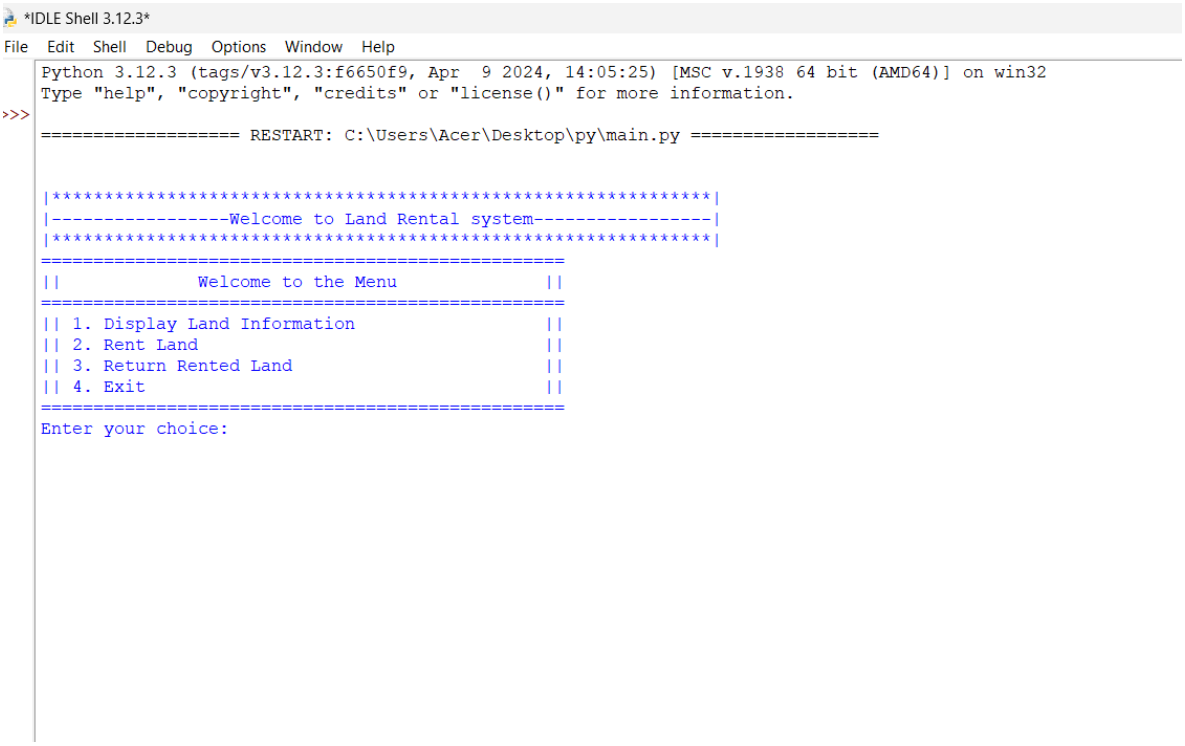


Figure 14: Welcome message.

If a user inputs any invalid inputs like strings or value except 1,2,3,4, an error message is displayed to the user that an invalid input has been entered and guide them to provide a valid input.

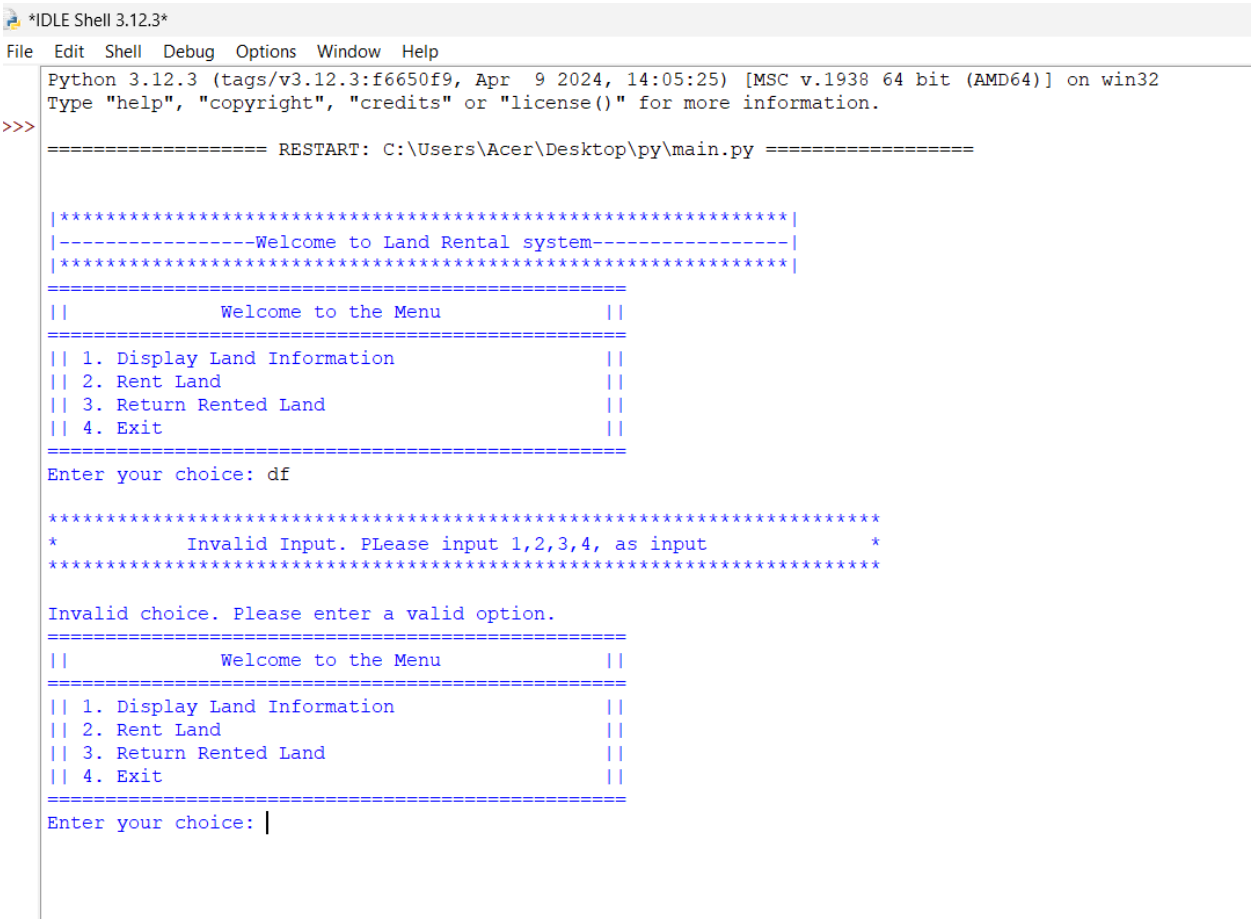


Figure 15: Invalid inputs and showing error message.

If a user selects to display land information from option 1, the program displays all the land which are stored in text file. It presents the data in text file in a structured manner, including details such as kitaa number, city/district, direction, area price and availability.

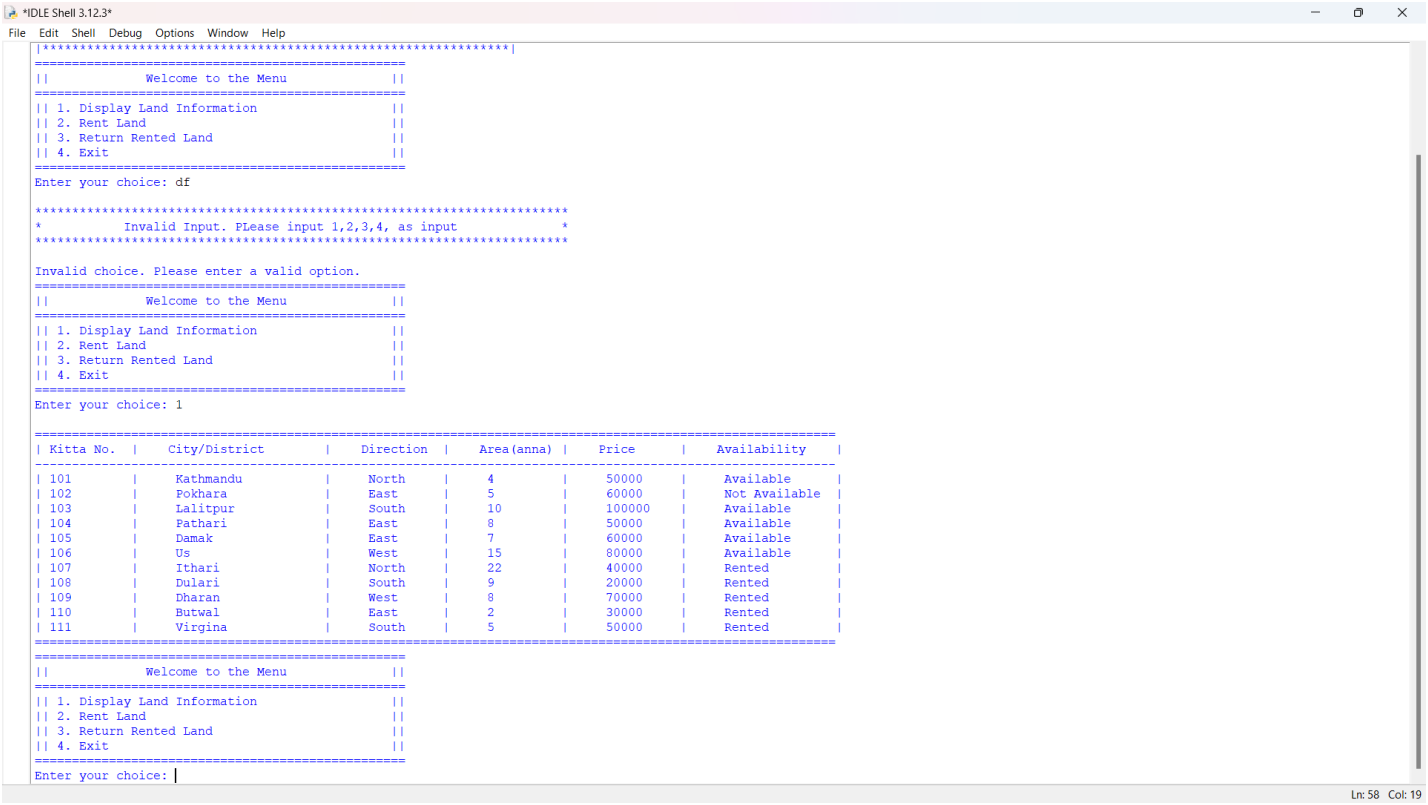


Figure 16: Displaying error message.

When the user selects option 2 to rent land, the program first asks to input the kitta number, their name, the duration of rent in month, and their phone number. After validating all the inputs, it checks if the specified land is available for rent by checking its status in the land file. If the land is available for rent, the program updates the file to mark as rented land. After that it generates invoice and also prints the invoice in terminal. This invoice includes information such as customer's name, phone number, rented land details, duration of rent, and total amount. Once the rental process is completed the program will return to the main menu and ready for further user interaction.

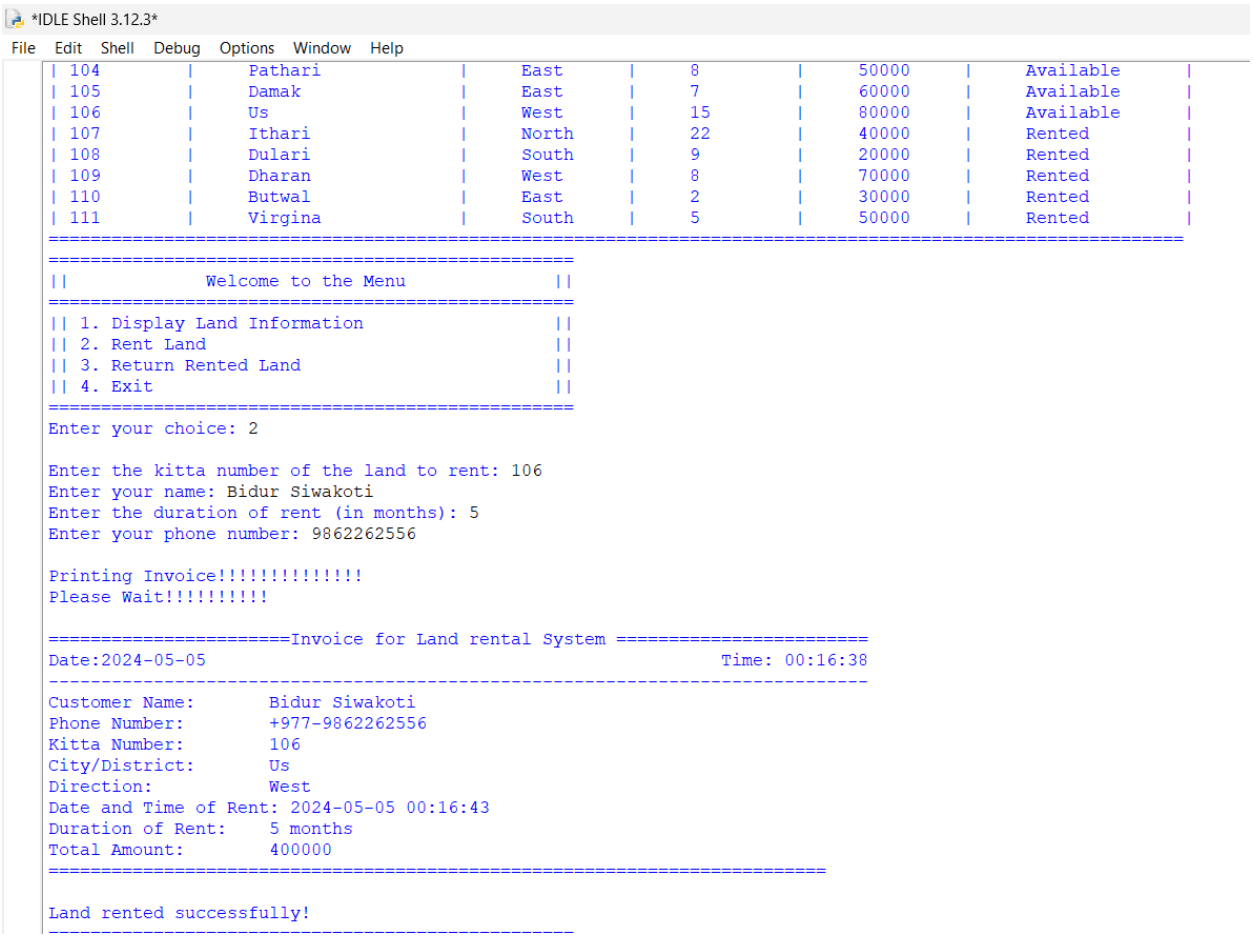


Figure 17: Completion of renting process.

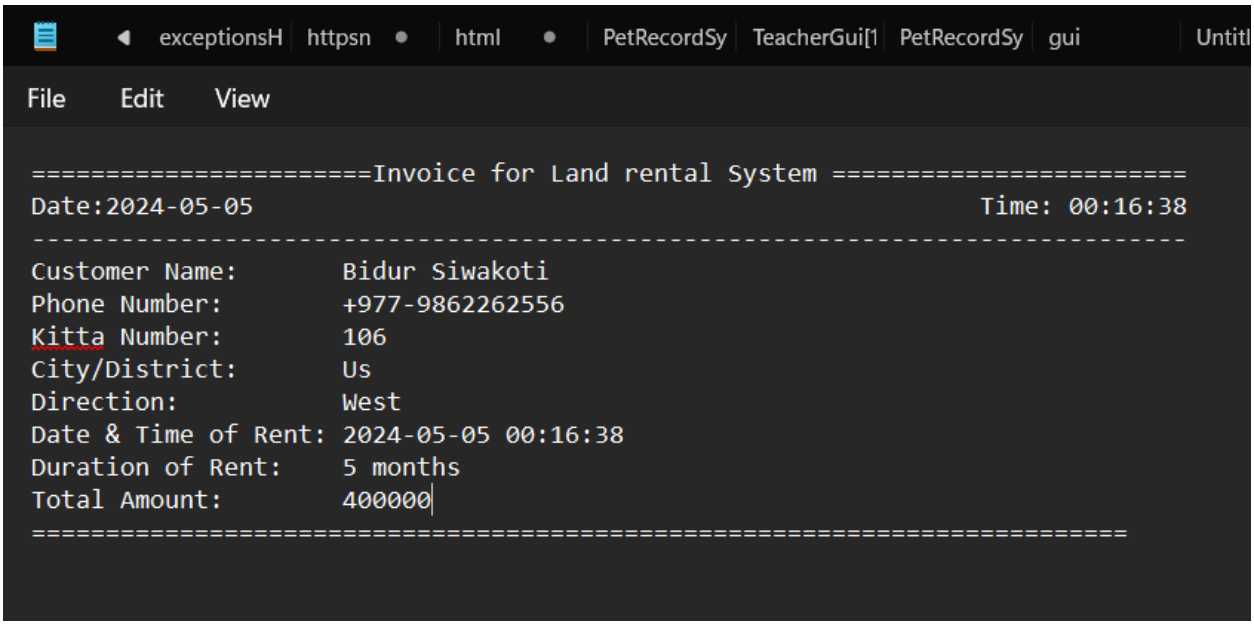


Figure 18: Invoice in text file.

When the user chooses 3 to return rented land, the program asks user to input kitta number of the land they are returning, their names, the duration of the rent, the duration of return, and their phone numbers. After validating all the inputs, it proceeds to check the status of the land. If the land is found to be rented, the program updates the file to mark it as available again and calculates the fines if the return duration exceeds the rent duration. After that, the program generates and prints the invoice in Text file and terminal respectively. The invoice includes information like customer's name, phone number, kitaa number, city or district, direction d ate and time etc. After completing the return process, the program returns to the main menu, allowing the user to select other options or exit the program.

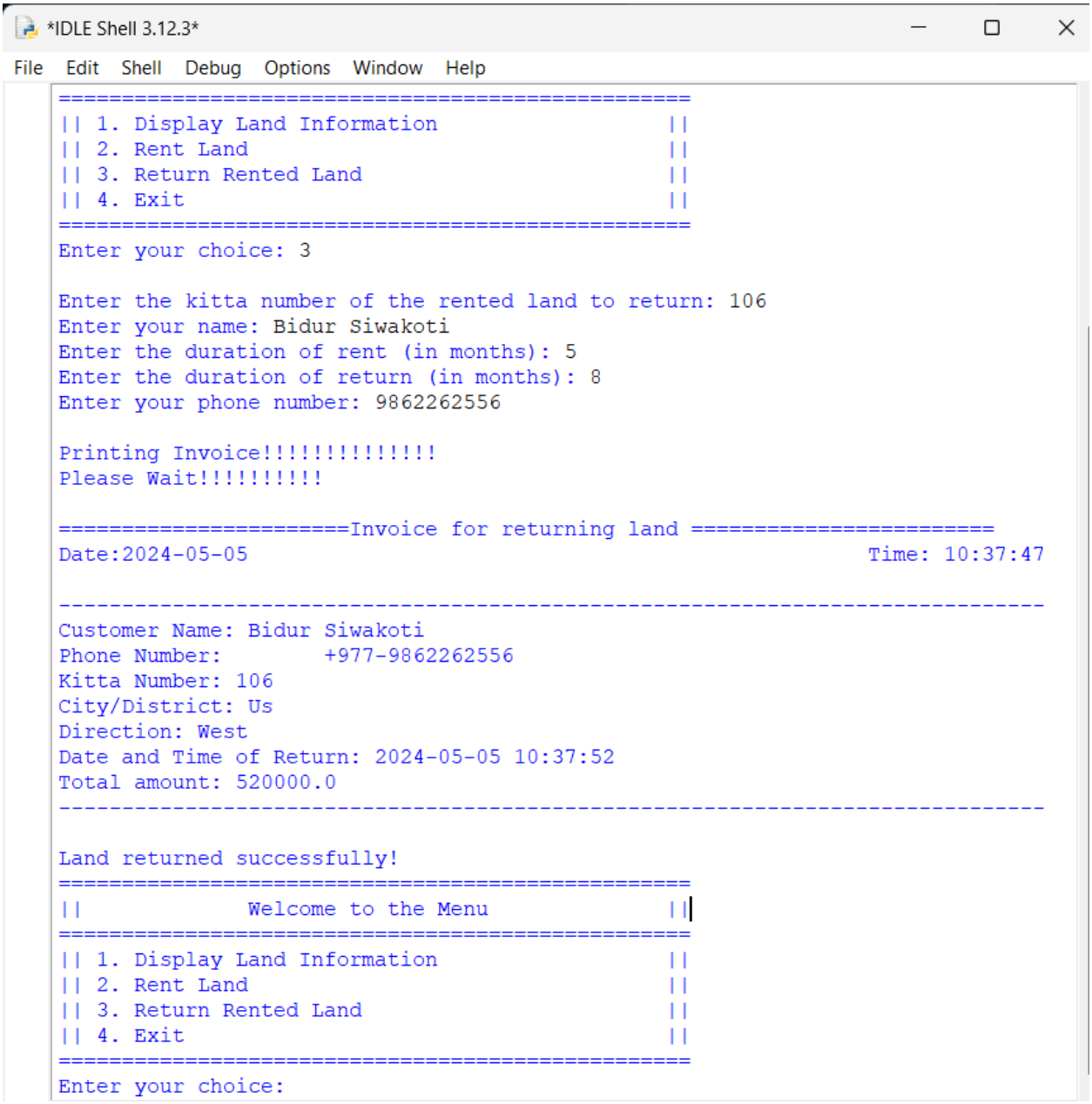


Figure 19: returning Invoice.

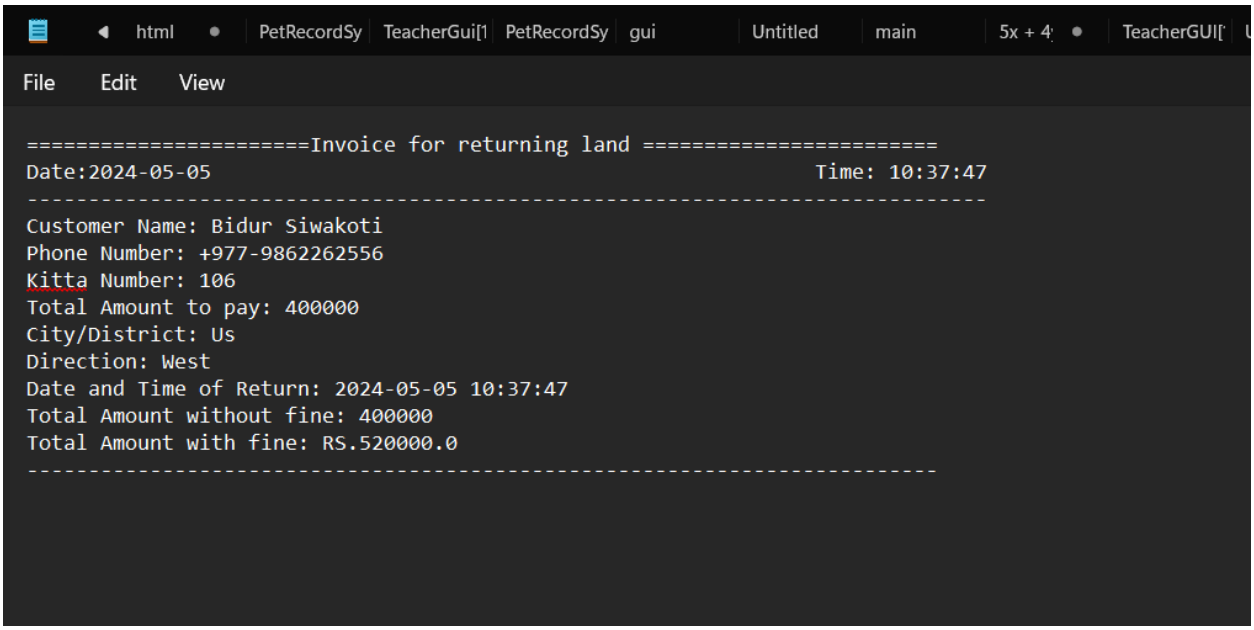


Figure 20: invoice in text file

When the user chooses option 4 to exit the program, the program displays a message, thanking the user for using the land rental system. Throughout the process, the program ensures validation of user input, handles exceptions and provides a seamless and user-friendly experiences for renting land within the system.

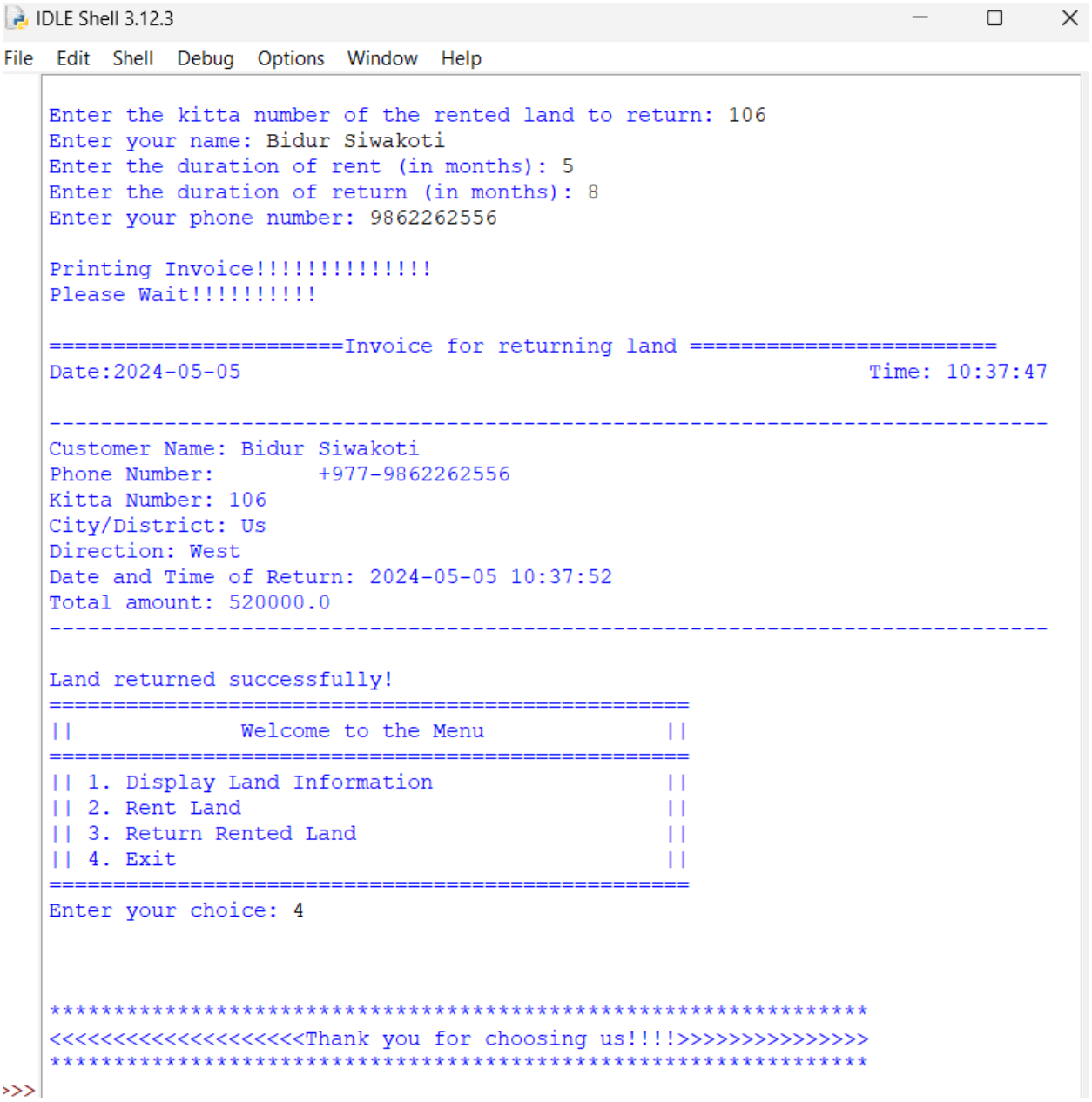


Figure 21: Closing program.

7 Testing

7.1 Test1:

Table 2: Test1: Try & Except implementation.

Objective:	To show the implementation of try, except
Action:	Run the program and enter the negative or string value for the duration. It will show error message and ask for re-input
Expected Result:	Error message should be displayed and ask for re-input
Actual result:	Error message saying “invalid duration” is displayed and ask for re-inputs
Conclusion	Test 1 is successful

The screenshots involve in this testing are:

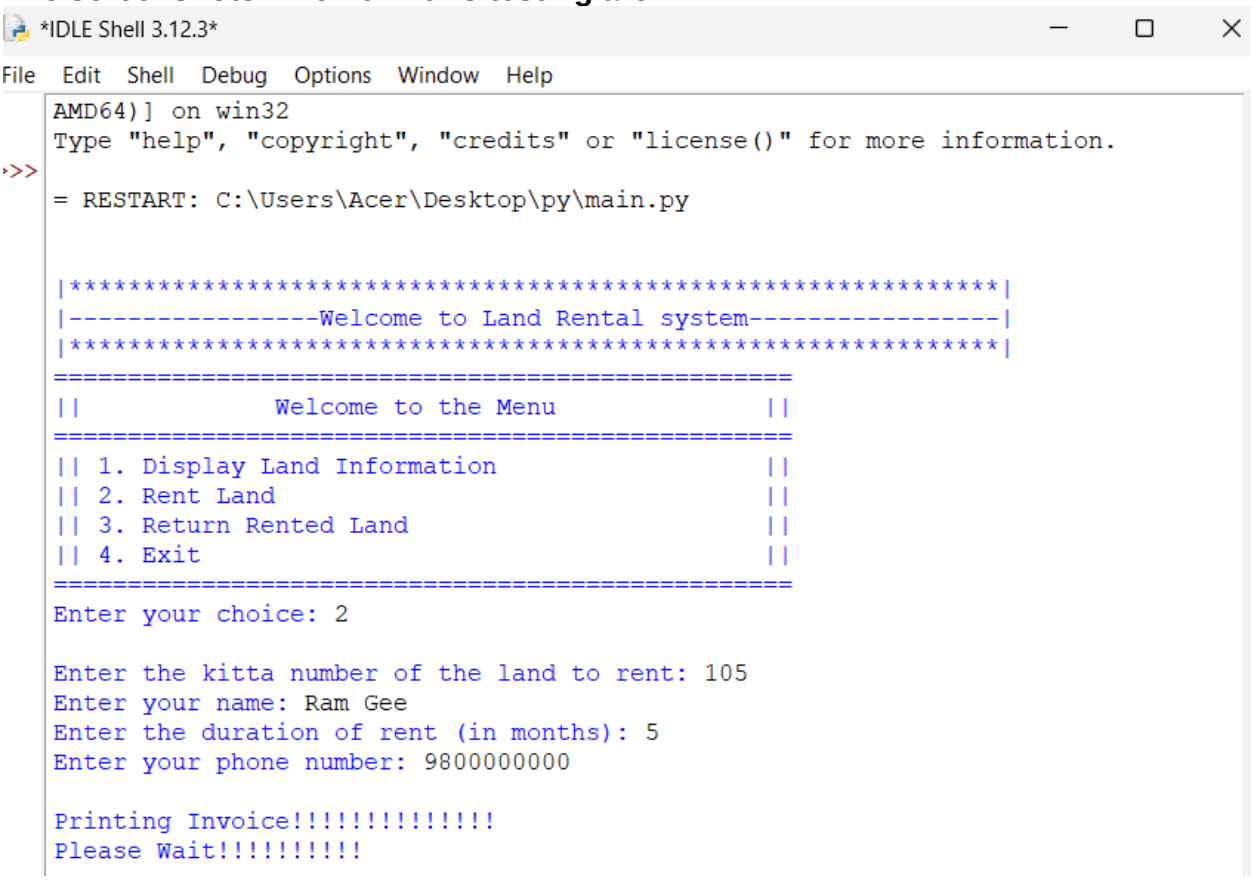


Figure 22: Running program as required Value.

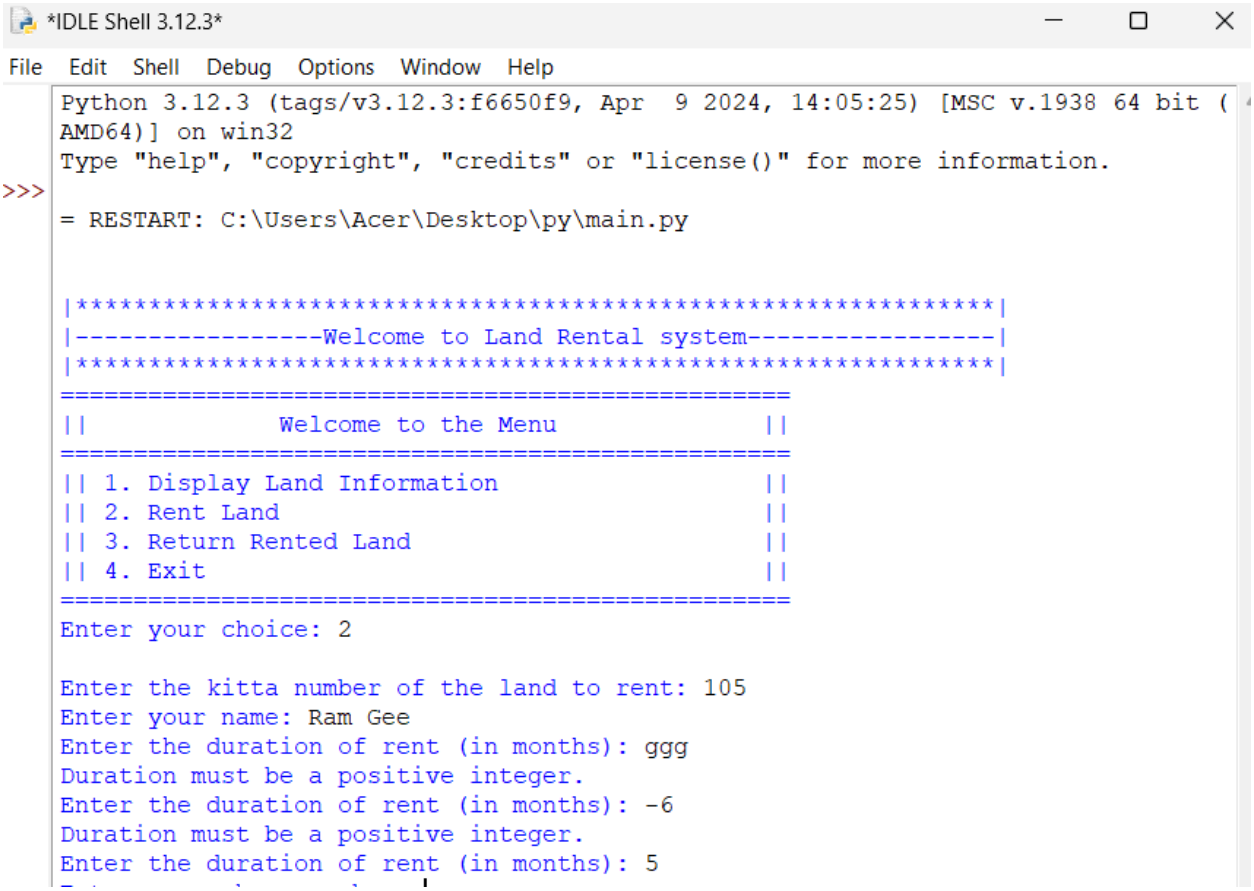


Figure 23: Running program with invalid inputs.

7.2 Test2:

Table 3: validation of input

Objective:	To provide non-existent or negative value for kitta number while renting or returning the land.
Action:	Open the program and select option 2. When it asks for kitta number, enter an invalid kitta number
Expected Result:	It should not accept those invalid kitta numbers and it will display an error message also asking for new input.
Actual result:	Yes, it does not accept those invalid kitta numbers and display an error message. It asks for new valid kitta number
Conclusion	

The screenshots involve in this testing are:

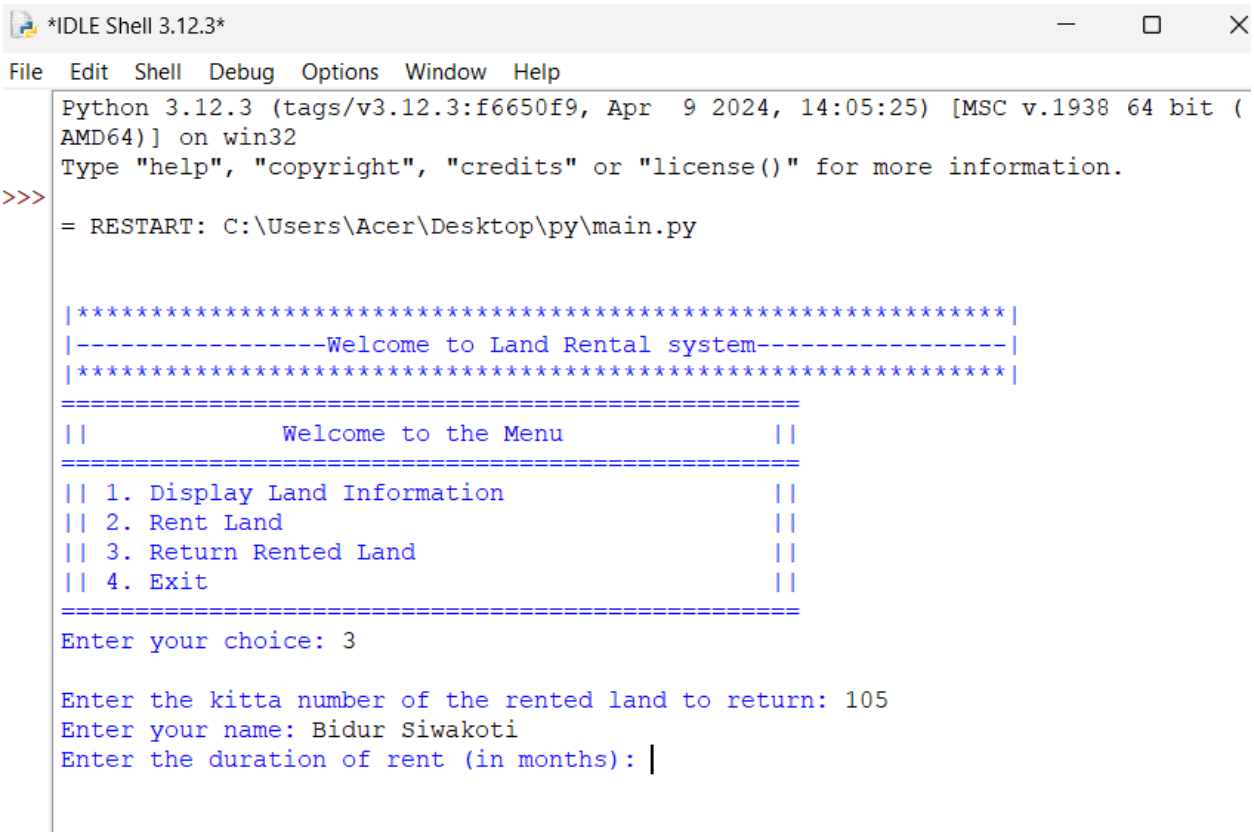


Figure 24: Executing the program with valid inputs.

```
|****************************************************************|
|-----------------Welcome to Land Rental system-----------------|
|****************************************************************|
==================================================
||             Welcome to the Menu              ||
==================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
==================================================
Enter your choice: 2

Enter the kitta number of the land to rent: 56
Kitta number not found. Please enter the valid kita number
Enter the kitta number of the land to rent: hjhj
Kitta number not found. Please enter the valid kita number
Enter the kitta number of the land to rent: 1111
Kitta number not found. Please enter the valid kita number
Enter the kitta number of the land to rent: -101
Kitta number not found. Please enter the valid kita number
Enter the kitta number of the land to rent: 105
Enter your name:
```

Figure 25: Executing program with Invalid Inputs will renting.

```
|*****************************************************************|
|-----------------Welcome to Land Rental system-----------------|
|*****************************************************************|
==================================================
||             Welcome to the Menu              ||
==================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
==================================================
Enter your choice: 3

Enter the kitta number of the rented land to return: 77
Kitta number not found. Please enter the valid kita number
Enter the kitta number of the rented land to return: kkjbkjlb
Kitta number not found. Please enter the valid kita number
Enter the kitta number of the rented land to return: aesrdtxfycguvhibjnk
Kitta number not found. Please enter the valid kita number
Enter the kitta number of the rented land to return: 55531351325425
Kitta number not found. Please enter the valid kita number
Enter the kitta number of the rented land to return: 105
Enter your name:
```

Figure 26: Executing program with Invalid Inputs will renting.

7.3 Test 3

Table 4:Test3: To update stock file.

Objective:	To the update of stock file.
Action:	Open the program, it will ask for the user choice enter 1 for displaying all the lands information
Expected Result:	It should update the file.
Actual result:	Yes, it updates.
Conclusion	Test 3 is successful.

The screenshots involve in this testing are:

```
==================================================
||             Welcome to the Menu              ||
==================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
==================================================
Enter your choice: 2

Enter the kitta number of the land to rent: 106
Enter your name: Bidur siwakoti
Enter the duration of rent (in months): 5
Enter your phone number: 9800000000

Printing Invoice!!!!!!!!!!!!!
Please Wait!!!!!!!!!!

=======================Invoice for Land rental System ========================
Date:2024-05-05                                                  Time: 23:27:39
--------------------------------------------------------------------------------
Customer Name:       Bidur siwakoti
Phone Number:        +977-9800000000
Kitta Number:        106
City/District:       Us
Direction:           West
Date and Time of Rent: 2024-05-05 23:27:44
Duration of Rent:    5 months
Total Amount:        400000
================================================================================

Land rented successfully!
==================================================
||             Welcome to the Menu              ||
==================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
==================================================
Enter your choice:
```

Figure 27: Renting kitta process.

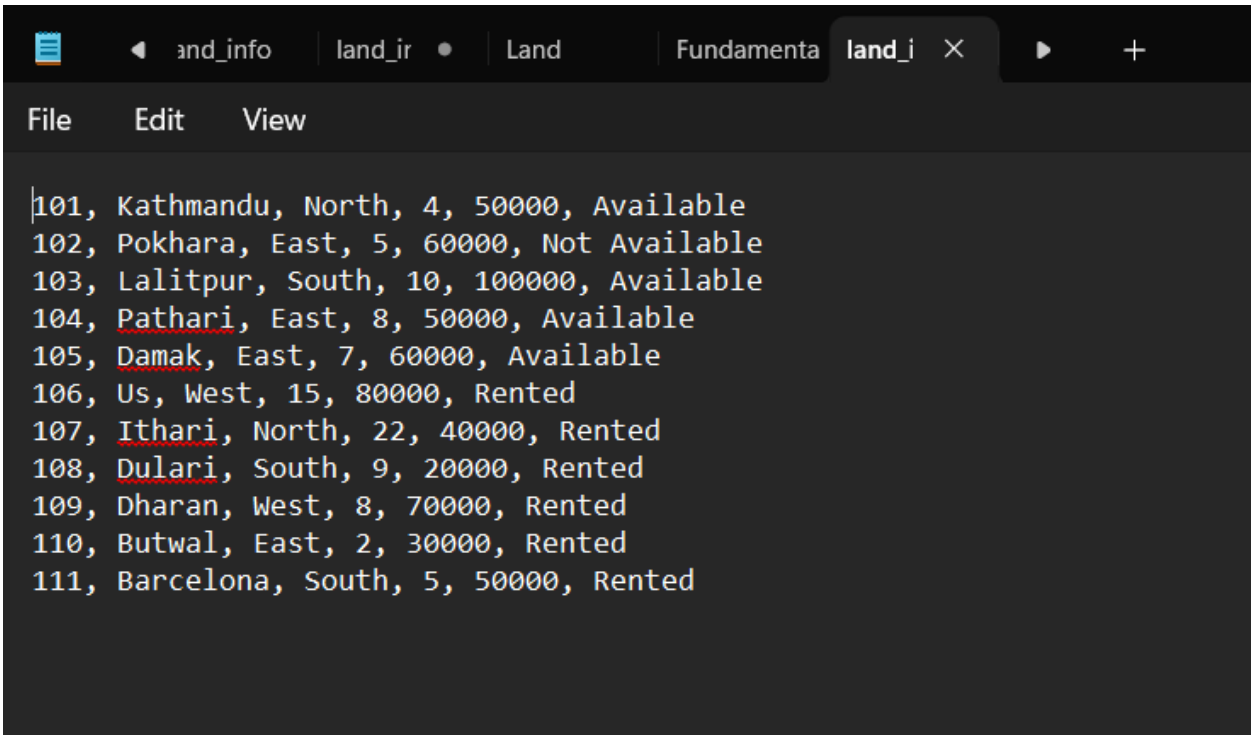


Figure 28: changed to rented

```
Edit  Shell  Debug  Options  Window  Help
Customer Name:        Bidur siwakoti
Phone Number:         +977-9800000000
Kitta Number:         106
City/District:        Us
Direction:            West
Date and Time of Rent: 2024-05-05 23:27:44
Duration of Rent:     5 months
Total Amount:         400000
================================================================================

Land rented successfully!
==================================================
||             Welcome to the Menu              ||
==================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
==================================================
Enter your choice: 3

Enter the kitta number of the rented land to return: 106
Enter your name: Bidur siwakoti
Enter the duration of rent (in months): 5
Enter the duration of return (in months): 6
Enter your phone number: 9800000000

Printing Invoice!!!!!!!!!!!!!
Please Wait!!!!!!!!!!

=======================Invoice for returning land ========================
Date:2024-05-05                                                Time: 23:30:16

--------------------------------------------------------------------------------
Customer Name: Bidur siwakoti
Phone Number:         +977-9800000000
Kitta Number: 106
City/District: Us
Direction: West
Date and Time of Return: 2024-05-05 23:30:21
Total amount: 440000.0
--------------------------------------------------------------------------------

Land returned successfully!
==================================================
||             Welcome to the Menu              ||
==================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
```

Figure 29: Returning process

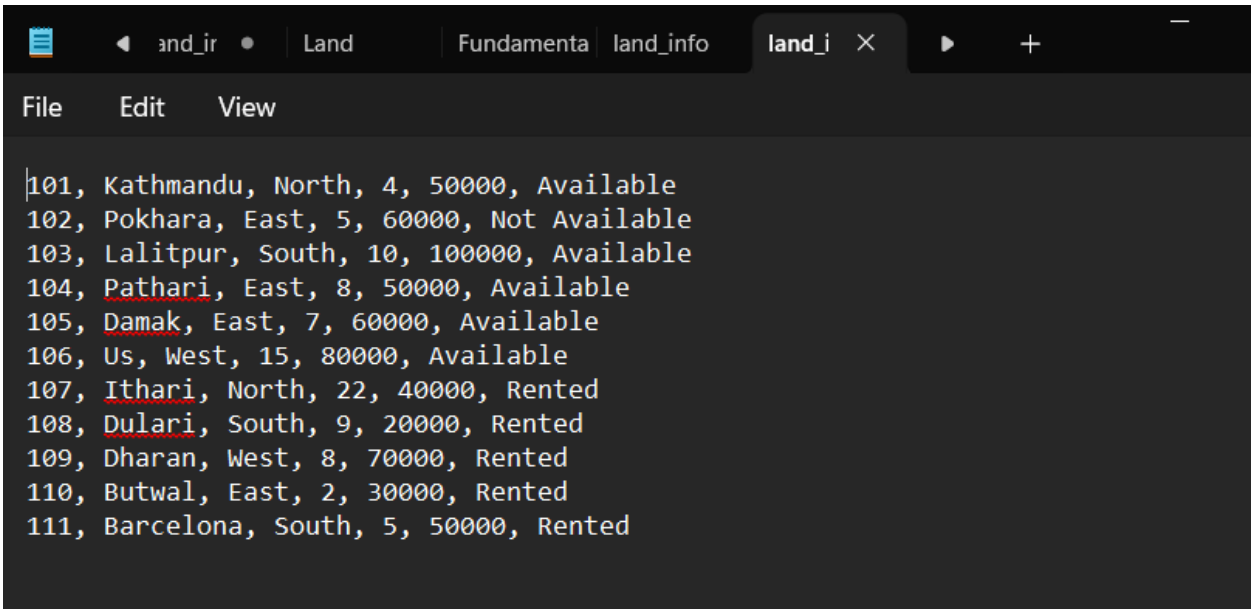


Figure 30: Changing status

7.4 Test 4:

Table 5: Test4: To rent the available land.

Objective:	To rent the available land
Action:	From the option menu enter 2 for renting Fill in the required details for renting the land. Print the bill in the terminal. Views the bills in Text file.
Expected Result:	Land must be rented, and invoices must be printed and generated in a text file. It should also change the status of land to rented.
Actual result:	Land is rented and invoice is printed and generated in text file.
Conclusion	Test 4 is successful.

The screenshots involve in this testing are:

```
= RESTART: C:\Users\Acer\Desktop\py\main.py


|*****************************************************************|
|-----------------Welcome to Land Rental system-----------------|
|*****************************************************************|
==================================================
||             Welcome to the Menu              ||
==================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
==================================================
Enter your choice: 1

=============================================================================================================
| Kitta No.  |    City/District        |    Direction  |    Area(anna) |    Price      |    Availability    |
-------------------------------------------------------------------------------------------------------------
| 101        |     Kathmandu           |     North     |     4         |     50000     |     Available      |
| 102        |     Pokhara             |     East      |     5         |     60000     |     Not Available  |
| 103        |     Lalitpur            |     South     |     10        |     100000    |     Available      |
| 104        |     Pathari             |     East      |     8         |     50000     |     Available      |
| 105        |     Damak               |     East      |     7         |     60000     |     Available      |
| 106        |     Us                  |     West      |     15        |     80000     |     Available      |
| 107        |     Ithari              |     North     |     22        |     40000     |     Rented         |
| 108        |     Dulari              |     South     |     9         |     20000     |     Rented         |
| 109        |     Dharan              |     West      |     8         |     70000     |     Rented         |
| 110        |     Butwal              |     East      |     2         |     30000     |     Rented         |
| 111        |     Barcelona           |     South     |     5         |     50000     |     Rented         |
=============================================================================================================
==================================================
||             Welcome to the Menu              ||
==================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
==================================================
Enter your choice: 2

Enter the kitta number of the land to rent: 105
Enter your name: Niraj Poudel
Enter the duration of rent (in months): 5
Enter your phone number: 9812345678

Printing Invoice!!!!!!!!!!!!!!
Please Wait!!!!!!!!!!
```

Figure 31: Filling required details for renting land.

```
Printing Invoice!!!!!!!!!!!!!!
Please Wait!!!!!!!!!!

=======================Invoice for Land rental System ========================
Date:2024-05-05                                                  Time: 11:44:34
--------------------------------------------------------------------------------
Customer Name:       Niraj Poudel
Phone Number:        +977-9812345678
Kitta Number:        105
City/District:       Damak
Direction:           East
Date and Time of Rent: 2024-05-05 11:44:39
Duration of Rent:    5 months
Total Amount:        300000
================================================================================

Land rented successfully!
=====================================================
||             Welcome to the Menu                ||
=====================================================
|| 1. Display Land Information                     ||
|| 2. Rent Land                                    ||
|| 3. Return Rented Land                           ||
|| 4. Exit                                         ||
=====================================================
Enter your choice:
```

Figure 32: Invoice in terminal.

change.log | index.html | new 1.java | new 1.py | TeacherGUI.java | Invoice_Niraj Poudel_05-05.txt

```
1   =======================Invoice for Land rental System ========================
2   Date:2024-05-05                                                  Time: 11:44:34
3   --------------------------------------------------------------------------------
4   Customer Name:       Niraj Poudel
5   Phone Number:        +977-9812345678
6   Kitta Number:        105
7   City/District:       Damak
8   Direction:           East
9   Date & Time of Rent: 2024-05-05 11:44:34
10  Duration of Rent:    5 months
11  Total Amount:        300000
12  ================================================================================
```

Figure 33: Invoice In text file.

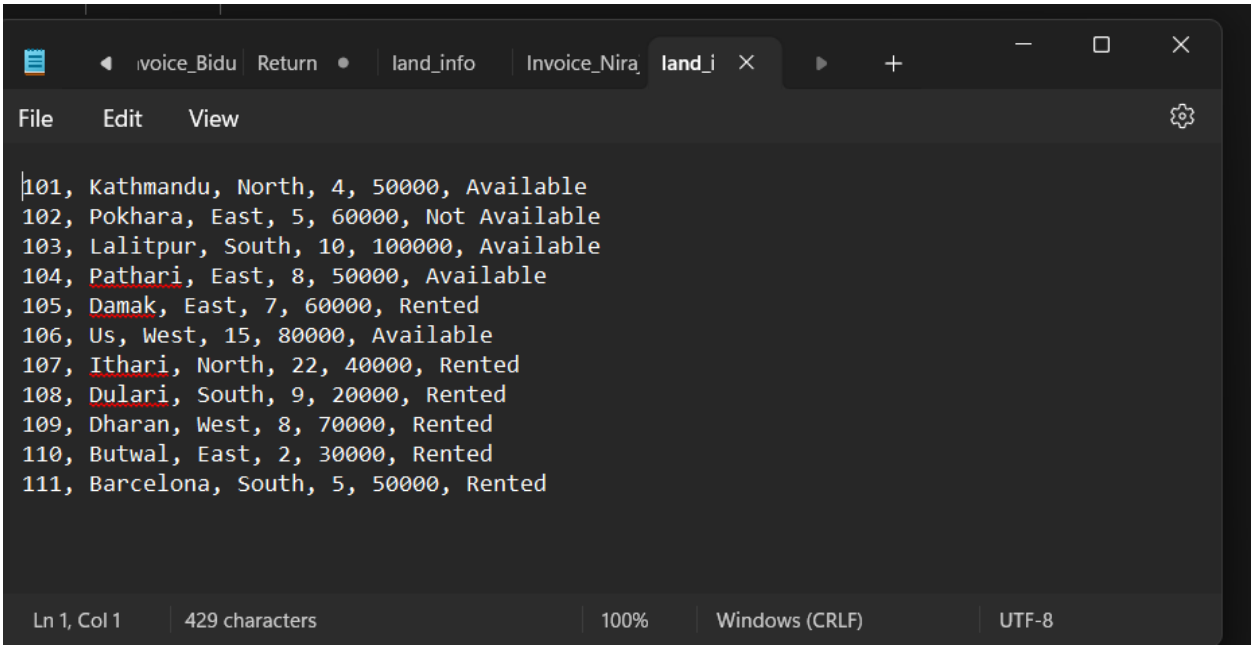


Figure 34: Status of land changed to rented

7.5 Test 5:

Table 6: Test5: To return land

Objective:	To return the rented land.
Action:	From the option menu enter 3 for returning of land Fill in the required details for returning the land. Print the bill in the terminal. Views the bills in Text file.
Expected Result:	Land must be returned, and invoices must be printed and generated in a text file. It should also change the status of land to available.
Actual result:	Land is rented and invoice is printed and generated in text file.
Conclusion	Test 5 is successful

The screenshots involve in this testing are:

```
Enter your choice: 1

=============================================================================================================
| Kitta No.  |    City/District        |    Direction  |    Area(anna) |    Price      |    Availability    |
-------------------------------------------------------------------------------------------------------------
| 101        |     Kathmandu           |     North     |     4         |     50000     |     Available      |
| 102        |     Pokhara             |     East      |     5         |     60000     |     Not Available  |
| 103        |     Lalitpur            |     South     |     10        |     100000    |     Available      |
| 104        |     Pathari             |     East      |     8         |     50000     |     Available      |
| 105        |     Damak               |     East      |     7         |     60000     |     Rented         |
| 106        |     Us                  |     West      |     15        |     80000     |     Available      |
| 107        |     Ithari              |     North     |     22        |     40000     |     Rented         |
| 108        |     Dulari              |     South     |     9         |     20000     |     Rented         |
| 109        |     Dharan              |     West      |     8         |     70000     |     Rented         |
| 110        |     Butwal              |     East      |     2         |     30000     |     Rented         |
| 111        |     Barcelona           |     South     |     5         |     50000     |     Rented         |
=============================================================================================================
================================================
||             Welcome to the Menu              ||
================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
================================================
Enter your choice: 3

Enter the kitta number of the rented land to return: 105
Enter your name: Niraj poudel
Enter the duration of rent (in months): 5
Enter the duration of return (in months): 8
Enter your phone number: 9812345678

Printing Invoice!!!!!!!!!!!!!
Please Wait!!!!!!!!!!
```

Figure 35: Filling all the required details for renting land.

```
================================================
||             Welcome to the Menu              ||
================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
================================================
Enter your choice: 3

Enter the kitta number of the rented land to return: 105
Enter your name: Niraj poudel
Enter the duration of rent (in months): 5
Enter the duration of return (in months): 8
Enter your phone number: 9812345678

Printing Invoice!!!!!!!!!!!!!!
Please Wait!!!!!!!!!!

=======================Invoice for returning land ========================
Date:2024-05-05                                                Time: 11:58:22

--------------------------------------------------------------------------------
Customer Name: Niraj poudel
Phone Number:        +977-9812345678
Kitta Number: 105
City/District: Damak
Direction: East
Date and Time of Return: 2024-05-05 11:58:27
Total amount: 390000.0
--------------------------------------------------------------------------------

Land returned successfully!
================================================
||             Welcome to the Menu              ||
================================================
|| 1. Display Land Information                  ||
|| 2. Rent Land                                 ||
|| 3. Return Rented Land                        ||
|| 4. Exit                                      ||
================================================
Enter your choice:
```

Figure 36: Invoice for returning land.

Invoice_Niraj Poudel_05-05.txt | Return_Niraj poudel_2024-05-05_11-58-22.txt

```
1   ========================Invoice for returning land ========================
2   Date:2024-05-05                                          Time: 11:58:22
3   ------------------------------------------------------------------------------
4   Customer Name: Niraj poudel
5   Phone Number: +977-9812345678
6   Kitta Number: 105
7   Total Amount to pay: 300000
8   City/District: Damak
9   Direction: East
10  Date and Time of Return: 2024-05-05 11:58:22
11  Total Amount without fine: 300000
12  Total Amount with fine: RS.390000.0
13  ------------------------------------------------------------------------
14
15
```

Figure 37: Invoice text file.

eturn | land_info | Invoice_Nira | land_info | land_i

File Edit View

```
101, Kathmandu, North, 4, 50000, Available
102, Pokhara, East, 5, 60000, Not Available
103, Lalitpur, South, 10, 100000, Available
104, Pathari, East, 8, 50000, Available
105, Damak, East, 7, 60000, Available
106, Us, West, 15, 80000, Available
107, Ithari, North, 22, 40000, Rented
108, Dulari, South, 9, 20000, Rented
109, Dharan, West, 8, 70000, Rented
110, Butwal, East, 2, 30000, Rented
111, Barcelona, South, 5, 50000, Rented
```

Ln 1, Col 1 | 432 characters | 100% | Windows (CRLF) | UTF-8

Figure 38: Land file after returning land.

8 Conclusion

Finally, I reached the conclusion part of this coursework, the closing section of this report. The report is the documentation of a program developed for the module 'fundamental of computing'. At the starting point of this coursework, the coursework seemed a bit confusing for me. In this coursework we are assigned to develop a land rental system through which we can rent and return the land. We need to design an errorless and functional system.

At the beginning I did not even understand the question for this coursework but with the help me my module leader, tutor, and other online means I got it. Since this was our first assessment given to us in this module, and python being new to me, I faced multiple difficulties completing this coursework. I encountered many bugs and errors related to function, logic and variables which makes my program useless but with the help of teacher, module leader and friends I make my program more functional. Our teacher and tutor guided us throughout the completion of the program. We also received study materials which contain necessary information to develop the land rental system.

Finally, it was very nice and important coursework that taught me many things that I had not learned before. From this coursework I got the platform to use theoretical programming knowledge in real life scenarios. It provided a comprehensive exploration of fundamental concepts in computing, including data structure, algorithms and programming techniques. I learned file handling from this coursework along with functional programming, error handling as well as exceptional handling. This module has been a great journey for me. I have gained a lot of new knowledge that I never knew before. I am grateful for this module and the teacher. Finally, I appreciate the entire course for providing me with this valuable learning experience.

9 References

Geeks for Geeks, 2021. *Introduction to ms-word.* [Online]
Available at: https://www.geeksforgeeks.org/introduction-to-microsoft-word/
[Accessed 05 04 2024].
Geeks for Geeks, 2023. *Python Introduction.* [Online]
Available at: https://www.geeksforgeeks.org/introduction-to-python/
[Accessed 16 4 2024].
Geeks for Geeks, 2023. *What is Algorithm.* [Online]
Available at: https://www.geeksforgeeks.org/introduction-to-algorithms/
[Accessed 20 4 2024].
Geeks for Geeks, 2024. *Dictionaries in python.* [Online]
Available at: https://www.geeksforgeeks.org/python-dictionary/
[Accessed 22 4 2024].
Hebb, N., n.d. *What is a Flowcharts?.* [Online]
Available at: https://www.breezetree.com/articles/what-is-a-flow-chart
[Accessed 20 4 2024].
HIgssoftware, n.d. *what is python idle?.* [Online]
Available at: http://higssoftware.com/what-is-python-idle.php
[Accessed 04 05 2024].
The economic times, 2021. *What is pseudocode.* [Online]
Available at: https://economictimes.indiatimes.com/definition/pseudocode
[Accessed 20 04 2024].

10 Appendix:

10.1 Main.py Source code

```
# main
import time

from display_land_info import displayInfo
from rent_land import rent_land
from return_land import return_land

def print_header():

print("\n\n|*************************************************************|"
)
    print("|----------------Welcome to Land Rental system--------------
---|")

print("|*************************************************************|")

def print_menu():
    border = "=" * 50
    print(border)
    time.sleep(1)
    print("||{:^46}||".format("Welcome to the Menu"))
    time.sleep(0.8)
    print(border)
    time.sleep(0.6)
    print("|| {:<44} ||".format("1. Display Land Information"))
    time.sleep(0.7)
    print("|| {:<44} ||".format("2. Rent Land"))
    time.sleep(0.4)
    print("|| {:<44} ||".format("3. Return Rented Land"))
    time.sleep(0.2)
    print("|| {:<44} ||".format("4. Exit"))
    time.sleep(0.1)
    print(border)

def print_footer():

print("\n\n*************************************************************")
    print("<<<<<<<<<<<<<<<<<<<Thank you for choosing
us!!!!>>>>>>>>>>>>>>>")

print("*************************************************************")


def char():
```

```
 38.     print("-----------------------------------------------------------
---")
 39.
 40.
 41. def error_input():
 42.     print(
 43.
"*************************************************************************\n"
 44.         "*           Invalid Input. PLease input 1,2,3,4, as input
*\n"
 45.
"*************************************************************************\n"
 46.     )
 47.
 48. def validate_kitta_number(kitta_no, land_data):
 49.         """
 50.         This is the validation function designed to validate the kitta
number input by the user
 51.          returns true if the kitta number exist in file otherwise it is
false
 52.         """
 53.         for line in land_data:
 54.             data = line.strip().split(',')
 55.             if kitta_no in data:
 56.                 return True
 57.         return False
 58.
 59.
 60. def validate_duration(duration):
 61.     """
 62.     Validate the duration of rent and return provided by the user.
returns True if the duration is a positive integer, False otherwise.
 63.     """
 64.     while True:
 65.                 try:
 66.                     duration = int(duration)
 67.                     if duration > 0:
 68.                         return duration
 69.                     else:
 70.                         print("Duration must be a positive integer.")
 71.                 except ValueError:
 72.                     print("Duration must be a positive integer.")
 73.                 duration = input("Enter the duration of rent (in
months): ")
 74.
 75.
 76.
 77.
 78. def is_land_rented(kitta_no, file_name):
 79.     """
```

```
    Check if the land with the given kitta number is already rented.
    Returns True if the land is rented, False otherwise.
    """
    with open(file_name, 'r') as file:
        for line in file:
            data = line.strip().split(',')
            if kitta_no == data[0]:
                if data[-1].strip() == "Rented":
                    return True
                else:
                    return False
    return False


def main():
    """ The main method of the program from where all the function are called in loop"""
    file_name = "land_info.txt"
    print_header()
    while True:

        print_menu()

        choice = input("Enter your choice: ")
        print()

        if choice == "1":
            displayInfo(file_name)
        elif choice == "2":
            while True:
                kitta_no = input("Enter the kitta number of the land to rent: ")
                if validate_kitta_number(kitta_no, open(file_name)):
                    break  # Exit the loop if the kitta number is valid
            customer_name = input("Enter your name: ")
            duration = validate_duration(input("Enter the duration of rent (in months): "))
            ph_number = input("Enter your phone number: ")
            rent_land(file_name, kitta_no, customer_name, duration, ph_number)
        elif choice == "3":
            while True:
                kitta_no = input("Enter the kitta number of the rented land to return: ")
                if validate_kitta_number(kitta_no, open(file_name)):
                    break  # Exit the loop if the kitta number is valid
            customer_name = input("Enter your name: ")
            rent_dur = validate_duration(input("Enter the duration of rent (in months): "))
```

```
123.             return_dur = validate_duration(input("Enter the duration of
return (in months): "))
124.             ph_number = input("Enter your phone number: ")
125.             return_land(file_name, kitta_no, customer_name, rent_dur,
return_dur, ph_number)
126.         elif choice == "4":
127.             print_footer()
128.             break
129.         else:
130.             error_input()
131.             print("Invalid choice. Please enter a valid option.")
132.
133.
134. if __name__ == "__main__":
135.     main()
136.
```

10.2 Rent.py source code

```
 1. import time
 2. # rent
 3. from datetime import datetime
 4.
 5. def rent_land(file_name, kitta_no, customer_name, duration, ph_number):
 6.     """The function is designed for renting the available land on
land_info.txt  """
 7.     now = datetime.now()
 8.     formatted_datetime = now.strftime(" %H:%M:%S")
 9.     date = now.date()
10.     try:
11.         with open(file_name, 'r') as file:
12.             lines = file.readlines()
13.         for i, line in enumerate(lines):
14.             if line.startswith(kitta_no):
15.                 data = line.strip().split(', ')
16.                 if data[5] == "Available" and kitta_no == data[0]:
17.                     lines[i] = f"{data[0]}, {data[1]}, {data[2]},
{data[3]}, {data[4]}, Rented\n"
18.                     invoice_name =
f"Invoice_{customer_name}_{datetime.now().strftime('%m-%d')}.txt"
19.                     with open(invoice_name, 'w') as invoice:
20.                         # Invoice details
21.
22.                         invoice.write("======================Invoice for
Land rental System ========================\n")
23.                         invoice.write(f"Date:{date}
\t\t\t\t\t\t\t\t\tTime:{formatted_datetime}\n" )
24.                         invoice.write("----------------------------------
-----------------------------------------\n")
```

```
25.                         invoice.write(f"{'Customer Name:':<20}
{customer_name}\n")
26.                         invoice.write(f"{'Phone Number:':<20} +977-
{ph_number}\n")
27.                         invoice.write(f"{'Kitta Number:':<20}
{data[0]}\n")
28.                         invoice.write(f"{'City/District:':<20}
{data[1]}\n")
29.                         invoice.write(f"{'Direction:':<20} {data[2]}\n")
30.                         invoice.write(f"{'Date & Time of Rent:':<20}
{datetime.now().strftime('%Y-%m-%d %H:%M:%S')}\n")
31.                         invoice.write(f"{'Duration of Rent:':<20}
{duration} months\n")
32.                         total_amount = int(data[4]) * duration
33.                         invoice.write(f"{'Total Amount:':<20}
{total_amount}\n")
34.                         print("\nPrinting Invoice!!!!!!!!!!!!!!")
35.                         print("Please Wait!!!!!!!!!!\n")
36.                         time.sleep(5)
37.
invoice.write("================================================================
============")
38.                         print("======================Invoice for Land
rental System ========================")
39.                         print(f"Date:{date}
\t\t\t\t\t\tTime:{formatted_datetime}" )
40.                         print("----------------------------------------
-----------------------------------")
41.                         print(f"{'Customer Name:':<20} {customer_name}")
42.                         print(f"{'Phone Number:':<20} +977-{ph_number}")
43.                         print(f"{'Kitta Number:':<20} {data[0]}")
44.                         print(f"{'City/District:':<20} {data[1]}")
45.                         print(f"{'Direction:':<20} {data[2]}")
46.                         print(f"{'Date and Time of Rent:':<20}
{datetime.now().strftime('%Y-%m-%d %H:%M:%S')}")
47.                         print(f"{'Duration of Rent:':<20} {duration}
months")
48.                         print(f"{'Total Amount:':<20} {total_amount}")
49.
print("================================================================
====")
50.                         print("\nLand rented successfully!")
51.                     break
52.         else:
53.             print("Land not found or already rented.")
54.
55.         with open(file_name, 'w') as file:
56.             file.writelines(lines)
57.     except FileNotFoundError:
58.         print("File not found!")
```

```
59.     except Exception as e:
60.         print("An error occurred:", str(e))
61.
```

10.3 Return.py source code

```
 1. import time
 2. from datetime import datetime
 3. def return_land(file_name, kitta_no, customer_name, rent_dur, return_dur,
ph_number):
 4.     """ This is the function for returning the rented land.
 5.     The arguments used in this function are file_name, kitta_no,
customer_name, ret_dur, return_dur, ph_number"""
 6.     now = datetime.now()
 7.     formatted_datetime = now.strftime(" %H:%M:%S")
 8.     date = now.date()
 9.     try:
10.         with open(file_name, 'r') as file:
11.             lines = file.readlines()
12.         for i, line in enumerate(lines):
13.             if line.startswith(kitta_no):
14.                 data = line.strip().split(', ')
15.                 if kitta_no == data[0] and data[5] == "Rented":
16.                     lines[i] = f"{data[0]}, {data[1]}, {data[2]},
{data[3]}, {data[4]}, Available\n"
17.                     invoice_name =
f"Return_{customer_name}_{datetime.now().strftime('%Y-%m-%d_%H-%M-%S')}.txt"
18.                     with open(invoice_name, 'w') as invoice:
19.                         invoice.write("======================Invoice for
returning land ========================\n")
20.                         invoice.write(f"Date:{date}
\t\t\t\t\t\t\t\t\tTime:{formatted_datetime}\n")
21.                         invoice.write("---------------------------------
-----------------------------------------\n")
22.                         invoice.write(f"Customer Name:
{customer_name}\n")
23.                         invoice.write(f"Phone Number: +977-
{ph_number}\n")
24.                         invoice.write(f"Kitta Number: {data[0]}\n")
25.                         total_amount = int(data[4]) * rent_dur
26.                         invoice.write(f"Total Amount to pay:
{total_amount}\n")
27.                         invoice.write(f"City/District: {data[1]}\n")
28.                         invoice.write(f"Direction: {data[2]}\n")
29.                         invoice.write(f"Date and Time of Return:
{datetime.now().strftime('%Y-%m-%d %H:%M:%S')}\n")
30.                         if return_dur > rent_dur:
31.                             total_amount = int(data[4]) * rent_dur
```

```
32.                             invoice.write(f"Total Amount without fine:
{total_amount}\n")
33.                             fine_amount = 0.1 * total_amount*(return_dur-
rent_dur)
34.                             total_amount += fine_amount
35.                             invoice.write(f"Total Amount with fine:
RS.{total_amount}\n")
36.                         elif return_dur <= rent_dur:
37.                             total_amount = int(data[4]) * rent_dur
38.                             invoice.write(f"Total Amount:
RS.{total_amount}\n")
39.                         invoice.write("---------------------------------
-------------------------------------\n\n")
40.                         # invoice.write(f"You have been fined for
{fine_amount} as your deadline exceeds")
41.                         print("\nPrinting Invoice!!!!!!!!!!!!!")
42.                         print("Please Wait!!!!!!!!!!\n")
43.                         time.sleep(5)
44.                         print("======================Invoice for
returning land ========================")
45.                         print(f"Date:{date}
\t\t\t\t\t\tTime:{formatted_datetime}\n")
46.                         print("---------------------------------------
-----------------------------------")
47.                         print(f"Customer Name: {customer_name}")
48.                         print(f"{'Phone Number:':<20} +977-{ph_number}")
49.                         print(f"Kitta Number: {data[0]}")
50.                         print(f"City/District: {data[1]}")
51.                         print(f"Direction: {data[2]}")
52.                         print(f"Date and Time of Return:
{datetime.now().strftime('%Y-%m-%d %H:%M:%S')}")
53.                         print(f"Total amount: {total_amount}")
54.                         print("---------------------------------------
-----------------------------------")
55.                         print("\nLand returned successfully!")
56.                     break
57.         else:
58.             print("Land not found or not rented.")
59.
60.         with open(file_name, 'w') as file:
61.             file.writelines(lines)
62.     except FileNotFoundError:
63.         print("File not found!")
64.     except Exception as e:
65.         print("An error occurred:", str(e))
66.
```

10.4 DisplayInfo source code

```
 1. def displayInfo(filename):
 2.     """ display all the information form  text file of land information.
it uses single arguments 'filename'
 3.     if the specified file is not founds it throws file not found
exception """
 4.     try:
 5.         gap = ' ' * 3  # inter-field gap or 3 spaces
 6.         heading = f"| {'Kitta No.':<10s} | {gap}{'City/District':<20s} |
{gap}{'Direction':<10s} | {gap}{'Area(anna)':<10s} | {gap}{'Price':<10s} |
{gap}{'Availability':<15s} |"
 7.         print("=" * 108)
 8.         print(heading)
 9.         print("-" * 108)
10.
11.         with open(filename, 'r') as file:
12.             for line in file:
13.                 data = line.strip().split(',')
14.
15.                 rec = f"| {data[0]:<10s} | {gap}{data[1]:<20s} |
{gap}{data[2]:<10s} | {gap}{data[3]:<10s} | {gap}{data[4]:<10s} |
{gap}{data[5]:<15s} |"
16.                 print(rec)
17.
18.         print("=" * 108)
19.     except FileNotFoundError as exception:
20.         print("File not found")
21.         print(" ")
22.
```

11 Plagiarism test

Originality report

COURSE NAME

FOC

STUDENT NAME

IIC BIdur

FILE NAME

FOC coursework checking rep

REPORT CREATED

May 5, 2024

Summary

Flagged passages	12	3%
Cited/quoted passages	4	1%

Web matches

geeksforgeeks.org	4	1%
studyx.ai	3	0.7%
breezetree.com	2	0.6%
w3schools.com	1	0.5%
medium.com	1	0.3%
techtarget.com	1	0.3%
collegesidekick.com	1	0.3%

bartleby.com	1	0.3%
reliasoftware.com	1	0.2%
slyautomation.com	1	0.2%

1 of 16 passages

Student passage FLAGGED

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. It was created by Guido van Rossum in…

Top web match

5 **Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility**.

Introduction to Python. Key Features: | by Techthon - Medium
https://medium.com/@techathoncert/python-is-ahigh-level-interpreted-programming-language-known-for-its-simplicity-readability-e06b903b1caf

2 of 16 passages

Student passage FLAGGED

…Geeks, 2023). Python's syntax is designed to be clear **and concise**, utilizing **indentation to denote** block structure **instead of traditional braces or keywords**

Top web match

Readability: Python emphasizes clean **and concise** code, using **indentation to denote** code blocks **instead of traditional braces or keywords**.

What is Python? Unveiling the Power of Python - Relia Software https://reliasoftware.com/blog/what-is-python

3 of 16 passages

Student passage FLAGGED

…comes along with python installer for windows. (Hlgssoftware, n.d.). **it** provides **a** user-friendly development interface **for writing, executing, and debugging python code**

Top web match

Python IDLE, short for Integrated Development and Learning Environment, is a bundled tool that comes with every Python installation. **It** serves as **a** beginner-friendly environment **for writing,**…

Python IDLE: A Comprehensive Guide for Beginners - Sly Automation https://www.slyautomation.com/blog/python-idle-a-comprehensive-guide-for-beginners/

---

4 of 16 passages

Student passage CITED

The words Algorithm means "**A set of finite rules or instructions to be followed in** calculation **or other problem-solving operations**" or "A procedure for solving a mathematical problem in…

Top web match

Definition of algorithm The word Algorithm means " **A set of finite rules or instructions to be followed in** calculations **or other problem-solving operations** " Or

What is Algorithm | Introduction to Algorithms - GeeksforGeeks https://www.geeksforgeeks.org/introduction-toalgorithms/

---

5 of 16 passages

Student passage CITED

…be followed in calculation or other problem-solving operations" or "**A procedure for solving a mathematical problem in a finite number of steps that frequently involves recursive operations**

Top web match

" **A procedure for solving a mathematical problem in a finite number of steps that frequently involves recursive operations**". Therefore Algorithm refers to a sequence of finite steps to solve a…

What is Algorithm | Introduction to Algorithms - GeeksforGeeks https://www.geeksforgeeks.org/introduction-toalgorithms/

---

6 of 16 passages

Student passage FLAGGED

Some of the common **areas where algorithms are used** are **computer science**, mathematics, operation research, **artificial intelligence, and** data sciences.

Top web match

Some of the key **areas where algorithms are used** include: **Computer Science**: Algorithms form the basis of computer programming and are used to solve problems ranging from simple sorting and searching to…

UNIT-I_058823c368e758050d98030ff08d5639.pdf - College Sidekick

https://www.collegesidekick.com/studydocs/3257586

7 of 16 passages

Student passage FLAGGED

…program planning tool to develop the logic for program. **Each step in the process is represented by a different symbol and contains a short description of the process step**

Top web match

A flow chart is a graphical or symbolic representation of a process. **Each step in the process is represented by a different symbol and contains a short description of the process step**.

What is a Flow Chart? - BreezeTree Software https://www.breezetree.com/articles/what-is-a-flow-chart

8 of 16 passages

Student passage CITED

…and contains a short description of the process step. **The flow chart** symbol is **linked together with arrows showing the process flow direction**. (Hebb, n.d.). The symbols used in flowcharts are given…

Top web match

The flow chart symbols are **linked together with arrows showing the process flow direction**. Excel Flowchart Wizard. FlowBreeze is a flowchart add-in for ...

What is a Flow Chart? - BreezeTree Software https://www.breezetree.com/articles/what-is-a-flow-chart

9 of 16 passages

Student passage FLAGGED

A data structure is a specialized format for organizing, processing, retrieving, and storing data. Data structures provide a efficient way of organizing and…

Top web match

A data structure is a specialized format for organizing, processing, retrieving and storing data. There are several basic and advanced types of data structures, all designed to arrange data to suit a…

What are Data Structures? - Definition from WhatIs.com - TechTarget https://www.techtarget.com/searchdatamanagement/definition/data-structure

10 of 16 passages

Student passage FLAGGED

…fundamental data types used to represent truth values. They **can** only **have only two possible values** which are **true or false**. Booleans are the crucial parts for conditional statements and…

Top web match

There is a special type of data that **can have** one of **only two possible values: (true or false**). What is it called? O boolean O binary O logical O veracity.

Answered: There is a special type of data that… - Bartleby.com https://www.bartleby.com/questions-andanswers/there-is-a-special-type-of-data-that-can-have-one-of-only-two-possible-values-true-or-false.-whatis/94af09c1-80a3-4cdb-8e47-c6706689bcda

---

11 of 16 passages

Student passage FLAGGED

Dictionaries in python is a data structure, used to store values in key:value format. This makes it different from lists tuples, and arrays…

Top web match

What is a dictionary in Python? **Dictionaries in Python is a data structure, used to store values in key:value format**.

Dictionaries in Python - GeeksforGeeks https://www.geeksforgeeks.org/python-dictionary/

---

12 of 16 passages

Student passage CITED

…data structure, used to store values in key:value format. **This makes it different from lists tuples, and arrays as in a dictionary each key has an associated value**

Top web match

Dictionaries in Python is a data structure, used to store values in key:value format. **This makes it different from lists, tuples, and arrays as in a dictionary each key has an associated value**.

Dictionaries in Python - GeeksforGeeks https://www.geeksforgeeks.org/python-dictionary/

13 of 16 passages

Student passage FLAGGED

…is unique and is associated with a corresponding value. **Dictionaries are mutable, allowing for the addition, modification, and deletion of key value**

Top web match

In Python, **dictionaries are mutable** objects, **allowing for the addition, modification, and**

deletion of keyvalue pairs. The del keyword is ...

Which would you use to delete an existing key | StudyX https://studyx.ai/homework/100152550-which-wouldyou-use-to-delete-an-existing-key-value-pair-from-a-dictionary-del-remove

---

14 of 16 passages

Student passage FLAGGED

Lists are one of the **built-in data types in python** which are **used to store collections of data** which **are** created using square brackets. They **are ordered, changeable and allow duplicate values**

Top web match

Lists are one of 4 **built-in data types in Python used to store collections of data**, the other 3 **are** Tuple,
Set, and Dictionary, all with different qualities and usage…List items **are ordered,**…

Python Lists - W3Schools https://www.w3schools.com/python/python_lists.asp

---

15 of 16 passages

Student passage FLAGGED

…invoice and also prints the invoice in terminal. This **invoice** includes information such **as** customer's **name**, phone number, rented **land** details, **duration of rent, and total amount**

Top web match

When land is rented, a note/**invoice** should be generated (**as** a .txt file) which must contain the kitta number of lands, **name** of the city/district, the direction of the land (Land Faced), area of **land**…

TechnoPropertyNepal is a private company that | StudyX https://studyx.ai/homework/100341195technopropertynepal-is-a-private-company-that-has-a-stock-of-different-land-on-a-contract

---

16 of 16 passages

Student passage FLAGGED

When the user chooses 3 to return rented **land, the** program asks user to input **kitta number of the land** they are **returning**, their names, **the duration of** the **rent**, the duration **of** return, **and**

Top web match

When the land is returned back, a note/invoice should be generated again which should include the name of **the** customer, **kitta number of the land**, name of the city/district, the direction of the land…
TechnoPropertyNepal is a private company that | StudyX

https://studyx.ai/homework/100341195technopropertynepal-is-a-private-company-that-has-a-stock-of-different-land-on-a-contract