

# Module 4: Working with Modules and Handling Exceptions

---

## Assignment – 1

edureka!

**edureka!**

© Brain4ce Education Solutions Pvt. Ltd.

1. A Robot moves in a Plane starting from the origin point (0,0). The robot can move UP, DOWN, LEFT, and RIGHT. The trace of Robot movement is as given following:

UP 5  
DOWN 3  
LEFT 3  
RIGHT 2

(The numbers after directions are steps)

Write a program to compute the current distance from the origin point after sequencing of movements.

**Hint:** Use the math module.

2. Data of XYZ company is stored in a sorted list. Write a program to search for specific data from that list.

**Hint:** Use if/elif to deal with conditions.

3. A weather forecasting organization wants to show whether it is day or night. Write a program to find whether is it dark outside or not based on the local system time.

**Hint:** Use the time module.

4. Write a program to find the distance between two locations when their latitude and longitudes are given.

**Hint:** Use the math module.

5. Design a banking system software with options like cash withdrawal, cash credit, and change password. The software must display appropriate results based on user inputs.

**Hint:** Use if else statements and functions.

6. Write a program that will find all numbers which are divisible by 7 but are not a multiple of 5, between 2,000 and 3,200 (both included). The numbers obtained should be printed in a comma-separated sequence on a single line.
7. Write a program that can compute the factorial of given numbers. Use recursion to find it.
8. Write a program that calculates and prints the value according to the given formula:  
$$Q = \text{Square root of } [(2 * C * D)/H]$$
Following are the fixed values of C and H: C is 50. H is 30.  
D is the variable whose values should be input to your program in a comma-separated sequence.

### Example

Let us assume the following comma-separated input sequence is given to the program:

100,150,180

The output of the program should be:

18,22,24

9. Write a program that takes 2 digits, X, Y as input and generates a 2-dimensional array. The element value in the *i*th row and *j*th column of the array should be  $i*j$ .  
**Note:**  $i = 0, 1, \dots, X - 1$ ;  $j = 0, 1, \dots, Y - 1$ .

### Example

Suppose the following inputs are given to the program:

3,5

Then, the output of the program should be:

[[0, 0, 0, 0, 0], [0, 1, 2, 3, 4], [0, 2, 4, 6, 8]]

10. Write a program that accepts a comma-separated sequence of words as input and prints the words in a comma-separated sequence after sorting them alphabetically.

Suppose the following input is supplied to the program:

without, hello, bag, world

Then, the output should be:

bag, hello, without, world

11. Write a program that accepts a sequence of lines as input and prints the lines after making all characters in the sentence capitalized.

Suppose the following input is supplied to the program:

- Hello world
- Practice makes perfect

Then, the output should be:

- HELLO WORLD
- PRACTICE MAKES PERFECT

12. Write a program that accepts a sequence of whitespace-separated words as input and prints the words after removing all duplicate words and sorting them alphanumerically.

Suppose the following input is supplied to the program:

hello world and practice makes perfect and hello world again

Then, the output should be:

again and hello makes perfect practice world

13. Write a program that accepts a sequence of comma-separated 4-digit binary numbers as its input and then check whether they are divisible by 5 or not. The numbers that are divisible by 5 are to be printed in a comma-separated sequence.

**Example**

0100,0011,1010,1001

Then the output should be:

1010

14. Write a program that accepts a sentence and calculate the number of upper-case letters and lower-case letters.

Suppose the following input is supplied to the program:

Hello world!

Then, the output should be:

UPPER CASE 1

LOWER CASE 9

15. Give an example of fsum and sum function of math library.

16. Case Study:

**Domain:** Banking Marketing

**Focus:** Optimization

#### Business challenge/requirement

Bank of Portugal runs a marketing campaign to offer loans to clients. The loan is offered to only clients in selective professions.

A list of successful campaigns (with client data) is given in the attached data set. You must come up with a program that reads the file and builds a set of unique profession lists. Based on the given input professions of the client, the system tells whether the client is eligible to be approached for the marketing campaign.

#### Key issues

Tele Caller can only make x number of cold calls in a day. Hence to increase her effectiveness only eligible customers should be contacted.

#### Considerations

The current system does not differentiate clients based on age and profession.

#### Data volume

447 records in bank-data.csv

#### Additional information

- NA

#### Business benefits

The company can achieve between 15% to 20% higher conversion by targeting the right clients.

#### Approach to Solve

You must use the fundamentals of Python taught in Module 3.

1. Read file bank-data.csv
2. Build a set of unique jobs
3. Read the input from the command line – profession
4. Check if the profession is on the list
5. Print whether the client is eligible

### Enhancements for code

You can try these enhancements in code

1. Compute max and min age for loan eligibility based on data in csv file.
2. Store max and min age in the dictionary.
3. Make the professional check case insensitive.

Currently program ends after the check. Take the input in the **while** loop and end only if the user types "END" for profession.

### 17. Case Study:

**Domain:** E-Commerce

**Focus:** Optimization

### Business challenge/requirement

GoodsKart—the largest e-commerce company in Indonesia, with revenue of \$2B+ has acquired another e-commerce company FairDeal. FairDeal has its own IT system to maintain the records of customers, sales, and so on. For ease of maintenance and cost savings, GoodsKart is integrating customer databases of both organizations hence customer data of FairDeal must be converted into GoodsKart customer data format.

### Key issues

GoodsKart customer data has more fields than in FairDeal customer data. Hence FairDeal data needs to be split and stored in GoodsKart Customer Object-Oriented Data Structure.

### Considerations

The system should convert the data at run time.

### Data volume

- NA

### Additional information

- NA

### Business benefits

GoodsKart can eventually restructure the IT systems of FairDeal and reduce its cost by 20-30%

### Approach to Solve

You must use the fundamentals of Python taught in Module 3.

1. Read FairDealCustomerData.csv

2. Name field contains full name – use a regular expression to separate title, first name, and last name
3. Store the data in Customer Class
4. Create Custom Exception – CustomerNotAllowedException
5. Pass a customer to function "create order" and throw CustomerNotAllowedException in case of blacklisted value is 1

#### Enhancements for code

You can try these enhancements in code.

1. Change function **createOrder** to take **productname** and product code as **input**
2. Create **Class Order**  
Return object of type **Order** in case customer is eligible.

edureka!