

Programming and Problem Solving for Data Analytics

Week 2 Exercises

Learning Objectives

1. Create scripts or programs that display outputs on screen.
2. Create scripts or programs that assign values to variables and displays outputs.
3. Describe the use of different arithmetic operators.
4. Create programs that can retrieve user input, assign such input to variables, and perform arithmetic, logical, assignment, or output operations.
5. Create programs that use selection statements, including nested selection.
6. Create programs that perform input validation
7. Create and execute programs that use loops, nested loops, and a mixture of loops, branching, and other flow control mechanisms.
8. Perform basic debugging operations.

Tasks

- 1) **Display**: Print a simple string, e.g., Hello World!
- 2) **Display and Variables**: Print a greeting of the form "Hello, Michael!". "Michael" should be retrieved from a variable.
- 3) **Display and Variables**: Write a python program that displays the following verse from the poem, The Scientific Method:

The scientific method is the way to go
When you absolutely definitely need to know
How things happen in our work everyday
Scientists helps show all of us the way.

- a) Use only the print statement(s)
 - b) Use variable(s) to store the verse then print the result.
- 4) **Type-based Arithmetic**: What is the output of the following mathematical expressions?
 - $10+2*6/4$
 - $10+2*6//4$
 - `type(10+2*6/4)`
 - `type(10+2*6//4)`

Why?

- 5) **Debugging**: What happens when you execute the following script?

```
x=input("Enter age:")  
x**2
```
- 6) **Interactive input**: Read name, age at next birthday, course, nationality, and display in the format:

Name: Greg Hazimana

Age Next Birthday: 29

Course: MSEAI

Nationality: Rwanda

- 7) **Arithmetic, Conditionals, Two-Way Branching:** Write a program to ask for an integer input and determine if the number is even or odd. An even number is divisible by 2. The program should display an appropriate message.
- 8) **Input validation and Multi-way Branching:** Write a program that reads a mark in the range 0-100 (inclusive). Any values outside the range must be rejected with an appropriate message and the program exits after displaying the message. For a valid input, print out the grade as per the following grading system.
 - 92-100(inclusive): "A"
 - 82-91(inclusive): "B"
 - 70-81(inclusive): "C"
 - 0-69(inclusive): "Fail"
- 9) **Nested Loops, Input Validation, Multi-way Selection:** Modify the program in 8) above so that it asks the user for the number of students then reads the mark and displays the grade for each student. Modify the mark validation code such that the user will be requested to enter the right mark again and again until the input is valid.
- 10) **Arithmetic:** Write a program that calculates exponentials of the form x^y . Your program asks the user for a base x and an exponent y, and calculates x^y .
 - a) Use ** operator
 - b) Use pow function
- 11) **Looping, logical operators, arithmetic:** Write a program which will find all such numbers which are divisible by 13 but are not a multiple of 3, between 771 and 1542 (both included).
 - a) Use a while loop
 - b) Use a for loop
- 12) **Looping, logical operators, arithmetic:** Modify the code in 11(a) or (b) such that no even numbers are printed. Hint: Use continue statement.
- 13) **Looping, logical operators, arithmetic:** Modify the code in 11(a) or (b) such that as soon as a value divisible by 7 is encountered, the value is printed with the message "<value> divisible by 7, loop terminates!". Hint: Use break statement.