TASK 9 Implement Exceptions and Exceptional handling in Python.

PROBLEM 1

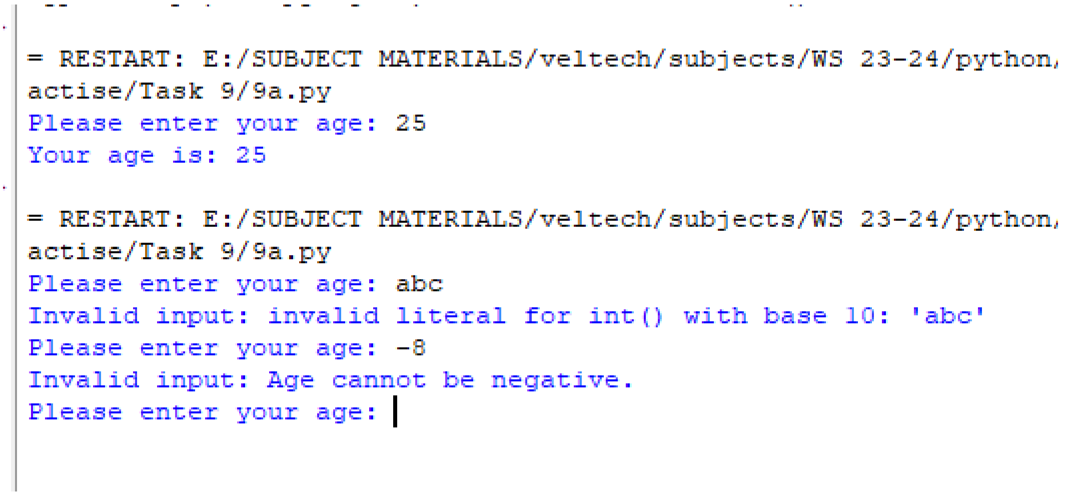
You're developing a program for a quiz application where users need to input their ages to participate. You want to ensure that only valid integers are accepted as ages. Write a Python program that prompts the user to input their age, and if the input is not a valid integer, it raises a ValueError exception.

PROGRAM

```
def get_age():
    while True:
        try:
            age = int(input("Please enter your age: "))
            if age < 0:
                raise ValueError("Age cannot be negative.")
            return age
        except ValueError as ve:
            print("Invalid input:", ve)


try:
    age = get_age()
    print("Your age is:", age)
except ValueError as ve:
    print("An error occurred:", ve)
```

OUTPUT



```
= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/python,
actise/Task 9/9a.py
Please enter your age: 25
Your age is: 25

= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/python,
actise/Task 9/9a.py
Please enter your age: abc
Invalid input: invalid literal for int() with base 10: 'abc'
Please enter your age: -8
Invalid input: Age cannot be negative.
Please enter your age: 
```

PROBLEM 2

You're developing a program for a ticketing system where users can view details of available seats in a venue. Each seat is represented by a list of attributes such as seat number, section, and availability status. You want to implement a feature that allows users to select a seat from the list and perform an operation on it, like marking it as booked or displaying its details. Write a Python program that executes an operation on a list representing seats and handles an IndexError exception if the index provided by the user is out of range.

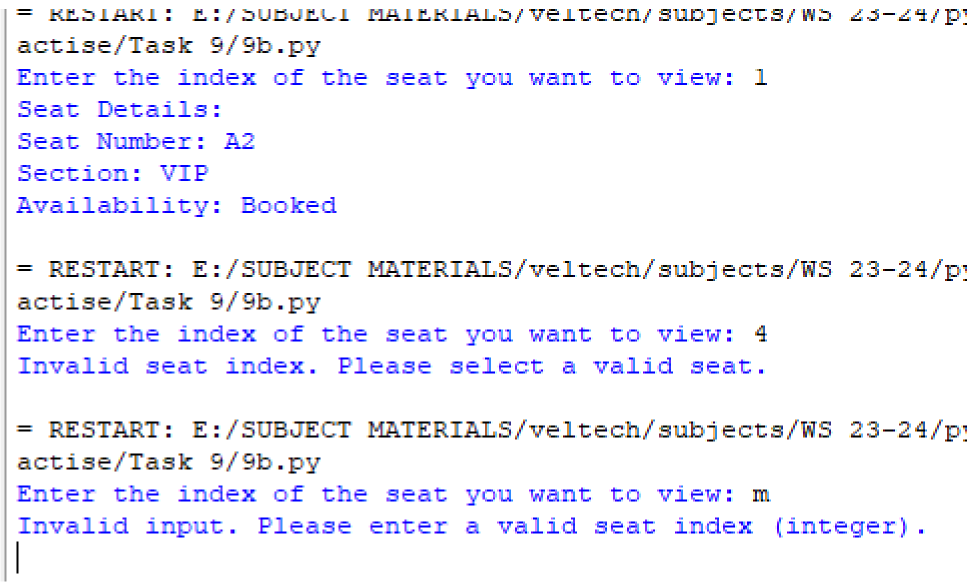
PROGRAM

```
def view_seat_details(seat_list, index):
    try:
        seat = seat_list[index]
        print("Seat Details:")
        print("Seat Number:", seat.get("number"))
        print("Section:", seat.get("section"))
        print("Availability:", "Booked" if seat.get("is_booked") else "Available")
    except IndexError:
        print("Invalid seat index. Please select a valid seat.")


# Sample list of seats
seats = [
    {"number": "A1", "section": "VIP", "is_booked": False},
    {"number": "A2", "section": "VIP", "is_booked": True},
    {"number": "B1", "section": "General", "is_booked": False},
    {"number": "B2", "section": "General", "is_booked": False}
]

# Test the function
try:
    seat_index = int(input("Enter the index of the seat you want to view: "))
    view_seat_details(seats, seat_index)
except ValueError:
    print("Invalid input. Please enter a valid seat index (integer).")
```

OUTPUT



```
= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/p
actise/Task 9/9b.py
Enter the index of the seat you want to view: 1
Seat Details:
Seat Number: A2
Section: VIP
Availability: Booked

= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/p
actise/Task 9/9b.py
Enter the index of the seat you want to view: 4
Invalid seat index. Please select a valid seat.

= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/p
actise/Task 9/9b.py
Enter the index of the seat you want to view: m
Invalid input. Please enter a valid seat index (integer).
```

PROBLEM 3

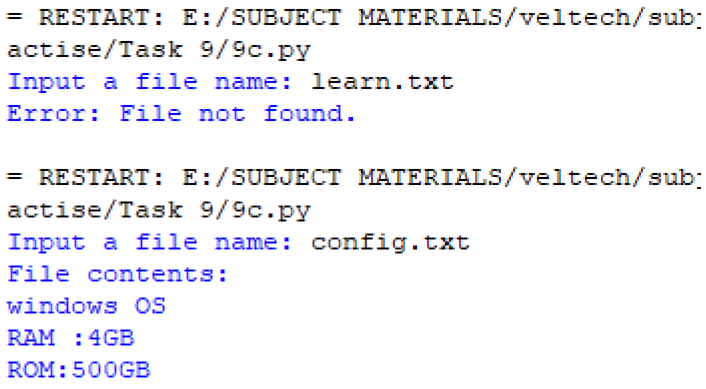
Develop a Python program for a company's data processing system. The program needs to read data from a configuration file named **config.txt** which contains important settings for the system. Write a Python program that opens the **config.txt** file and handles a **FileNotFoundError** exception if the file does not exist. If the file exists, it should print out the contents of the configuration file.

PROGRAM

```
def open_file(filename):
    try:
        # Attempt to open the specified file in read mode ('r').
        file = open(filename, 'r')
        # Read the contents of the file and store them in the 'contents' variable.
        contents = file.read()
        # Print a message to indicate that the file contents will be displayed.
        print("File contents:")
        # Print the contents of the file.
        print(contents)
        # Close the file to release system resources.
        file.close()
    except FileNotFoundError:
        # Handle the exception if the specified file is not found.
        print("Error: File not found.")


# Prompt the user to input a file name and store it in the 'file_name' variable.
file_name = input("Input a file name: ")
# Call the open_file function with the provided file name.
open_file(file_name)
```

OUTPUT



```
= RESTART: E:/SUBJECT MATERIALS/veltech/sub
actise/Task 9/9c.py
Input a file name: learn.txt
Error: File not found.

= RESTART: E:/SUBJECT MATERIALS/veltech/sub
actise/Task 9/9c.py
Input a file name: config.txt
File contents:
windows OS
RAM :4GB
ROM:500GB
```

PROBLEM 4

During testing, a user attempted to use your calculator to calculate the return on investment (ROI) for an investment portfolio. However, instead of entering a numeric value, the user mistakenly entered a string ("apple") when prompted to input the initial investment amount. Discuss the error message displayed to the user and how the program guides them to correct the input error using multiple except in python.

PROGRAM

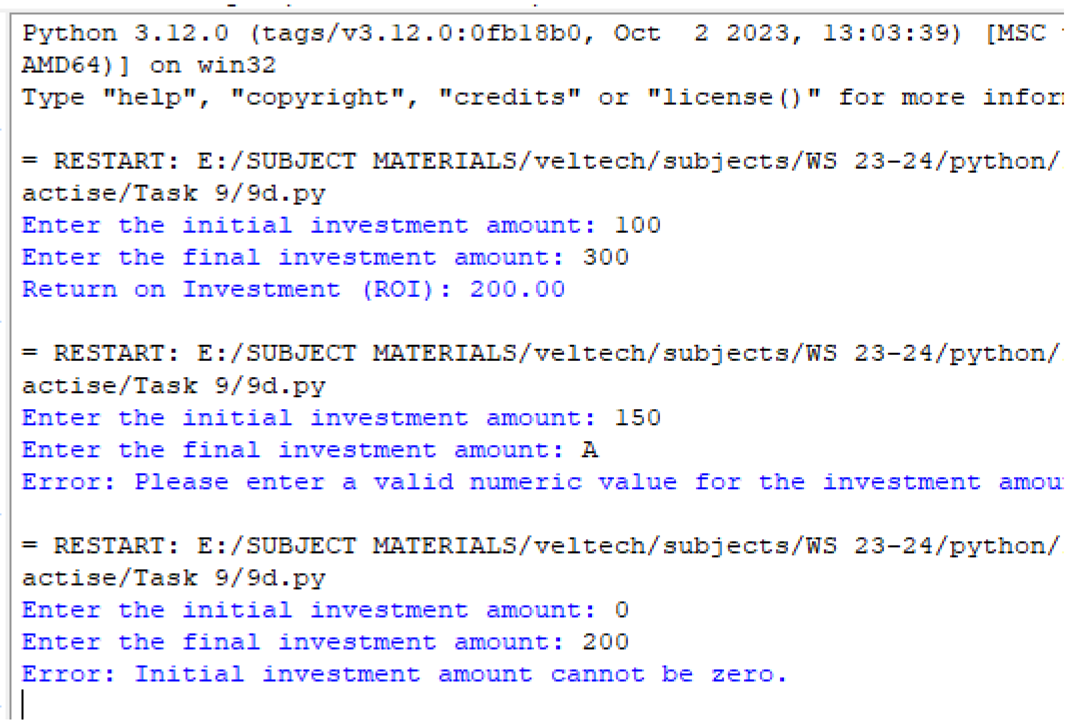
```
try:
   # Attempt to get the initial investment amount from the user
   initial_investment = float(input("Enter the initial investment amount: "))

   # Attempt to get the final investment amount from the user
   final_investment = float(input("Enter the final investment amount: "))

   # Calculate the return on investment (ROI)
   roi = (final_investment - initial_investment) / initial_investment * 100
   print("Return on Investment (ROI): {:.2f}".format(roi))

except ValueError:
   # Handle the ValueError exception
   print("Error: Please enter a valid numeric value for the investment amount.")
except ZeroDivisionError:
   # Handle the ZeroDivisionError exception
   print("Error: Initial investment amount cannot be zero.")
```

OUTPUT



```
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct  2 2023, 13:03:39) [MSC
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more infor

= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/python/
actise/Task 9/9d.py
Enter the initial investment amount: 100
Enter the final investment amount: 300
Return on Investment (ROI): 200.00

= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/python/
actise/Task 9/9d.py
Enter the initial investment amount: 150
Enter the final investment amount: A
Error: Please enter a valid numeric value for the investment amou

= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/python/
actise/Task 9/9d.py
Enter the initial investment amount: 0
Enter the final investment amount: 200
Error: Initial investment amount cannot be zero.
```