

## Week 3

Write and run your programs with IDLE editor. Submit finished programs to CodeGrade. Note that some tasks have several steps (A, B, C, ...) in CodeGrade.

**IMPORTANT:** End each input-command string with a newline symbol `\n`. For example:

```
variable = input("Some text:\n")
```

**Task 1:** Create a program that checks if a given year (`int`) is a **leap year**. Leap years are divisible by 4. If a year is divisible by 100, but not divisible by 400, leap year is skipped.

### Example run 1:

```
Enter a year:
2023
2023 is not a leap year.
```

### Example run 2:

```
Enter a year:
2000
2000 is a leap year.
```

### Example run 3:

```
Enter a year:
2100
2100 is not a leap year.
```

**Task 2:** Write a program that works according to the following selection structure:

1. Ask if the user wants to stop running the program and if the user enters the letter 'y' or 'Y', stop the program execution with the output "Bye!" (see Example run 1)
2. Otherwise, ask the user for name and password.
3. If the name is "Mark" and the password is "drowssap", print the text: "User identified" and end the program execution (see Example run 2)
4. Otherwise, print the length of the name and the information that name or the password was not correct: "You entered an invalid login name or password." (see Example run 3)

### Example run 1:

```
Do you want to stop the execution of the program (y/Y):
Y
Bye!
```

### Example run 2:

```
Do you want to stop the execution of the program (y/Y):
N
Enter username:
Mark
Enter password:
drowssap
User recognized.
```

**Example run 3:**

```
Do you want to stop the execution of the program (y/Y):  
N  
Enter username:  
Mark  
Enter password:  
supersecret  
You entered an invalid login name or password.
```

**Task 3:** Write a simple calculator which operates as described here:

1. The program starts by asking the user for two integers, which are stored in variables having the type `int`.
2. Program asks whether the user wants to 1) add, 2) subtract, 3) multiply or 4) divide the numbers.
3. Next, the calculator will perform the desired operation and output the result.
4. If the user tries to divide by zero, then the calculator gives an error message "Zero cannot be used as a divisor." The result of the division operation must be rounded to two decimal places.

**Note:** If the user does not select 1-4, the program prints the text "The operation was not recognized."

**Example run 1:**

```
Enter the first number:  
324  
Enter the second number:  
52  
The calculator can perform the following operations:  
1) Add  
2) Subtract  
3) Multiply  
4) Divide  
The numbers you entered are 324 and 52  
Select the operation (1-4):  
1  
Selection 1: 324 + 52 = 376
```

**Example run 2:**

```
Enter the first number:
737
Enter the second number:
199
The calculator can perform the following operations:
1) Add
2) Subtract
3) Multiply
4) Divide
The numbers you entered are 737 and 199
Select the operation (1-4):
3
Selection 3: 737 * 199 = 146663
```

**Example run 3:**

```
Enter the first number:
36
Enter the second number:
0
The calculator can perform the following operations:
1) Add
2) Subtract
3) Multiply
4) Divide
The numbers you entered are 36 and 0
Select the operation (1-4):
4
Error: Zero cannot be used as a divisor.
```

**Example run 4:**

```
Enter the first number:
24
Enter the second number:
13
The calculator can perform the following operations:
1) Add
2) Subtract
3) Multiply
4) Divide
The numbers you entered are 24 and 13
Select the operation (1-4):
7
The operation was not recognized.
```

**Task 4:** Write a program that makes testings on strings.

1. First the program asks for two words. Then, it compares these words according to the string comparison operation and print the strings according to the order (see example runs). If the words are the same, print: The words are the same.
2. After this, your program should test whether the given words contain the character 'z'. If the letter 'z' is found in either of the word (or both!), the program outputs: "The <word> contains 'z'." If the letter is not found, the program prints: "The letter 'z' was not found in either of the words."
3. Finally, your program should ask for a third string and determine whether it is a palindrome. Palindrome is a word that can be read in the same way from beginning to end as from end to beginning. If the given word is palindrome, print "The <word> is a palindrome." If not, print "The <word> is not a palindrome."

**Example 1:**

```
Enter word 1:
alpha
Enter word 2:
beta
'alpha' comes earlier in order than 'beta'.
The letter 'z' was not found in either of the words.
Enter a word to be tested:
rotator
'rotator' is a palindrome.
```

**Example 2:**

```
Enter word 1:
jazz
Enter word 2:
lazy
'jazz' comes earlier in order than 'lazy'.
Letter 'z' is found in word 'jazz'.
Letter 'z' is found in word 'lazy'.
Enter a word to be tested:
python
'python' is not a palindrome.
```

**Task 5:** Write a program that takes points (float) as input and then prints the corresponding grade based on the following grading scale:

- 90 – 100    5
- 80 – 89    4
- 70 – 79    3
- 60 – 69    2
- 50 – 59    1
- 0 – 49    0

**Example run 1:**

Enter your number of points:

93

Your grade is: 5

**Example run 2:**

Enter your number of points:

42.5

Your grade is: 0

**Example run 3:**

Enter your number of points:

72

Your grade is: 3