CS4051NI Fundamentals of Computing

60% Individual Coursework

2023/24 Spring

Student Name: ROHAN PRASAD ADHIKARI

London Met ID: 23047505

College ID: NP01NT4A230177

Assignment Due Date: Tuesday, May 7, 2024

Assignment Submission Date: Tuesday, May 7, 2024

Word Count: 6472

Project File Links:

YouTube Link:	Keep Unlisted YouTube URL of your Project Here
Google Drive Link:	Keep Google Drive URL of your Project Here with Anyone in Organization can View Option Enabled

I confirm that I understand my coursework needs to be submitted online via MySecondTeacher under the relevant module page before the deadline in order 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.

Table of Contents

List of Figures

List of Table

1. Introduction

1.1. Introduction about the coursework

The London Metropolitan University of London's Islington College assigned this individual project for course work. The developing of a software program to manage data for an event land renting business is the major focus of this coursework. The program should have functionality for returning and renting out equipment. For a set rental fee, customers are able to rent and return several anna of land. When consumers rent or return land, this application is meant to produce invoices containing the specifics of the transactions.

I have found this coursework to be very helpful because it has given me the chance to learn how to create an inventory management application in the setting of a real-world scenario. Additionally, I got knowledge of several collection datatypes, including dictionaries, lists, and file handling concepts, which will help me in obtaining a high-level position down the road. I gained the knowledge and abilities necessary to use technology to address any real-world problem over the course of this project.

1.2. Goals and Objective

The primary objective of this coursework is to create a user application that shows the available land's Kitta No., City, Direction, Anna, and Price. When consumers need to see specific land specifics, the application pulls information from a text file containing all of that information. The read.py, write.py, operation.py, and main.py modules are the four core modules of this software.

The store's details, such as its Kitta No., City, Direction, Anna, Price, and Availability, are contained in the main.py module. Additionally, it offers a variety of options for the consumer to select from, including seeing land details, renting land, returning land, and terminating the program.

The code in the operation.py module is designed to manage multiple transactions, Aana, and Kitta No./ID validation, among other things, during the equipment rental and return process.

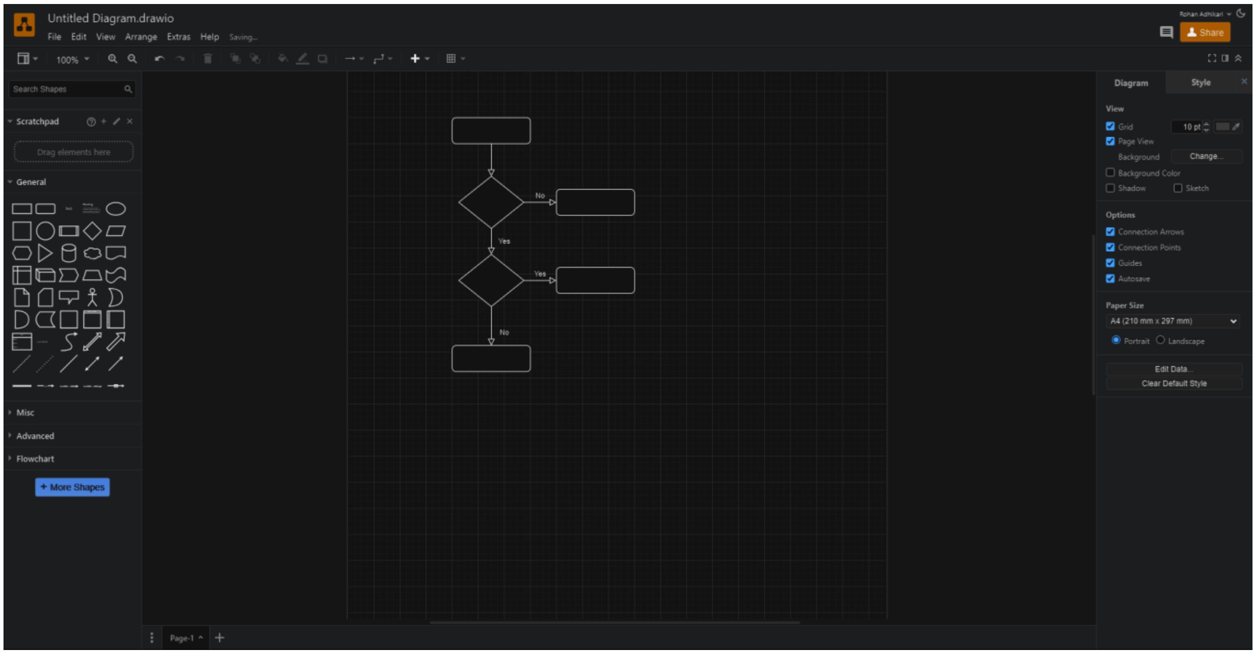
The code in the read.py module read a text file containing all the land's properties, including its Kitta No., City, Direction, Anna, and Price, and arranged it into 2-dimensional lists that the user could see on the screen.

Code for updating Anna and creating an invoice upon rental and return is contained in the last module, write.py. The transaction file also contains the unique invoice file that I created using a datetime library.

1.3. Tools used for the development of the Project

1.3.1. Draw.io

The well-known diagramming application Draw.io is used to create a wide range of diagrams, such as flowcharts, class diagrams, ER-Diagrams, and many more. It can be utilized to add additional looks to our assignments and schoolwork and is completely free. Individuals, groups, and companies use Draw.io frequently to create different types of diagrams for a variety of uses, including project management and software creation. (Alder, 2000)



View
☑ Grid 10 pt
☑ Page View
Background Change...
□ Background Color
□ Shadow □ Sketch

Options
☑ Connection Arrows
☑ Connection Points
☑ Guides
☑ Autosave

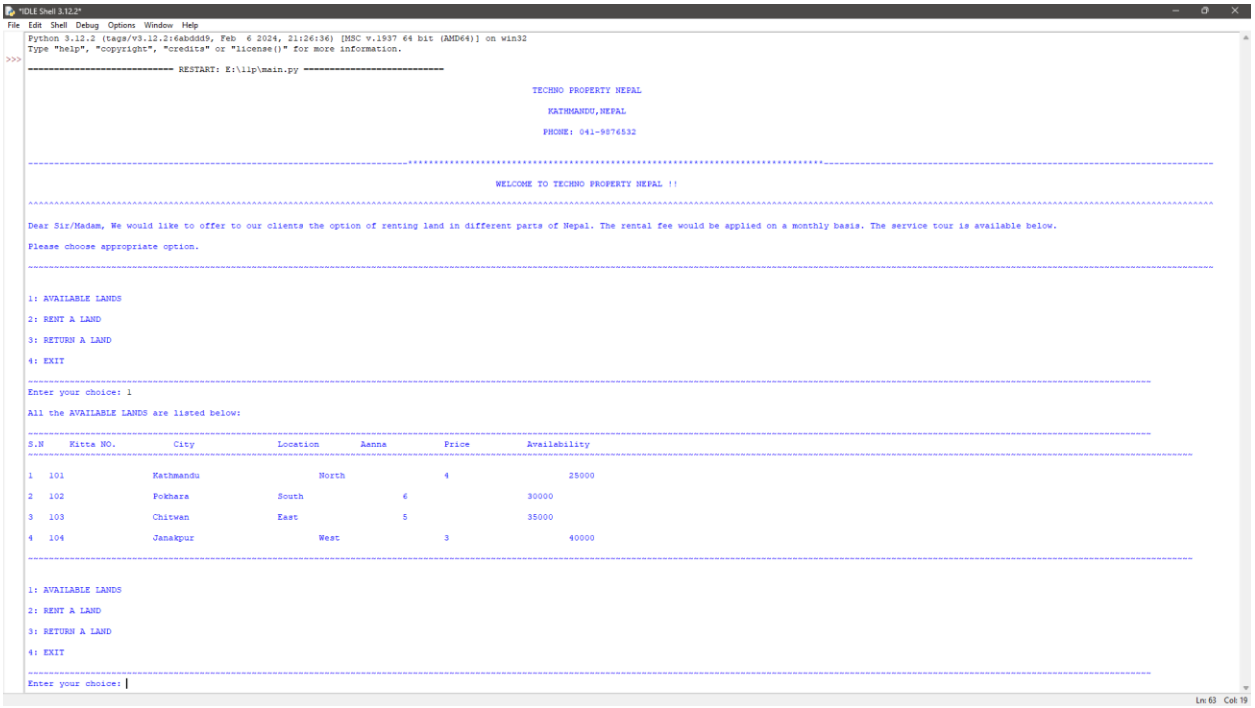
Paper Size
A4 (210 mm x 297 mm)
● Portrait ○ Landscape

Edit Data...
Clear Default Style

Figure 1 : Figure of Draw.io

1.3.2. IDLE

IDLE stands for Integrated Development and Learning Environment which is used for the python language. It includes two window types, the Shell window, and the Editor window. IDLE comes with an interactive Python shell that allows the user to execute program line by line. In a code editor, it includes features like syntax highlighting, automatic indentation, auto completion, and other features. It also includes customization features such as font size, color themes, indentation preferences to match your code. (Rossum, 1998)



```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========================== RESTART: E:\llp\main.py ==========================
                                                        TECHNO PROPERTY NEPAL

                                                          KATHMANDU,NEPAL

                                                        PHONE: 041-9876532


-----------------------------------------------------------------------*************************************************************************----------------------------------------------------------------------

                                                   WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Dear Sir/Madam, We would like to offer to our clients the option of renting land in different parts of Nepal. The rental fee would be applied on a monthly basis. The service tour is available below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1: AVAILABLE LANDS

2: RENT A LAND

3: RETURN A LAND

4: EXIT

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter your choice: 1

All the AVAILABLE LANDS are listed below:

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
S.N     Kitta NO.          City               Location        Aanna           Price          Availability
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1   101                Kathmandu                   North                 4                      25000

2   102                Pokhara             South                 6                      30000

3   103                Chitwan             East                  5                      35000

4   104                Janakpur                    West                  3                      40000

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1: AVAILABLE LANDS

2: RENT A LAND

3: RETURN A LAND

4: EXIT

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter your choice: 
```

Figure 2 : Figure of IDLE

1.3.3. Microsoft Word

Microsoft Word is a word processing program which was released in 1983 and is currently used by millions of people every day across the world. It is used to create simple and complex documents. This program can run on multiple platforms like Windows, macOS etc. We can do various things in word like add headers and footers, change font styles, page formatting and many more. We can also open and edit external PDF files as our choice. We can also insert images, video etc. MS Word can be used to create letters, business cards, bills etc. MS Word has a built-in dictionary for spell checking; misspelled words are marked with a red squiggly underline. MS Word offers text-level features such as bold, underline, italic and strike-through, and page-level features such as indentation, paragraphing, and justification. Word is compatible with many other programs, the most common being the other members of the Office suite. (Charles Simonyi, 1983)

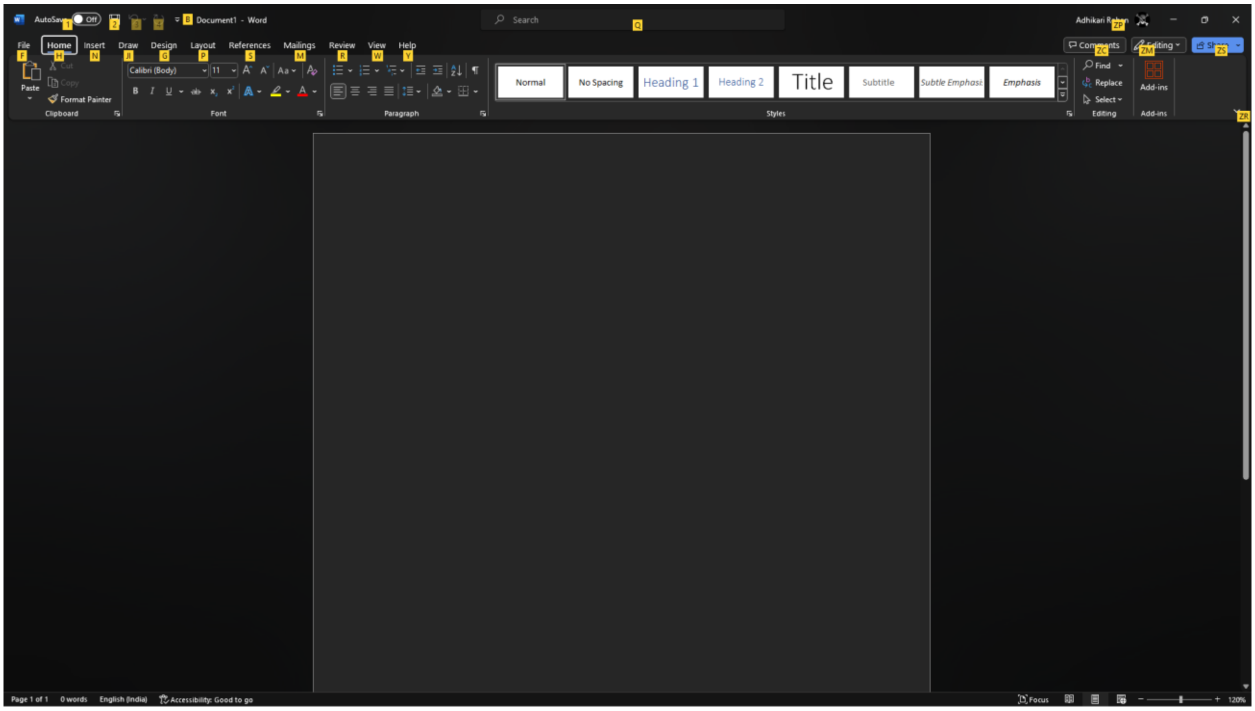


Figure 3 : Figure of MS Word

1.3.4. Notepad

With every Microsoft Windows operating system version, Notepad is a simple text editor that is pre-installed. Plain text documents are primarily created and edited with it. Files saved in Notepad have the.txt extension. Additionally, the user has the ability to write code that the command prompt can execute. Many shortcuts, such as Ctrl + N and Ctrl + O, make working in notepad easier for users. (Brodie, 1983)

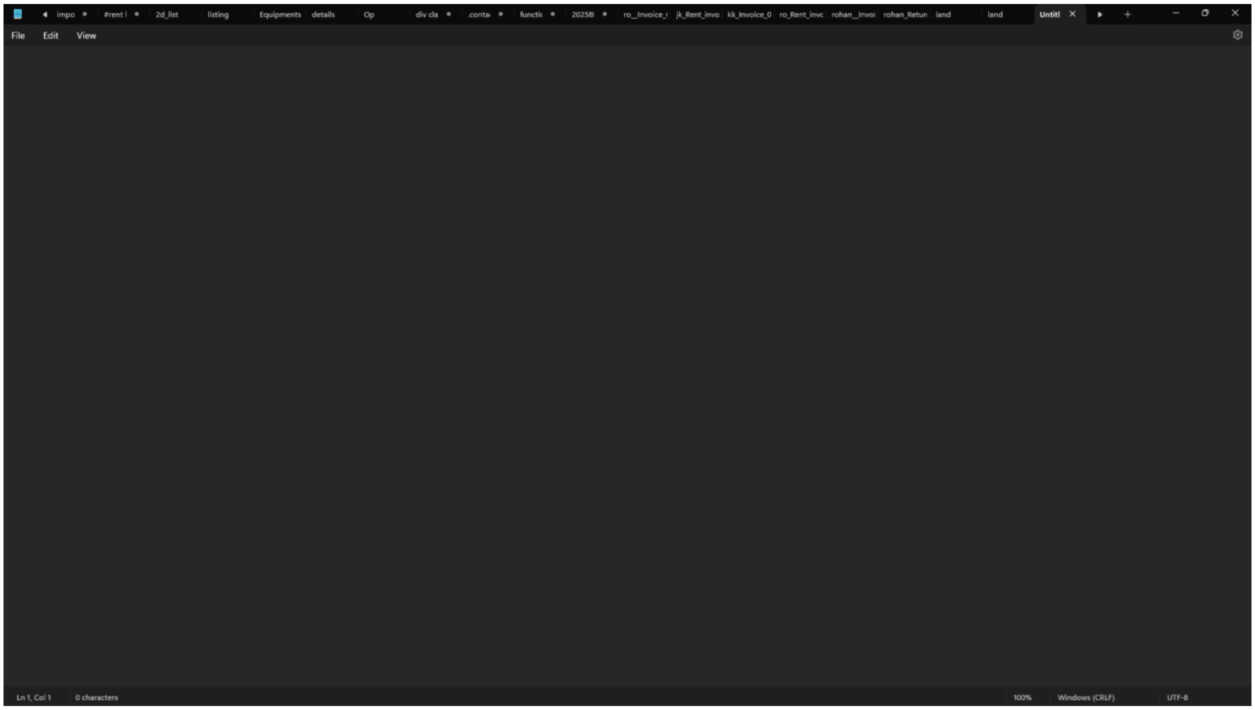


Figure 4 : Figure of Notepad

2. Algorithm

An algorithm is a well-defined collection of guidelines that help in the step-by-step solution of a problem. A program can be written using algorithms as the guide. It offers a methodical approach to resolving some difficult problems. Algorithms are useful in many fields, including mathematics, computer science, engineering, and everyday concerns.

2.1. Algorithm of the Program:

2.1.1. Algorithm for Read

Step 1: Start

Step 1.1: Define the function display_available_lands with parameter inventory

Step 1.2: Start the function display_available_lands to read land data from a file.

Step 1.3: Open the specified file in read mode.

Step 1.4: Initialize an empty list land to store land details.

Step 1.5: Read the file line by line.

Step 1.6: Append each line to the land list.

Step 1.7: Return the land list.

Step 1.8: Return inventory

2.1.2. Algorithm for the Rent Land Function:

Step 2: Check if the provided kitta number exists in the land records.

Step 2.1: If the kitta number exists:

Step 2.2: Set a flag found to true.

Step 2.3: Update the availability status of the land to "Not Available" in the land records.

Step 2.4: If the kitta number is found:

Step 2.5: Generate a rent invoice.

Step 2.6: Get the current date and time.

Step 2.7: Create an invoice with details including customer name, kitta number, rental date, and duration.

Step 2.8: Return a tuple containing the updated land records and the rent invoice.

Step 2.9: If the kitta number is not found:

Step 2.10: Return None for both records and invoice.

2.1.3. Algorithm for the Return Land Function:

Step 3: Check if the provided kitta number exists in the land records.

Step 3.1: If the kitta number exists:

Step 3.2: Set a flag found to true.

Step 3.3: Update the availability status of the land to "Available" in the land records.

Step 3.4: If the kitta number is found:

Step 3.5: Generate a return invoice.

Step 3.6Get the current date and time.

Step 3.7: Extract the rented month and customer name from the record.

Step 3.8: Generate an invoice for the return including customer name, kitta number, rental date, returned month, and price per month.

Step 3.9: Return a tuple containing the updated land records and the return invoice.

Step 3.10: If the kitta number is not found:

Step 3.11: Return None for both records and invoice.

2.1.4. Algorithm for the Write:

Step 4: Get the current date and time.

Step 4.1: Initialize an empty string invoice.

Step 4.2: Depending on the invoice_type:

If the invoice_type is "rent":

Step 4.3: Create a rent invoice.

Step 4.4: Include customer name, kitta number, transaction date, rental date, and duration in the invoice.

If the invoice_type is "return":

Step 4.5: Create a return invoice.

Step 4.6: Include customer name, kitta number, transaction date, return date, rented month, returned month, price per month, total price, delayed months, fine price, and amount with fine in the invoice.

Step 4.7: Return the generated invoice.

2.1.5. Algorithm for the Main function:

```
Step 5: Main Function:
Step 5.1: Initialize the company name and file path.
Step 5.2: Read land records from the file.
Step 5.3: While True:

Step 5.4: Display the menu.
Step 5.5: Prompt the user for their choice.
Step 5.6: If the choice is '1':
Step 5.6.1: Display company information.
Step 5.7: If the choice is '2':
Step 5.7.1: Display all lands.
Step 5.8: If the choice is '3':
Step 5.8.1: Display available lands.
Step 5.9: If the choice is '4':
Step 5.9.1: Display unavailable lands.
Step 5.10: If the choice is '5':
Step 5.10.1: Display lands for rent.
Step 5.10.2: Prompt the user to enter the kitta number of the land to rent.
Step 5.10.3: If the land is found:
Step 5.10.3.1: Prompt the user for their name and rental duration.
Step 5.10.3.2: Rent the land and generate an invoice.
Step 5.10.3.3: Print the success message and invoice.
Step 5.10.3.4: Write the invoice to the file.
Step 5.11: If the land is not found, print an error message.
Step 5.12: If the choice is '6':
Step 5.12.1: Prompt the user to enter the kitta number of the land to return.
Step 5.12.2: If the land is found:
Step 5.12.2.1: Prompt the user for their name, rented month, returned month, and price per month.
Step 5.12.2.2: Return the land and generate an invoice.
```

Step 5.12.2.3: Print the success message and invoice.

Step 5.12.2.4: Write the invoice to the file.

Step 6: If the land is not found, print an error message.

Step 6.1: If the choice is '7':

Step 6.1.1: Print the exit message.

Step 6.1.2: Write the updated land records to the file.

Step 6.1.3: Break the loop.

Step 7: If the choice is invalid, print an error message.

Step 8: End.

3. Flowchart

A flowchart is a graphical representation of an algorithm's process that highlights the information flow using various forms, symbols, and arrows. They offer a visual guide for understanding and analysing a problem's justification.

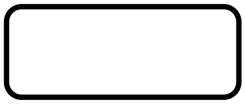
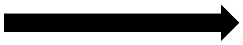
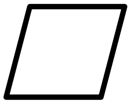
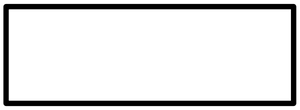
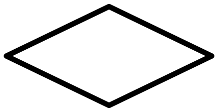
Symbol	Name	Function
	Start/End	It represents a start or end point.
	Arrows	It connects the representative shapes.
	Input/Output	It represents input or output.
	Process	It represents a process.
	Decision	It indicates a decision.

Table 1: Table of Flowchart Shapes

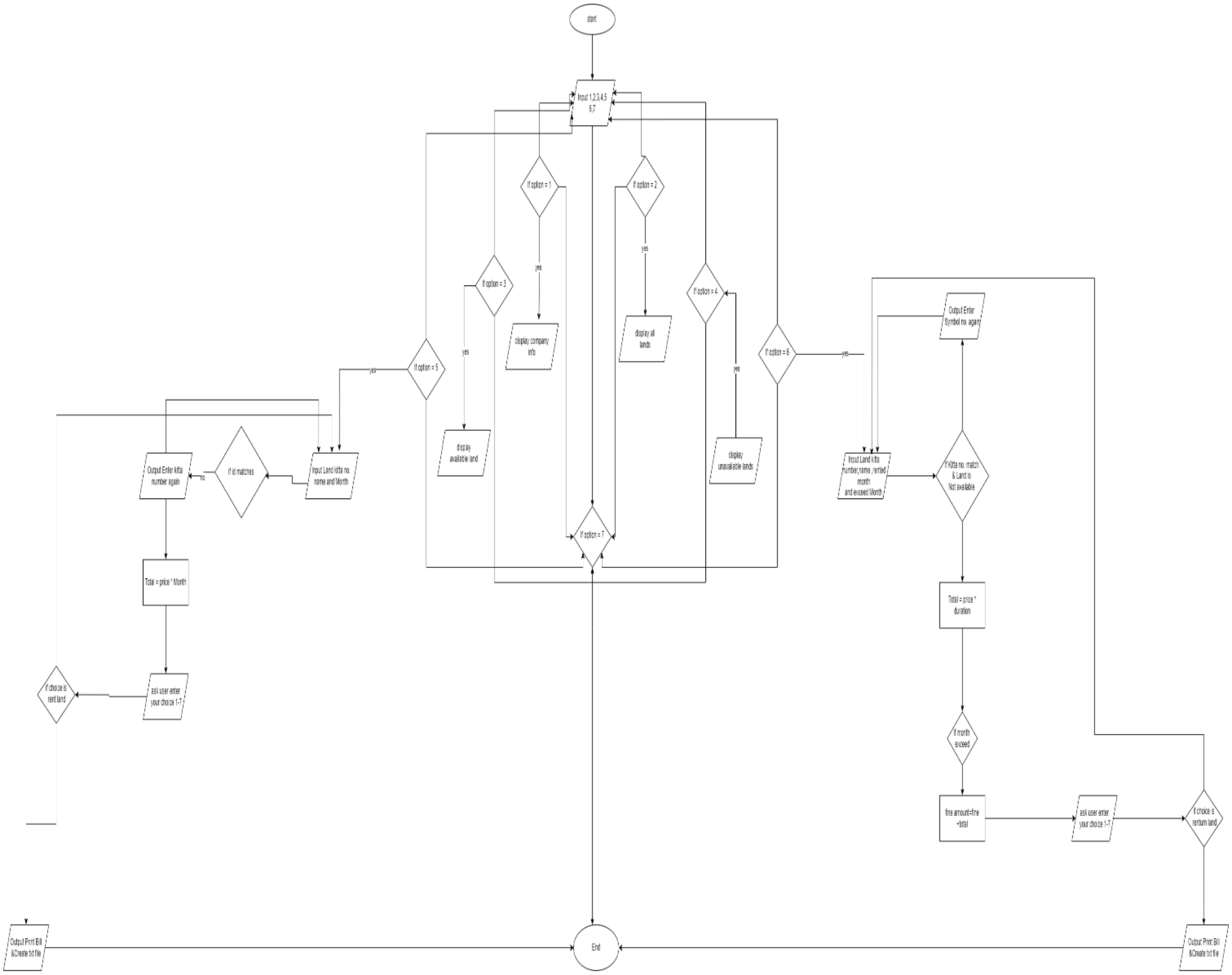


Figure 5 : Figure of Flowchart of Program

4. Pseudocode

Pseudocode is a form of programming that uses natural language and syntax similar to that of a computer language. Usually written in simple language, it makes use of suitable indentation and well-structured statements to show logic and control flow. When a computer program is written in Integrated Workplace for Development (IDE).

4.1. Pseudocode of main.py:

```
import read
import operation

        create a function name save_lands_data (filename, inventory)
                with open (filename,w) as file
                        for all kitta no and land data in inventory
                                file.write (land details)


        reads the land.txt file
        output blank line
        output ("\t\t\t\t\t\t\t\t\t\t\t TECHNO PROPERTY NEPAL
        \n\n\t\t\t\t\t\t\t\t\t\t\t KATHMANDU, NEPAL \n\n\t\t\t\t\t\t\t\t\t\t\t PHONE:
        041-9876532 \n”)
        output blank line
        output("------------------------------------------------------------------------
*******************************************************************************--------------
------------------------------------------------------------")
        output blank line
        output ("\t\t\t\t\t\t\t\t\t\t  WELCOME TO TECHNO PROPERTY NEPAL !!
\n")

        output("^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^")
        output blank line
        output ("Greetings, Sir or Madam We would want to provide our consumers
        with the opportunity to rent land in various regions of Nepal. Each month,
        the rental fee would be charged. You can access the service tour below.
         \n")
        output ("Please choose appropriate option.")
        output blank line

        output("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

```
        ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
        ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
        ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
        ~~~~~~~~~~")
        output blank line
        while true
            output (blank)
            output ("1. Display All Lands")
            output ("2. Rent Land")
            output ("3. Return Land")
            output ("4. Exit")

            choice equals to operation.get_integer_input("Enter your choice")

            if choice equals to one
                displays display_available_lands(inventory)

            elif choice equals to two
                display _available _lands(inventory)
                operations.rent_land(inventory)
                output ("Thanks for visiting us")

            elif choice equals to three
                operations.return_land(inventory)

            elif choice equals to four
                output ("Exiting...")
                output ("Thanks for Visting Us")
                return
            else
                output("Invalid Choice")
        save_land_data('land_inventory.txt',inventory)

if __name__equals to '__main__':
   main()
```

4.2. Pseudocode of operation.py:

```
Created generate_invoice_terminal(invoice_type, customer_name, kitta_no, duration equals to  None, fine equals to  None, rental_date equals to  None, return_date equals to None, price_per_month equals to  None)
    Set now equals to current date and time in "YYYY-MM-DD HH:MM:SS" format
    Set invoice equals to   "{invoice_type.capitalize()} Invoice\n"
    Append "-------------------------------------------------\n" to invoice
    Append "\t\t\t\t\t\t\t\t\t\t\t TECHNO PROPERTY NEPAL \n\n\t\t\t\t\t\t\t\t\t\t\t KATHMANDU,NEPAL \n\n\t\t\t\t\t\t\t\t\t\t\t   PHONE: 041-9876532 \n" to invoice
    Append "Customer Name: {customer_name}\n" to invoice
    Append "Kitta Number: {kitta_no}\n" to invoice
    Append "Transaction Date: {now}\n" to invoice

    If invoice_type equals to "rent"
        Append "Rental Date: {rental_date}\n" to invoice
        Append "Duration: {duration} months\n" to invoice
        Append "Price: {duration * price_per_month}\n" to invoice
    Else If invoice_type equals "return"
        Append "Return Date: {return_date}\n" to invoice
        Append "Duration: {duration} months\n" to invoice
        Append "Fine: {fine}\n" to invoice
        Append "Total Price: {duration * price_per_month + fine}\n" to invoice

    Output invoice

Created Function rent_land(inventory)
    Set now equals to   current date and time
    Output ( "---------------------------------------------------------------------\n")
    Set kitta_no equals to  get user input("Enter Kitta Number to rent: ")

    If kitta_no exists in inventory and inventory[kitta_no]['status'] equals 'Available'
        Set customer_name equals to  get user input("Enter Customer Name: ")
        Set duration equals to call get_integer_input with "Enter rental duration (in months): "


        Set inventory[kitta_no]['status'] equals to 'Not Available'

        If customer_name not in write.customer_transactions
            Set write.customer_transactions[customer_name] to empty list

        Append {
            'transaction_type': 'Rent',
            'land_info': {'kitta_no': kitta_no, inventory[kitta_no]},
            'duration': duration,
            'rented_date': now in 'YYYY-MM-DD HH:MM:SS' format
        } to write.customer_transactions[customer_name]

        Call write.generate_customer_invoice with customer_name
        Call generate_invoice_terminal with parameters "rent", customer_name, kitta_no, duration, None, now in 'YYYY-MM-DD HH:MM:SS' format, None, inventory[kitta_no]['price']
        Output "-----------------------------------------------------------------\n"
        Output "Land rented successfully!"
    Else
        Output “Land Unavailable or Please enter the correct Kitta No."

    Return inventory

Created Function return_land(inventory)
    Output  "------------------------------------------------------------------------\n"
    Set kitta_no equals to get user input("Enter Kitta Number to return: ")

    If kitta_no exists in inventory and inventory[kitta_no]['status'] equals 'Not Available'
        Set customer_name equals to get user input("Enter Customer Name: ")
        Set return_rented_duration equals to call get_integer_input with "Enter original rental duration (in months): "

        Set now equals to current date and time
        Set return_date equals to now in 'YYYY-MM-DD HH:MM:SS' format

        Try
          Set actual_rented_duration equals to write.customer_transactions[customer_name][0]['duration']
Catch
           Output "This customer hasn't done any transactions before."
           Return call return_land with inventory

        Set months_late equals to max(0, return_rented_duration - actual_rented_duration)
        Set price_per_month equals to inventory[kitta_no]['price']
        Set fine equals to round(0.1 * months_late * price_per_month)

        Call write.generate_return_invoice with customer_name, kitta_no, inventory[kitta_no], return_rented_duration, fine
        Call generate_invoice_terminal with parameters "return", customer_name, kitta_no, return_rented_duration, fine, None, return_date, price_per_month

        Set inventory[kitta_no]['status'] equals to 'Available'
        Output "-------------------------------------------------------\n"
        Output "Land returned successfully!"
    Else
        Output "Land not currently rented or invalid Kitta Number."
```

```
Created Function get_integer_input(prompt, error_message equals to "Invalid input. Please enter a number.")
    While True
        Try
            Return integer of user input(prompt)
        Catch
            Output  error_message
```

4.3. Pseudocode of write.py:

```
Create generate_customer_invoice(customer_name)
   If customer_name in customer_transactions
      Open file '{customer_name} _Rent_invoice.txt' as f
         Write ("TechnoPropertyNepal")
         Write ("--------------------------------------------------")
         total_amount = 0
         For each transaction in customer_transactions[customer_name]:
            Write ('Customer Name: {customer_name}')
            Write ('Transaction Type: {transaction["transaction_type"]}')
            Write ('Kitta Number: {transaction["land_info"]["kitta_no"]}')
            Write ('City/District: {transaction["land_info"]["city"]}')
            Write ('Direction: {transaction["land_info"]["direction"]}')
            Write ('Area of Land: {transaction["land_info"]["area"]} anna')
            Write ('Date and Time: {transaction["rented_date"]}')
            Write ('Duration of Rent: {transaction["duration"]} months')
            total_amount euqals to transaction["duration"] * transaction["land_info"]["price"]
         Write ('Total Amount: {total_amount}')
   Else:
      Output (“No transactions found for this customer.")

Created generate_return_invoice(customer_name, kitta_no, land_info, duration, fine)
   now = datetime.now()
   If customer_name in customer_transactions:
      Open file '{customer_name}_Return_invoice.txt' as f:
         Write ("TechnoPropertyNepal")
         Write ("--------------------------------------------------")
         Write ('Customer Name: {customer_name}')
         Write ('Transaction Type: Return')
         Write ('Kitta Number: {kitta_no}')
         Write ('City/District: {land_info["city"]}')
         Write ('Direction: {land_info["direction"]}')
         Write ('Area of Land: {land_info["area"]} anna')
         Write ('Date and Time of Return: {now}')
         Write ('Duration of Rent: {duration} months')
         total_amount = duration * land_info['price'] + fine
         Write ('Total Amount: {total_amount}')
         Write ('Fine: {fine}')
   Else:
      Output ("No transactions found for this customer.")
```

4.4. Pseudocode of read.py:

```
Created display_available_lands(inventory):
    output ("Available Lands:")
    For each kitta_no, land_data in inventory:
        If land_data['status'] is 'available':
            output "Kitta Number: {kitta_no}, City: {land_data['city']}, Area: {land_data['area']} anna, Status: {land_data['status']}"

created load_land_data(filename):
    inventory equals to empty dictionary
    Open file filename as file:
        For each line in file:
            data equals to split line by ','
            inventory[data[0]] equals to dictionary with keys:
                'city' equals to data[1]
                'direction' equals to data[2]
                'area' equals to convert data[3] to integer
                'price' equals to convert data[4] to integer
                'status' equals to strip data[5]
    Return inventory
```

5. Data Structure

The fundamental unit of storage or container used to organize and process data is called a data structure. Python offers a wide variety of data structures, including dictionaries, sets, tuples, lists, and many more. Programming requires the use of data structures in order to manage massive volumes of data efficiently and perform different algorithms, such bubble sort, searching, string, and other algorithms.

The data types that are used in this project are listed below:

- Integer
- String
- List
- Boolean

5.1. Integer

The most used data type in programming that can only store whole integers is the integer. The size of an integer data type varies based on the programming language; for example, Java uses 4 bytes, while other languages use 2 bytes or 1 byte. In some ranges, this datatype can also hold or store negative values. The purpose of selecting an integer data type for my coursework is to compute the total cost following the rental of lands. Additionally, I used integer data type to accept user input for choosing a choice in the main program.

5.2. String

In computer languages, an order of characters is represented by a string, a non-primitive datatype. Python allows us to generate strings by associated with a character sequence in a single, double, or triple quotation. Any type of data, including text, decimals, and integer values, can be stored in this datatype. For my coursework, I generated an invoice for rent and return transactions using string datatype and type casting while writing in a file.

5.3. List

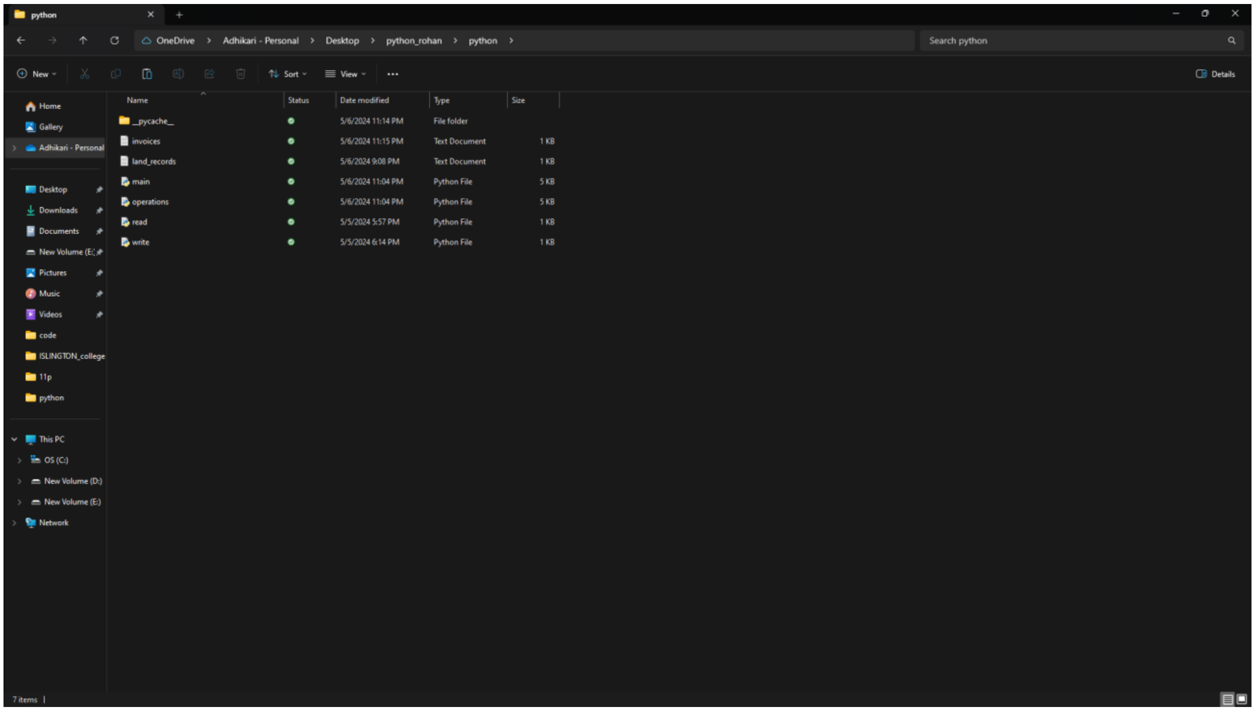
One of the most popular data structures in Python programming is the list, which allows you to declare a single variable and store numerous values. Python lists support duplicate values, are mutable, and are ordered. Although we can store values of several data types in a single variable, it functions somewhat like Java's ArrayList. It is represented by the symbol "[], “and commas "," are used to divide values or items inside the large bracket. We can dynamically expand or contract lists. In my coursework, after reading the file, a list data structure is utilized to store anna of land that is offered at the store in 2-D Array form.

5.4. Boolean

Another basic datatype in programming that represents a binary value that can be either true or false is boolean. Conditions, logical values, and the output of logical operations are all represented by Booleans. I utilized boolean data type in my project to show the user options until they wanted to exit.

6. Program

The program is mainly designed for rental stores that let customers check out and return various Lands. Multiple modules, including read.py, write.py, operation.py, and main.py, form this application. These modules all have various purposes and are connected to one another. A feature of the read.py module reads a text file and displays every piece of land’s information as a 2-dimensional list. The renting, returning, and lands display processes are handled by a number of functions in the operation.py module. Enabling the user to choose an option is the responsibility of the main.py module. Finally, invoices for both rental and return transactions are generated by the write.py module.

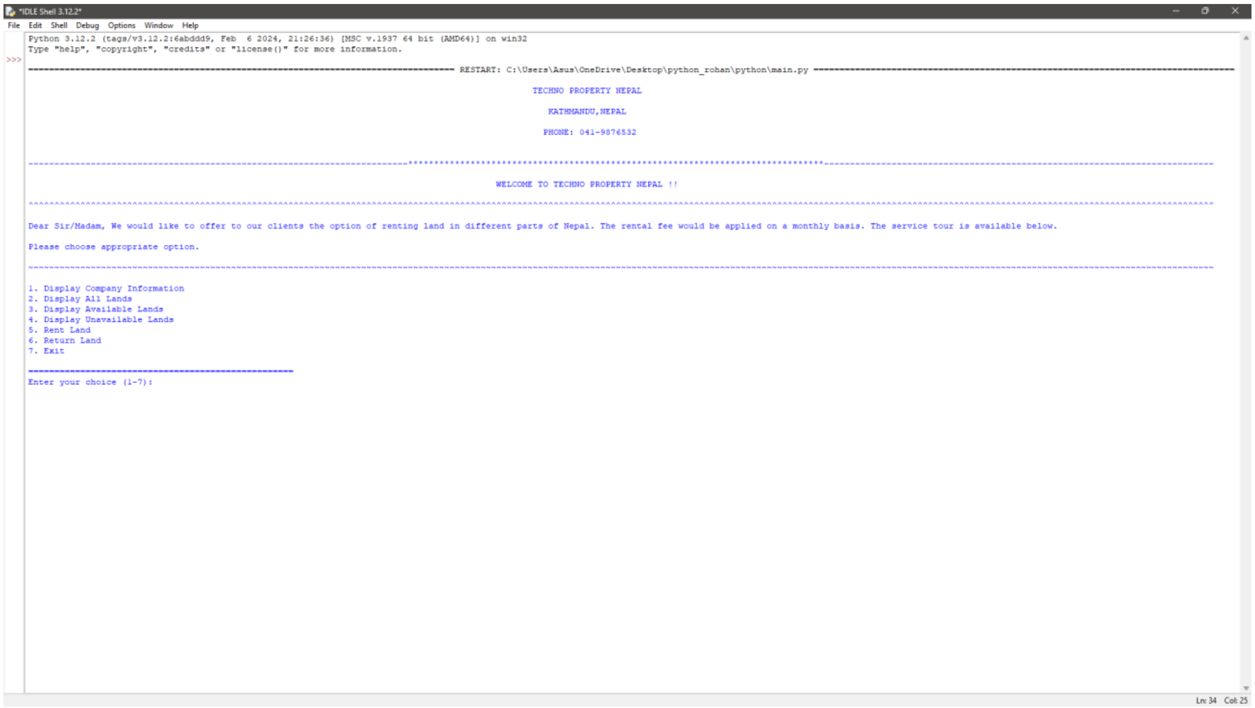


Name	Status	Date modified	Type	Size
__pycache__	✓	5/6/2024 11:14 PM	File folder	
invoices	✓	5/6/2024 11:15 PM	Text Document	1 KB
land_records	✓	5/6/2024 9:08 PM	Text Document	1 KB
main	✓	5/6/2024 11:04 PM	Python File	5 KB
operations	✓	5/6/2024 11:04 PM	Python File	5 KB
read	✓	5/5/2024 5:57 PM	Python File	1 KB
write	✓	5/5/2024 6:14 PM	Python File	1 KB

Figure 6 : Figure of modules of program

- **First interface when the user run the main program.**

  A number of options are displayed to the user in the primary application interface when it is running. The program's title, address, contact information (phone and email), and a welcome message are displayed at the top. After this, the user can make informed decisions by selecting from the several options in the centre. All of the land that is currently accessible, along with all of its details, can be seen on the screen by selecting the first option. To proceed with the rental process after that, choose the second option. Moreover, you can return the land using the third option. You can end the program by using the final option. An error warning will appear on the screen and the user will be given another chance to choose the correct option if they input an alphabetic value. We'll look at further tasks that need to be completed when choosing the option to rent and return.

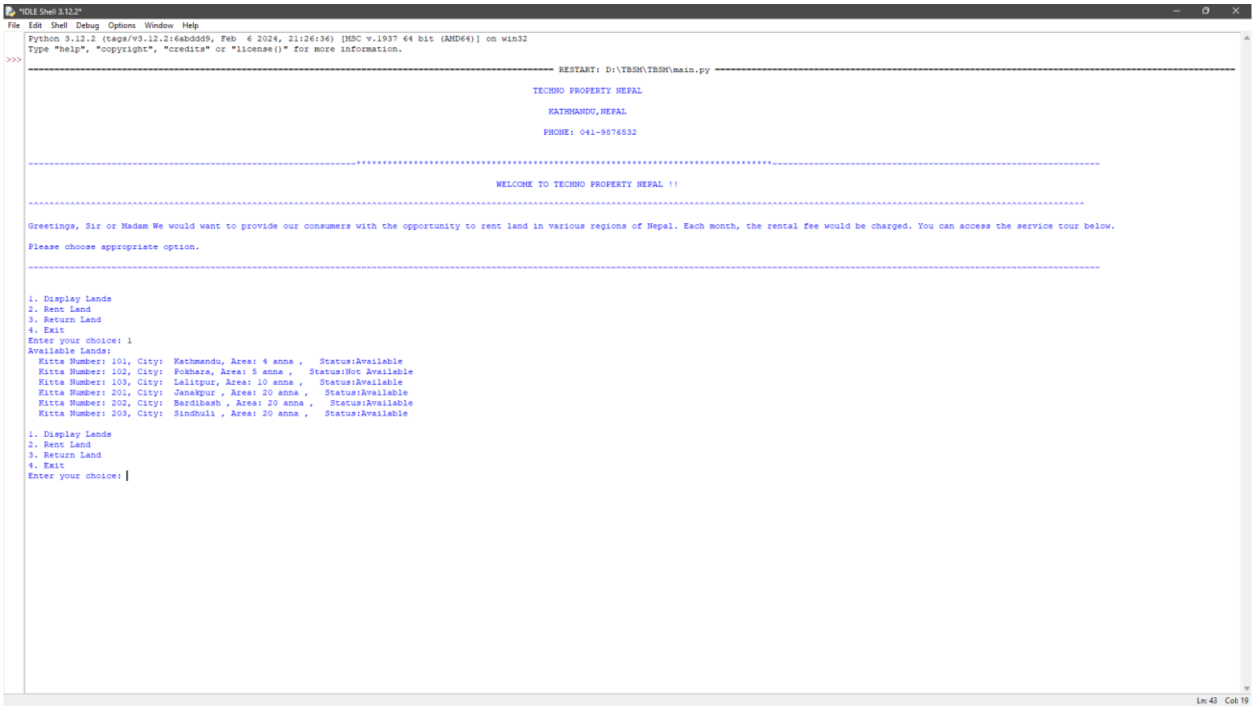


```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
============================ RESTART: C:\Users\Asus\OneDrive\Desktop\python_rohan\python\main.py ============================
                                        TECHNO PROPERTY NEPAL

                                          KATHMANDU,NEPAL

                                        PHONE: 041-9876532


-----------------------------*******************************************************-----------------------------

                                    WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Dear Sir/Madam, We would like to offer to our clients the option of renting land in different parts of Nepal. The rental fee would be applied on a monthly basis. The service tour is available below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1. Display Company Information
2. Display All Lands
3. Display Available Lands
4. Display Unavailable Lands
5. Rent Land
6. Return Land
7. Exit

===================================================
Enter your choice (1-7):
```

Figure 7 : Figure of Interface of program

- **When the user enters 1 as input.**

  All of the land that is available in the file is shown on the screen with details when the user chooses option 1. The Kitta No., City, Direction, Anna, and Price of the accessible land are among the details. All of the choices reappeared to carry out the program after viewing all of the lands.

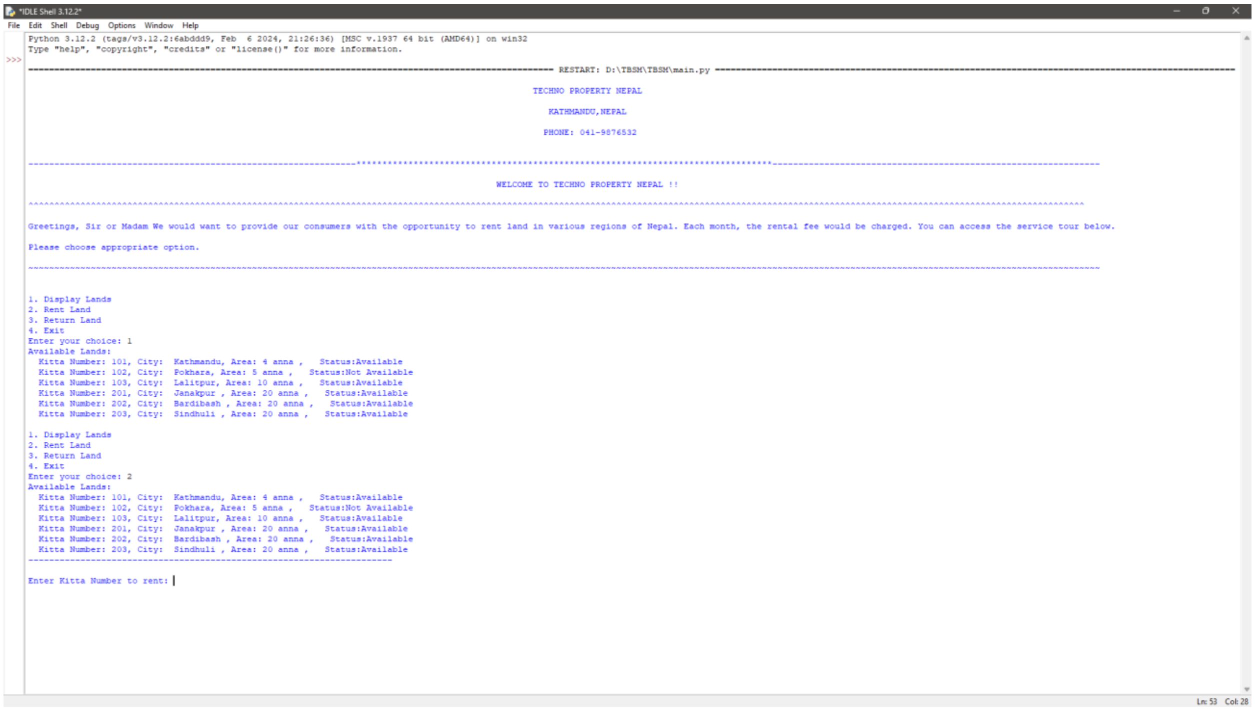


```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
================================ RESTART: D:\TBSM\TBSM\main.py ================================

                                        TECHNO PROPERTY NEPAL

                                          KATHMANDU,NEPAL

                                        PHONE: 041-9876532

--------------------------------------------*********************************************-----------------------------------------

                                  WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 1
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 
```

Figure 8 : Figure of user enters 1 as input

- **When the user selects option 2:**

The information about every anna of land that is available in the file is shown on the screen according to the land that the customer wants to rent. The information includes the Kitta No., City, Direction, Anna, and Price of the available land. For rental purposes, the program asks for the Kitta No. or ID. The code shows the information about the input Kitta No./ID's land after obtaining a valid Kitta No./ID. The user then gets asked by the code to provide a valid aana of land and days. The user is asked if they would like to continue renting after the first purchase is completed. The same procedure is repeated if the user wants to keep renting. In order to create an invoice and request the user's information, including their complete name, phone number, and address, the user must indicate that they want to stop renting. Following the acquisition of personal data, an invoice is created and shown both in the file and on the screen.

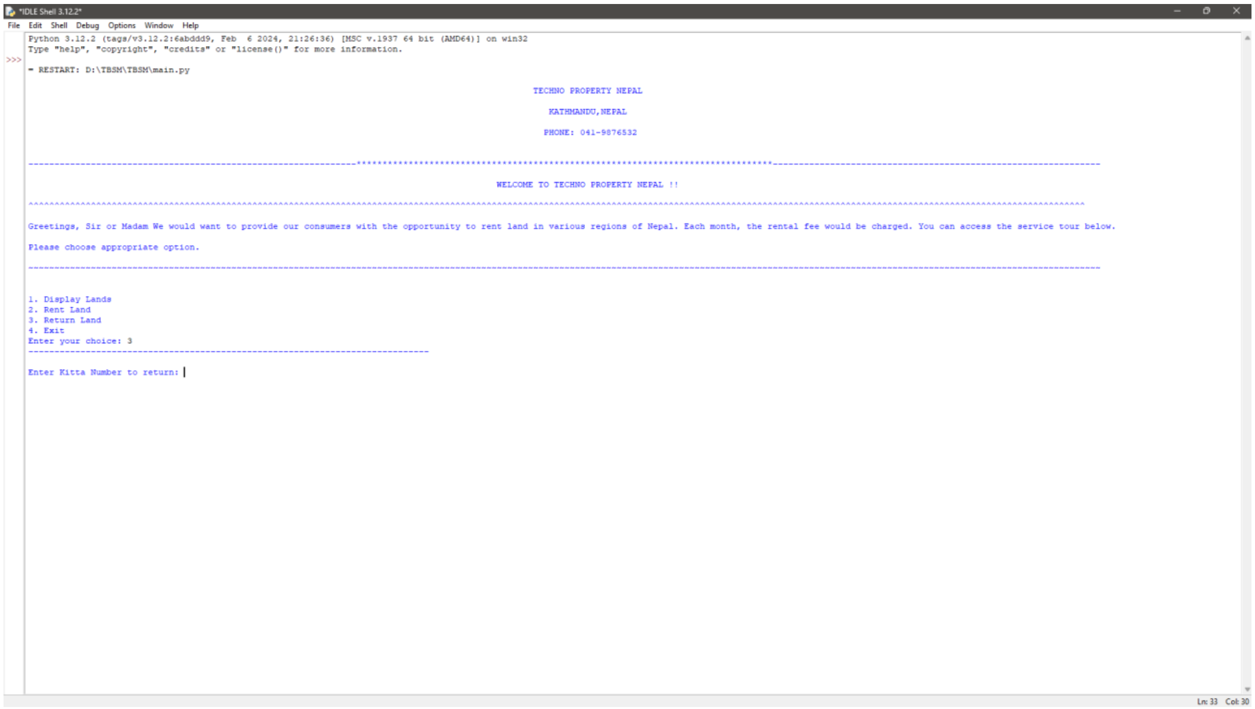


```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============================================ RESTART: D:\TBSM\TBSM\main.py ============================================
                                                          TECHNO PROPERTY NEPAL

                                                            KATHMANDU,NEPAL

                                                          PHONE: 041-9876532


------------------------------------------------------------***************************************************************-------------------------------------------------------------

                                                    WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 1
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 2
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available
----------------------------------------------------------------------

Enter Kitta Number to rent: 
```

Figure 9 : Figure of when user selects option 2

- **When the user enters 3:**

As input the details of every piece of land that is in the file are shown on the screen when the user chooses option 3, showing which land the customer wants to return. The available land's Kitta No., City, Direction, Anna, and Price are among the details. The program asks for the return of the Kitta No. Subsequently, the code asks the user to input a legitimate number of Anna, the duration of the rental, and the duration of the return visit. The user is asked to proceed with their next transaction or not after completing the first one. The same procedure is done if the user wants to keep coming back.

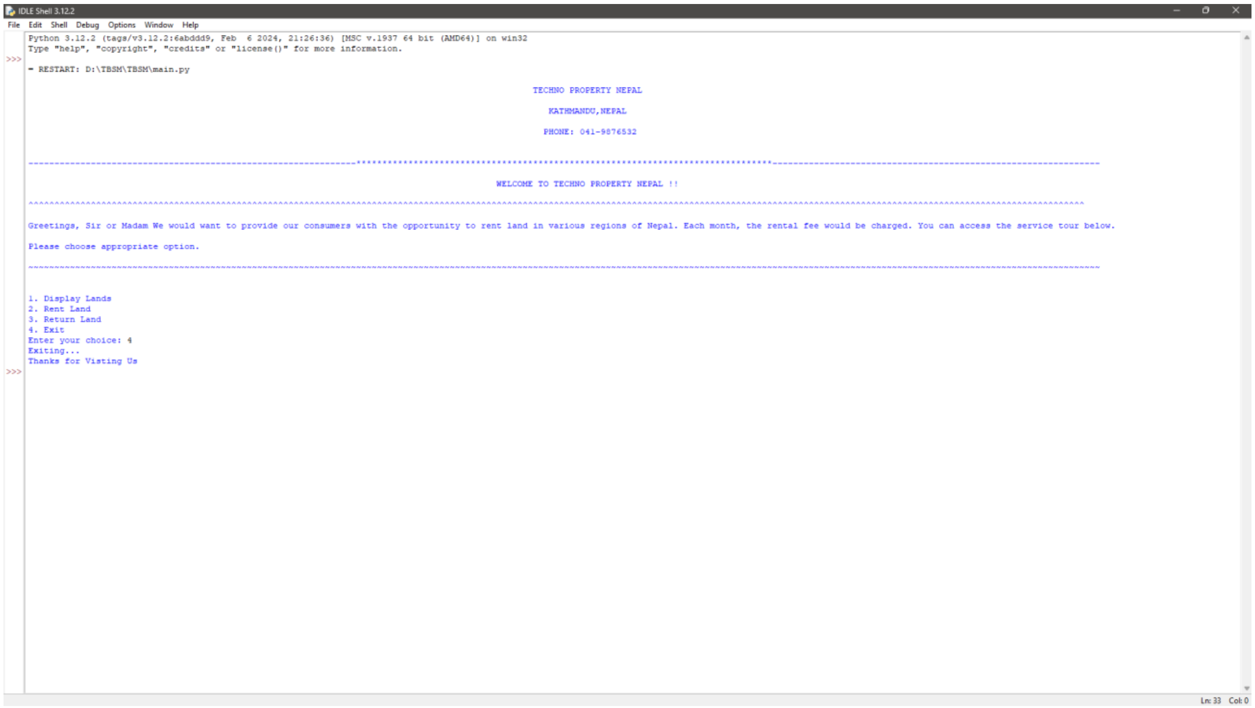


```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: D:\TBSM\TBSM\main.py

                                                                    TECHNO PROPERTY NEPAL

                                                                      KATHMANDU,NEPAL

                                                                    PHONE: 041-9876532


--------------------------------------------------------------*******************************************************************************-------------------------------------------------------------

                                                              WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 3
-----------------------------------------------------------------------------

Enter Kitta Number to return: 
```

Figure 10 : Figure of when user selects option 3

- **When the user enters 4**

  As input the program displays a message to the user on the screen and ends when the user chooses option 4.



```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: D:\TBSM\TBSM\main.py

                                                  TECHNO PROPERTY NEPAL

                                                    KATHMANDU,NEPAL

                                                  PHONE: 041-9876532

---------------------------------------------------------------*******************************************************************************----------------------------------------------------------------

                                                WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 4
Exiting...
Thanks for Visting Us
>>>
```

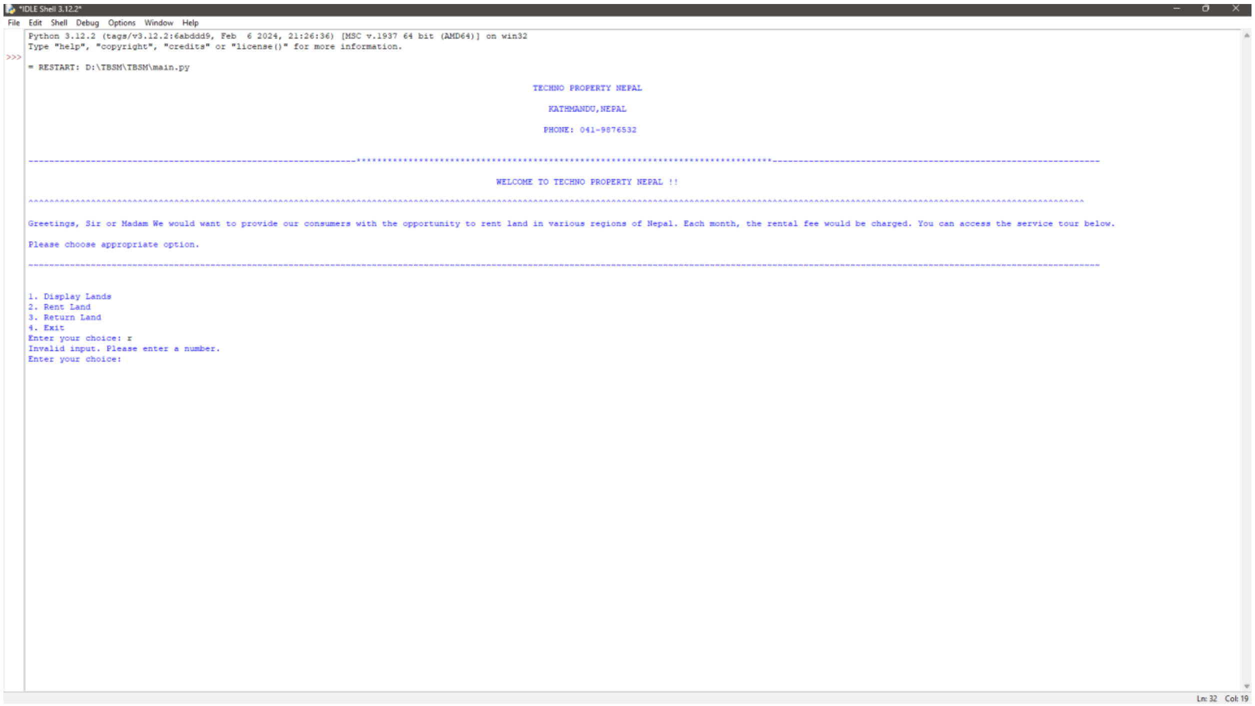
Figure 11 : Figure of when user enters 4

7. Testing

7.1. Test 1 – Implementation of try and except

Objective	Implementation of try and except
Action	Program was compiled • The main program was run. • Alphabetic value was entered as input.
Expected result	The user should get an error message and be given another chance to choose the option.
Actual result	The expected and actual results were same. The user was asked to select the option again when an error warning showed up on the screen.
Conclusion	Test successful

Table 2 : Implementation of try and except



```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: D:\TBSM\TBSM\main.py

                                                    TECHNO PROPERTY NEPAL

                                                      KATHMANDU,NEPAL

                                                    PHONE: 041-9876532

-----------------------------------------------------------------*****************************************************************-------------------------------------------------------------------

                                                WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: r
Invalid input. Please enter a number.
Enter your choice:
```

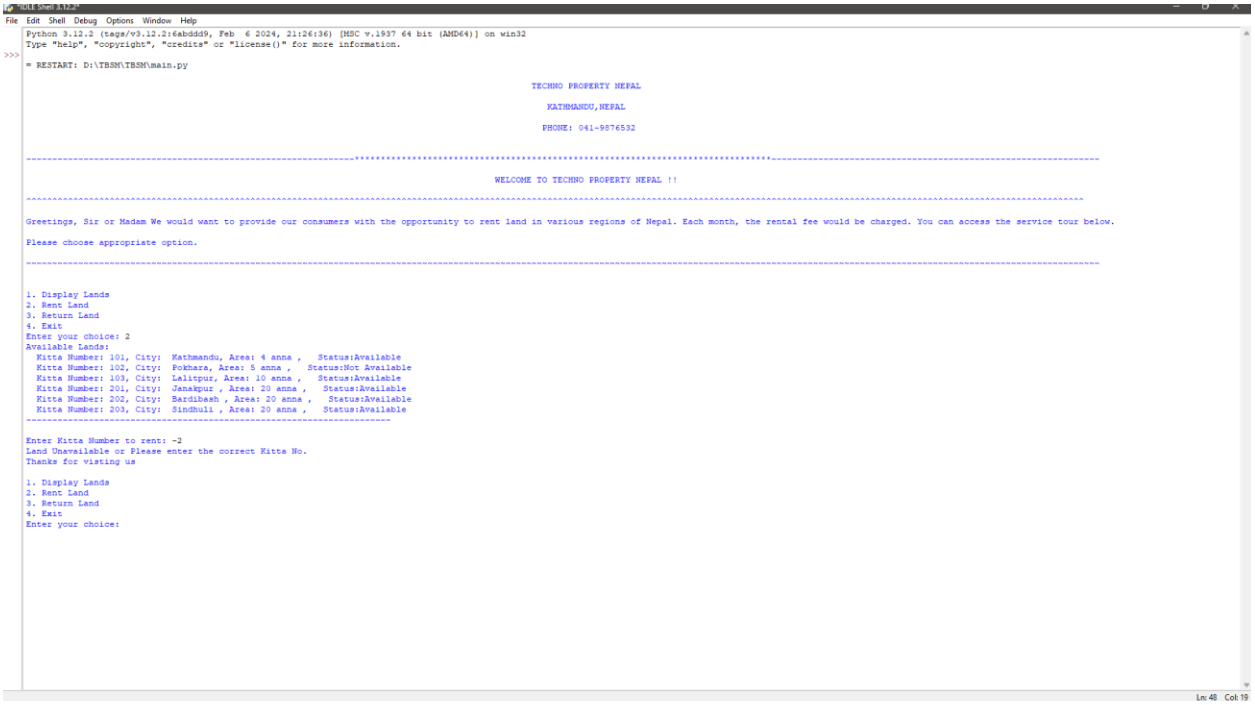
Figure 12 : Figure of Implementation of try and except

7.2. Test 2

7.2.1. Testing - Negative value and non-existed value as input while renting lands

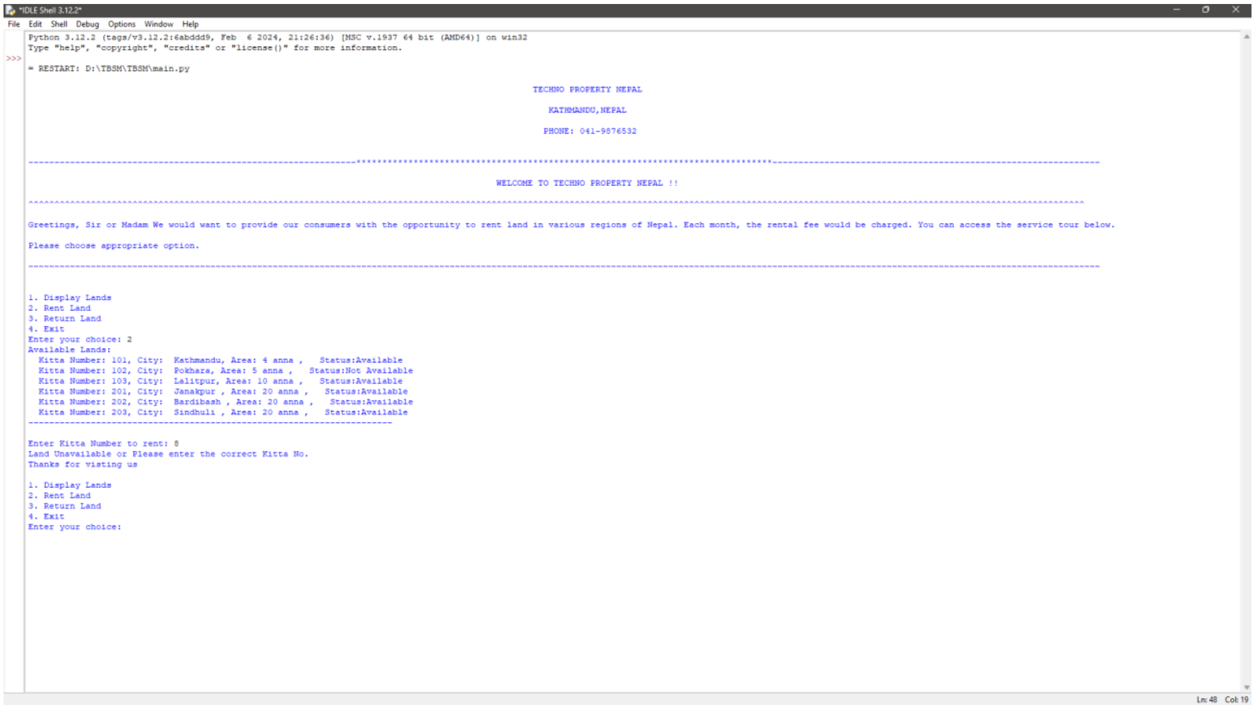
Objective	Negative value and non-existed value as input while renting lands
Action	The main program was executed. • When renting land, a negative value of -2 was first entered. • Later, a non-existing value of 8 was entered.
Expected result	In both cases, the user should to see an error message and be given chance to choose the right response.
Actual result	The expected and actual results were same. The user was able to choose the correct option after seeing an error warning on the screen.
Conclusion	Test successful

Table 3 : Negative value and non-existed value as input while renting lands



```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: D:\TBSM\TBSM\main.py

                                        TECHNO PROPERTY NEPAL

                                          KATHMANDU,NEPAL

                                         PHONE: 041-9876532

----------------------------------------------------------------***************************************************************************--------------------------------------------------------------

                                   WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 2
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available
----------------------------------------------------------------------

Enter Kitta Number to rent: -2
Land Unavailable or Please enter the correct Kitta No.
Thanks for visting us

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice:
```

Figure 13 : Negative value as input while renting lands



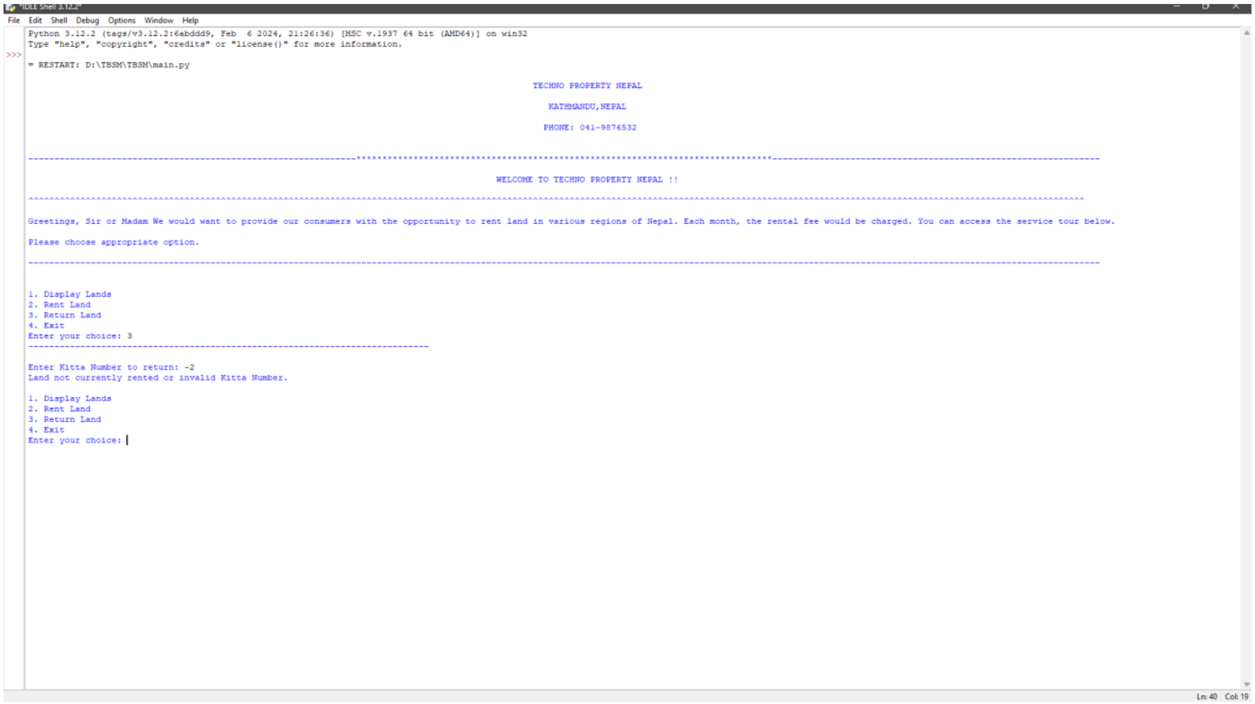
```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: D:\TBSM\TBSM\main.py

                                                                                           TECHNO PROPERTY NEPAL

                                                                                             KATHMANDU,NEPAL

                                                                                            PHONE: 041-9876532


---------------------------------------------------------------*******************************************************************************---------------------------------------------------------------

                                                                                       WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 2
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available
----------------------------------------------------------------------

Enter Kitta Number to rent: 8
Land Unavailable or Please enter the correct Kitta No.
Thanks for visting us

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice:
```

Figure 14 : Non-existed value as input while renting lands

7.2.2. Testing - Negative and non-existed value as input while returning lands

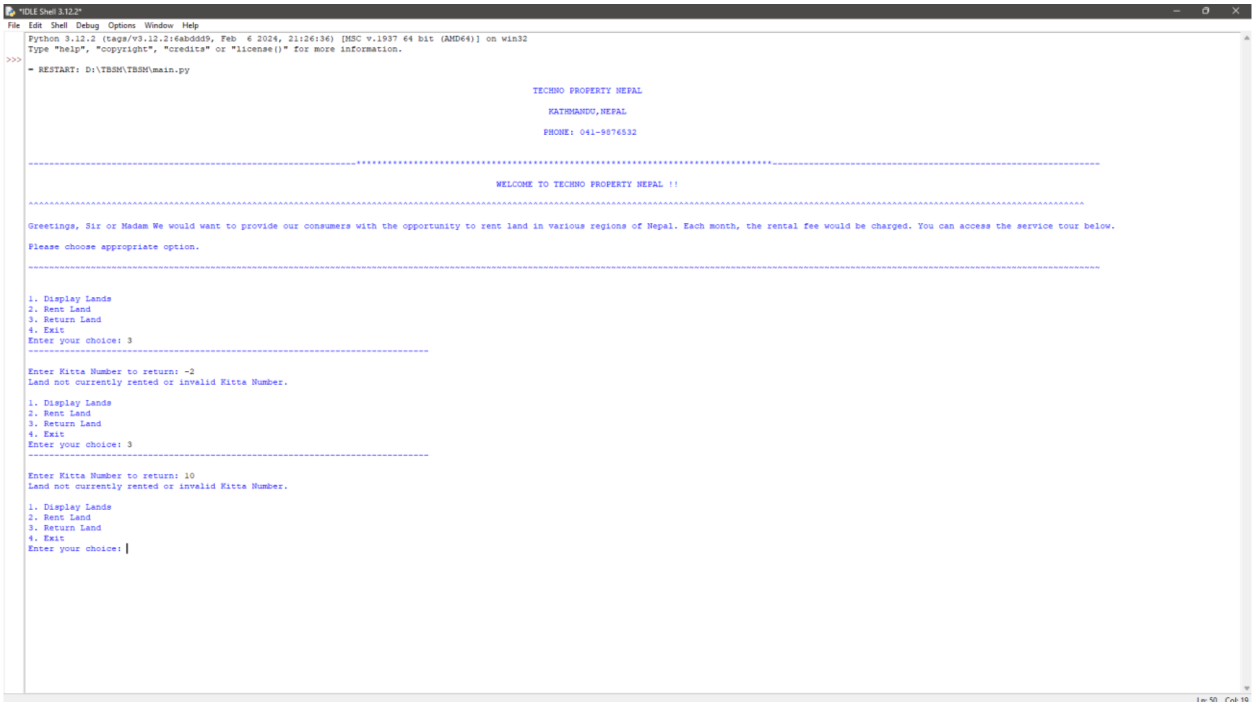
Objective	Negative and non-existed value as input while returning lands
Action	The main program was executed. • When renting land, a negative value of -2 was first entered. • Later, a non-existing value of 10 was entered.
Expected result	In both cases, the user should to see an error message and be given chance to choose the right response.
Actual result	The expected and actual results were same. The user was able to choose the correct option after seeing an error warning on the screen.
Conclusion	Test successful

Table 4 : Negative and non-existed value as input while returning lands



```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: D:\TBSM\TBSM\main.py

                                                            TECHNO PROPERTY NEPAL

                                                             KATHMANDU,NEPAL

                                                            PHONE: 041-9876532


----------------------------------------------------------------*******************************************************************************----------------------------------------------------------------

                                                       WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 3
-----------------------------------------------------------------------------

Enter Kitta Number to return: -2
Land not currently rented or invalid Kitta Number.

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 
```

Figure 15 : Non-existed value as input while returning lands



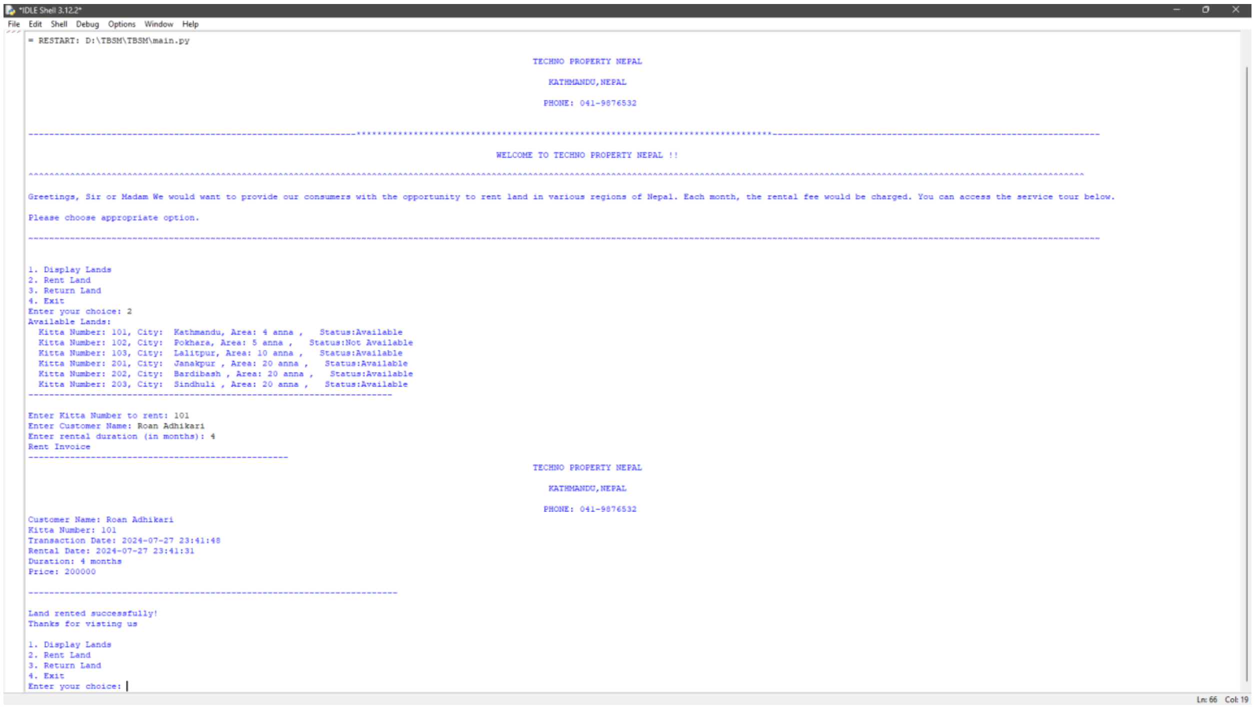
```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

= RESTART: D:\TBSM\TBSM\main.py

                                                                                          TECHNO PROPERTY NEPAL

                                                                                            KATHMANDU,NEPAL

                                                                                          PHONE: 041-9876532


---------------------------------------------------------------*******************************************************************************---------------------------------------------------------------

                                                                                   WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 3
-----------------------------------------------------------------------------

Enter Kitta Number to return: -2
Land not currently rented or invalid Kitta Number.

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 3
-----------------------------------------------------------------------------

Enter Kitta Number to return: 10
Land not currently rented or invalid Kitta Number.

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 
```

Figure 16 : Negative value as input while returning lands

7.3. Testing - The file generation while renting lands

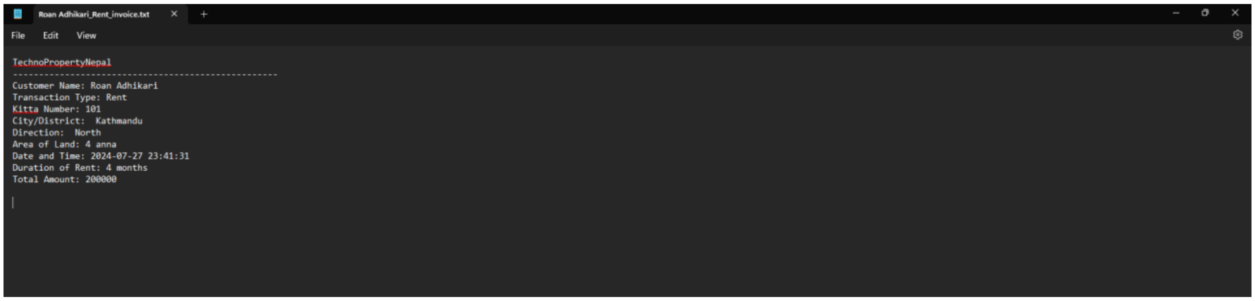
Objective	The file generation while renting lands
Action	The main program was executed. The following tasks was done while renting lands: ➢ The second option was entered for renting. ➢ The Kitta no. 101 was entered. ➢ The number of months was entered that was 4 Personal information of customer was entered for generating the invoice in the shell as well as in the file. ➢ Name: Roan Adhikari
Expected result	After finishing the rental procedure, an invoice with transaction data should be displayed in the shell and in the file.
Actual result	The expected and actual results were same. An invoice was appeared on the screen along with the details of transaction and was also generated in the file.
Conclusion	Test successful

Table 5 : The file generation while renting lands



```
= RESTART: D:\TBSM\TBSM\main.py
                                                                                TECHNO PROPERTY NEPAL

                                                                                  KATHMANDU,NEPAL

                                                                                 PHONE: 041-9876532


--------------------------------------------------------------**********************************************************************--------------------------------------------------------------

                                                                          WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 2
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available
----------------------------------------------------------------------

Enter Kitta Number to rent: 101
Enter Customer Name: Roan Adhikari
Enter rental duration (in months): 4
Rent Invoice
--------------------------------------------------
                                                                                TECHNO PROPERTY NEPAL

                                                                                  KATHMANDU,NEPAL

                                                                                 PHONE: 041-9876532
Customer Name: Roan Adhikari
Kitta Number: 101
Transaction Date: 2024-07-27 23:41:48
Rental Date: 2024-07-27 23:41:31
Duration: 4 months
Price: 200000

----------------------------------------------------------------------

Land rented successfully!
Thanks for visting us

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 
```

Figure 17 : The file generation while renting lands



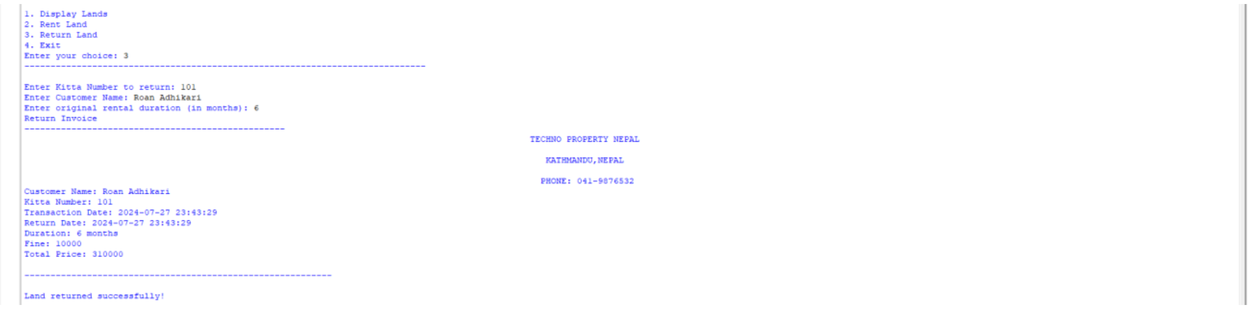
```
TechnoPropertyNepal
--------------------------------------------------
Customer Name: Roan Adhikari
Transaction Type: Rent
Kitta Number: 101
City/District:  Kathmandu
Direction:  North
Area of Land: 4 anna
Date and Time: 2024-07-27 23:41:31
Duration of Rent: 4 months
Total Amount: 200000
```

Figure 18 : The file generation while renting lands in notepad

7.4. Test 4 - The file generation while returning lands

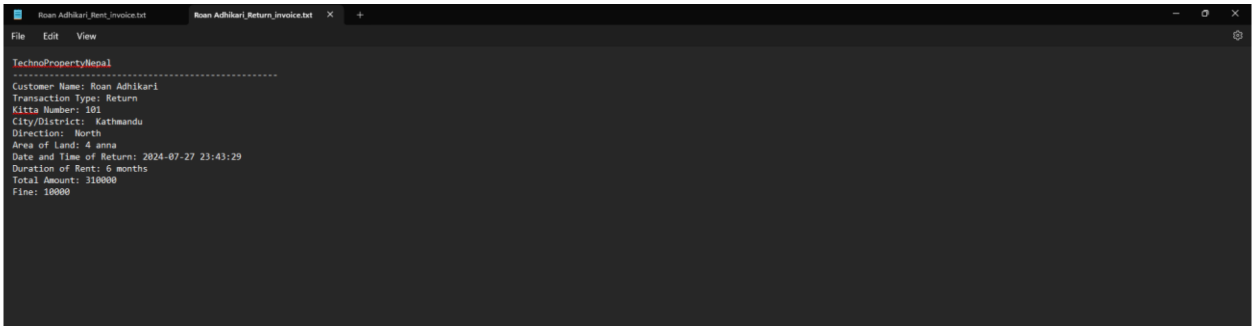
Objective	The file generation while returning lands
Action	The main program was executed. • The following tasks was done while returning multiple lands: ➢ The third option was entered for returning land. ➢ The Kitta no. 101 was entered. ➢ The number of months was entered while renting that was 4 ➢ The number of months was entered while returning that was 6 • Personal information of customer was entered for generating the invoice in the shell as well as in the file. ➢ Name: Roan Adhikari
Expected result	After finishing the returning procedure, an invoice with transaction data should be displayed in the shell and in the file.
Actual result	The expected and actual results were same. An invoice was appeared on the screen along with the details of transaction and was also generated in the file.
Conclusion	Test successful

Table 6 : The file generation while returning lands



```
1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 3
-----------------------------------------------------------------------------

Enter Kitta Number to return: 101
Enter Customer Name: Roan Adhikari
Enter original rental duration (in months): 6
Return Invoice
--------------------------------------------------
                                                                                                   TECHNO PROPERTY NEPAL

                                                                                                      KATHMANDU,NEPAL

                                                                                                    PHONE: 041-9876532
Customer Name: Roan Adhikari
Kitta Number: 101
Transaction Date: 2024-07-27 23:43:29
Return Date: 2024-07-27 23:43:29
Duration: 6 months
Fine: 10000
Total Price: 310000

-----------------------------------------------------------

Land returned successfully!
```

Figure 19 : The file generation while returning lands


Roan Adhikari_Rent_invoice.txt
Roan Adhikari_Return_invoice.txt
File
Edit
View

```
TechnoPropertyNepal
--------------------------------------------------
Customer Name: Roan Adhikari
Transaction Type: Return
Kitta Number: 101
City/District:  Kathmandu
Direction:  North
Area of Land: 4 anna
Date and Time of Return: 2024-07-27 23:43:29
Duration of Rent: 6 months
Total Amount: 310000
Fine: 10000
```

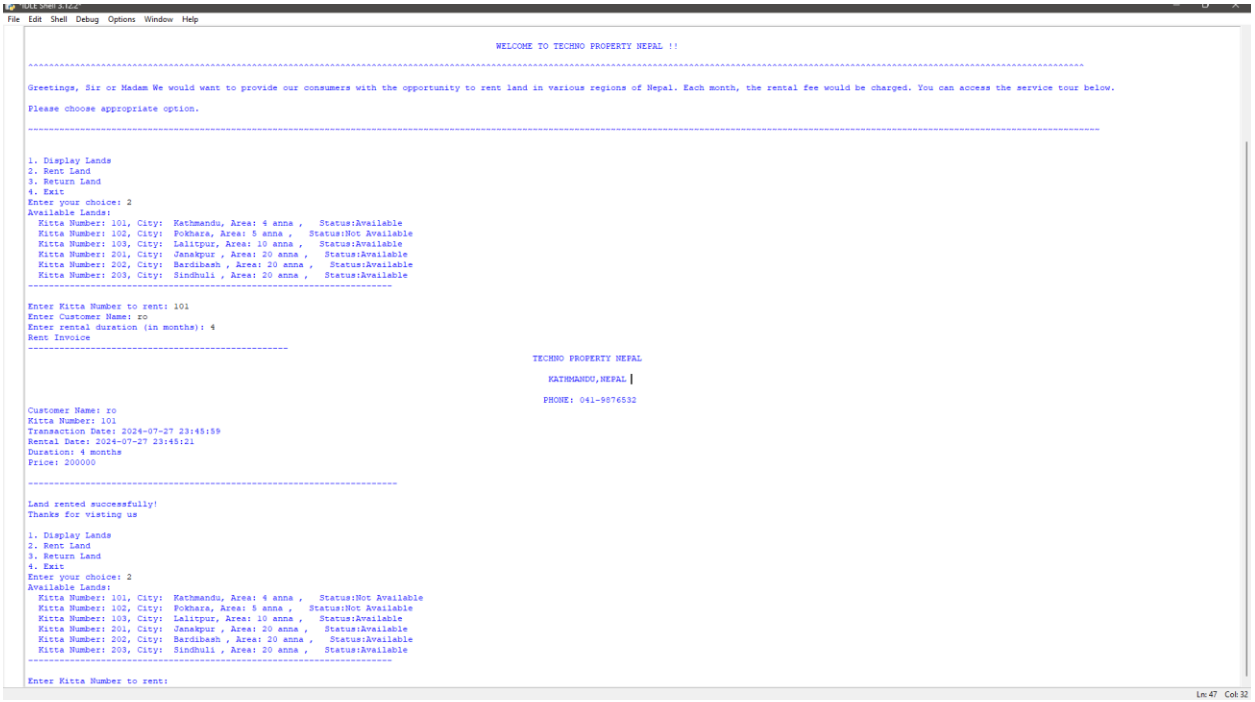
Figure 20 : The file generation while returning lands in notepad

7.5. Test 5

7.5.1. Testing – The availability land being rented to 'Not available' after renting the land

Objective	The availability land being rented to 'Not available' after renting the land
Action	The main program was executed. The anna of all the land were checked before renting. • Multiple land was rented. • After renting, the anna of all the land were checked again after renting.
Expected result	The available land should be updated as 'Not available' after renting the land
Actual result	The available land updated as 'Not available' after renting the land
Conclusion	Test successful

Table 7 : The availability land being rented to 'Not available' after renting the land



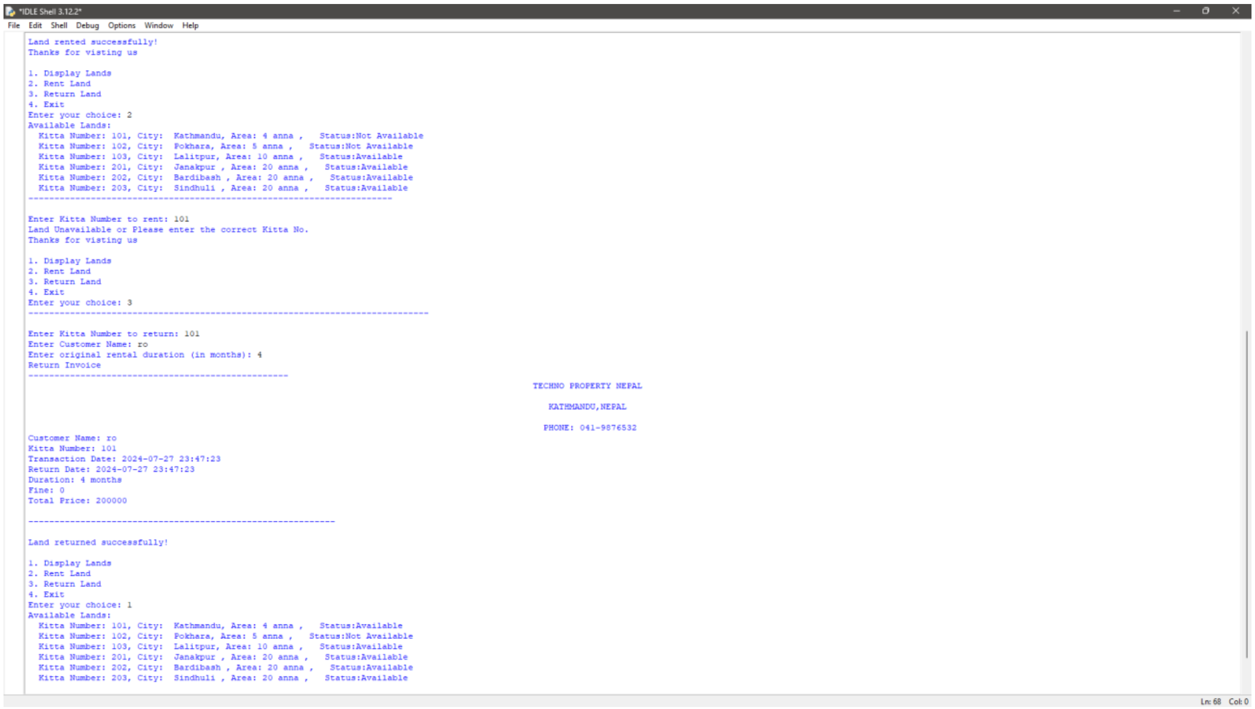
```
                                                              WELCOME TO TECHNO PROPERTY NEPAL !!

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Greetings, Sir or Madam We would want to provide our consumers with the opportunity to rent land in various regions of Nepal. Each month, the rental fee would be charged. You can access the service tour below.

Please choose appropriate option.

--------------------------------------------------------------------------------------------------------------------------------------------------


1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 2
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available
----------------------------------------------------------------------

Enter Kitta Number to rent: 101
Enter Customer Name: ro
Enter rental duration (in months): 4
Rent Invoice
--------------------------------------------------
                                                                     TECHNO PROPERTY NEPAL

                                                                      KATHMANDU,NEPAL

                                                                     PHONE: 041-9876532
Customer Name: ro
Kitta Number: 101
Transaction Date: 2024-07-27 23:45:59
Rental Date: 2024-07-27 23:45:21
Duration: 4 months
Price: 200000

-----------------------------------------------------------------------

Land rented successfully!
Thanks for visting us

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 2
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Not Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available
----------------------------------------------------------------------

Enter Kitta Number to rent:
```

Figure 21 : The availability land being before renting the land and after renting the land

7.5.2. Testing – The availability of land being returned to 'Available' after returning the land

Objective	The availability land being returned to 'Available' after returning the land
Action	The main program was executed. The anna of all the land were checked before returning. • All rented land was returned. • After returning, the anna of all the land were checked again
Expected result	The not available land should be updated as 'available' after renting the land
Actual result	The not available land updated as 'available' after renting the land
Conclusion	Test successful

Table 8 : The availability of land being returned to 'Available' after returning the land



```
Land rented successfully!
Thanks for visting us

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 2
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Not Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available
----------------------------------------------------------------------

Enter Kitta Number to rent: 101
Land Unavailable or Please enter the correct Kitta No.
Thanks for visting us

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 3
-----------------------------------------------------------------------------

Enter Kitta Number to return: 101
Enter Customer Name: ro
Enter original rental duration (in months): 4
Return Invoice
--------------------------------------------------
                                                                                                 TECHNO PROPERTY NEPAL

                                                                                                    KATHMANDU,NEPAL

                                                                                                   PHONE: 041-9876532

Customer Name: ro
Kitta Number: 101
Transaction Date: 2024-07-27 23:47:23
Return Date: 2024-07-27 23:47:23
Duration: 4 months
Fine: 0
Total Price: 200000

-----------------------------------------------------------

Land returned successfully!

1. Display Lands
2. Rent Land
3. Return Land
4. Exit
Enter your choice: 1
Available Lands:
  Kitta Number: 101, City:  Kathmandu, Area: 4 anna ,   Status:Available
  Kitta Number: 102, City:  Pokhara, Area: 5 anna ,   Status:Not Available
  Kitta Number: 103, City:  Lalitpur, Area: 10 anna ,   Status:Available
  Kitta Number: 201, City:  Janakpur , Area: 20 anna ,   Status:Available
  Kitta Number: 202, City:  Bardibash , Area: 20 anna ,   Status:Available
  Kitta Number: 203, City:  Sindhuli , Area: 20 anna ,   Status:Available
```

Figure 22 : The availability of land being returned to ' not Available' before returning the land and being returned to ‘Available' after returning the land

8. Conclusion

Throughout the project, we thoroughly understood the fundamentals of the Python programming language. We learned fundamental concepts such as file handling, exception handling, collection data types, and many more. We came up with a lot of new ideas while designing a purchasing program based on real-world events. As an individual enrolled in this course, I had the opportunity to put my Python programming skills, thoughts, and knowledge to the test.

When creating a program that offers rental and return services, we originally fell into a number of problems and obstacles. Presenting all the land details in a tabular style in the shell, including the price, direction, and name of the city, was really challenging for me. The 2-dimensional list was not suitable for my project. We didn't know where to begin or how to approach the assignment. After an excessive amount of research, commitment, and labour, we were able to construct the program's fundamental user interface. We had a lot of assistance from our teachers and other friends on our great journey. When defining our own function, we also encountered a lot of syntax errors. We used to forget to apply proper indentation while creation of code, which upset me a lot. In other hands, exception management was yet another significant challenge in developing this program. We used to enter string values as input instead of numeric values, which caused my program to crash while processing land rentals or returns.

Finally, we were able to successfully complete my project with the support of my fellow friends and teachers, who advised me on what to do and not do when developing the program. We encountered syntactic issues while implementing the idea of file handling. But in the end, we finished my homework after giving my full attention, time, and making some compromises. The concepts and skills we gained in this course will assist me in obtaining placements with major technology companies such as Google, Twitter, and others. This course of study also helped me improve my problem-solving abilities, creative thinking, and thinking capacity for new ideas. Overall, we liked the journey of designing this application.

9. Bibliography

Alder, G., 2000. *Diagram files.* [Online] Available at: https://www.drawio.com/
[Accessed 02 April 2024].

Brodie, R., 1983. *Notepad.* [Online] Available at: https://apps.microsoft.com/detail/9msmlrh6lzf3?hl=en-US&gl=US
[Accessed 4 April 2024].

Charles Simonyi, R. B., 1983. *MS Word.* [Online] Available at: https://www.microsoft.com/en/microsoft-365/word?market=af
[Accessed 03 April 2024].

Rossum, G. v., 1998. *Python IDLE.* [Online] Available at: https://docs.python.org/3/library/idle.html
[Accessed 02 April 2024].

10. Appendix

10.1. Main.py

```
import read
import operation

# this function loads all the land inventory information store in my file

def save_land_data(filename, inventory):
   with open(filename, 'w') as file:
       for kitta_no, land_data in inventory.items():
          file.write(f"{kitta_no},{land_data['city']},{land_data['direction']},{land_data['a
rea']},{land_data['price']},{land_data['status']}\n")

# this one here we rent our land based on the kita number

def main():
    inventory = read.load_land_data('land.txt')
    print()
    print("\t\t\t\t\t\t\t\t\t\t\t TECHNO PROPERTY NEPAL
\n\n\t\t\t\t\t\t\t\t\t\t\t    KATHMANDU,NEPAL \n\n\t\t\t\t\t\t\t\t\t\t\t   PHONE: 041-
9876532 \n" )
    print()
    print("-----------------------------------------------------------
****************************************************************************-------------
----------------------------------------------")
    print()
    print("\t\t\t\t\t\t\t\t\t\t\t  WELCOME TO TECHNO PROPERTY NEPAL !! \n")
    print("^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^")
    print()
    print("Greetings, Sir or Madam We would want to provide our consumers with
the opportunity to rent land in various regions of Nepal. Each month, the rental
fee would be charged. You can access the service tour below.  \n")
    print("Please choose appropriate option.")
    print()
    print("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~")
    print()
    while True:
```

```
        print()
        print("1. Display Lands")
        print("2. Rent Land")
        print("3. Return Land")
        print("4. Exit")

        choice = operation.get_integer_input("Enter your choice: ")

        if choice == 1:
            read.display_available_lands(inventory)

        elif choice == 2:
            read.display_available_lands(inventory)
            operation.rent_land(inventory)
            print("Thanks for visting us")

        elif choice == 3:
            operation.return_land(inventory)
        elif choice == 4:
            print("Exiting... ")
            print("Thanks for Visting Us")
            return
        else:
            print("Invalid Choice")

    save_land_data('land_inventory.txt', inventory)

if __name__ == '__main__':
    main()
```

10.2. Operation.py

```
import datetime
import write  # Assuming write is a module that handles customer transactions
and invoice generation

def generate_invoice_terminal(invoice_type, customer_name, kitta_no,
duration=None, fine=None, rental_date=None, return_date=None,
price_per_month=None):
    now = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    invoice = f"{invoice_type.capitalize()} Invoice\n"
    invoice += "--------------------------------------------------\n"

    invoice +="\t\t\t\t\t\t\t\t\t\t\t TECHNO PROPERTY NEPAL
\n\n\t\t\t\t\t\t\t\t\t\t\t    KATHMANDU,NEPAL \n\n\t\t\t\t\t\t\t\t\t\t\t   PHONE: 041-
9876532 \n"

    invoice += f"Customer Name: {customer_name}\n"
    invoice += f"Kitta Number: {kitta_no}\n"
    invoice += f"Transaction Date: {now}\n"

    if invoice_type == "rent":
        invoice += f"Rental Date: {rental_date}\n"
        invoice += f"Duration: {duration} months\n"
        invoice += f"Price: {duration * price_per_month}\n"
    elif invoice_type == "return":
        invoice += f"Return Date: {return_date}\n"
        invoice += f"Duration: {duration} months\n"
        invoice += f"Fine: {fine}\n"
        invoice += f"Total Price: {duration * price_per_month + fine}\n"

    print(invoice)

def rent_land(inventory):
    now = datetime.datetime.now()
    print("------------------------------------------------------------------------------\n")
    kitta_no = input("Enter Kitta Number to rent: ")

    if kitta_no in inventory and inventory[kitta_no]['status'] == 'Available':
        customer_name = input("Enter Customer Name: ")

        duration = get_integer_input("Enter rental duration (in months): ")

        inventory[kitta_no]['status'] = 'Not Available'

        if customer_name not in write.customer_transactions:
            write.customer_transactions[customer_name] = []

        write.customer_transactions[customer_name].append({
            'transaction_type': 'Rent',
            'land_info': {'kitta_no': kitta_no, **inventory[kitta_no]},
            'duration': duration,
            'rented_date': now.strftime('%Y-%m-%d %H:%M:%S')
        })

        write.generate_customer_invoice(customer_name)
        generate_invoice_terminal("rent", customer_name, kitta_no,
duration=duration, rental_date=now.strftime('%Y-%m-%d %H:%M:%S'),
price_per_month=inventory[kitta_no]['price'])
        print("-----------------------------------------------------------------------\n")
        print("Land rented successfully!")
    else:
        print("Land Unavailable or Please enter the correct Kitta No.")

    return inventory

def return_land(inventory):
    print("-----------------------------------------------------------------------------\n")
    kitta_no = input("Enter Kitta Number to return: ")

    if kitta_no in inventory and inventory[kitta_no]['status'] == 'Not Available':
        customer_name = input("Enter Customer Name: ")

        return_rented_duration = get_integer_input("Enter original rental duration (in
months): ")

        now = datetime.datetime.now()
        return_date = now.strftime('%Y-%m-%d %H:%M:%S')

        try:
            actual_rented_duration =
write.customer_transactions[customer_name][0]['duration']
        except (KeyError, IndexError):
            print("This customer hasn't done any transactions before.")
            return return_land(inventory)


        # Calculate total rented months

        '''
        this here is hard coded
        '''
        # Adding 1 to include the current month

        # Calculate months late
        months_late = max(0, return_rented_duration - actual_rented_duration)
        price_per_month = inventory[kitta_no]['price']

        # Calculate the fine
        fine = round(0.1 * months_late * price_per_month)

        # Generate and display invoice
        write.generate_return_invoice(customer_name, kitta_no, inventory[kitta_no],
return_rented_duration, fine)
        generate_invoice_terminal("return", customer_name, kitta_no,
duration=return_rented_duration, fine=fine, return_date=return_date,
price_per_month=price_per_month)

        # Update inventory status
        inventory[kitta_no]['status'] = 'Available'
        print("-------------------------------------------------------\n")
        print("Land returned successfully!")
    else:
        print("Land not currently rented or invalid Kitta Number.")

def get_integer_input(prompt, error_message="Invalid input. Please enter a
number."):
    while True:
        try:
            return int(input(prompt))
        except ValueError:
            print(error_message)
```

10.3. Read.py

```
def display_available_lands(inventory):
    print("Available Lands:")
    for kitta_no, land_data in inventory.items():
        # if status of the land is 'available' then we simply show it to the user that the
land is available for rent
            print(f"  Kitta Number: {kitta_no}, City: {land_data['city']}, Area:
{land_data['area']} anna ,   Status:{land_data['status']}")

def load_land_data(filename):
     inventory = {}
     with open(filename, 'r') as file:
        for line in file:
            data = line.strip().split(',')
            # i am using the dictionary data structure to store the land inventory
information
            inventory[data[0]] = {
                'city': data[1],
                'direction': data[2],
                'area': int(data[3]),
                'price': int(data[4]),
                'status': data[5].strip()
            }
        return inventory
```

10.4. Write.py

```
import datetime

customer_transactions = {}
def generate_customer_invoice(customer_name):
    if customer_name in customer_transactions:
        with open(f'{customer_name}_Rent_invoice.txt', 'w') as f:
            f.write("TechnoPropertyNepal\n")
            f.write("--------------------------------------------------\n")
            total_amount = 0
            for transaction in customer_transactions[customer_name]:
                f.write(f'Customer Name: {customer_name}\n')
                f.write(f'Transaction Type: {transaction["transaction_type"]}\n')
                f.write(f'Kitta Number: {transaction["land_info"]["kitta_no"]}\n')
                f.write(f'City/District: {transaction["land_info"]["city"]}\n')
                f.write(f'Direction: {transaction["land_info"]["direction"]}\n')
                f.write(f'Area of Land: {transaction["land_info"]["area"]} anna\n')
                f.write(f'Date and Time: {transaction["rented_date"]}\n')
                f.write(f'Duration of Rent: {transaction["duration"]} months\n')
                total_amount += transaction["duration"] *
transaction["land_info"]["price"]
            f.write(f'Total Amount: {total_amount}\n\n')
    else:
        print("No transactions found for this customer.")

def generate_return_invoice(customer_name, kitta_no, land_info, duration, fine):
    now = datetime.datetime.now()
    if customer_name in customer_transactions:
        with open(f'{customer_name}_Return_invoice.txt', 'w') as f:
            f.write("TechnoPropertyNepal\n")
            f.write("--------------------------------------------------\n")
            f.write(f'Customer Name: {customer_name}\n')
            f.write(f'Transaction Type: Return\n')
            f.write(f'Kitta Number: {kitta_no}\n')
            f.write(f'City/District: {land_info["city"]}\n')
            f.write(f'Direction: {land_info["direction"]}\n')
            f.write(f'Area of Land: {land_info["area"]} anna\n')
            f.write(f'Date and Time of Return: {now.strftime("%Y-%m-%d
%H:%M:%S")}\n')
            f.write(f'Duration of Rent: {duration} months\n')
            total_amount = duration * land_info['price'] + fine
            f.write(f'Total Amount: {total_amount}\n')
            f.write(f'Fine: {fine}\n\n')
    else:
        print("No transactions found for this customer.")
```