

# ENDG 233 – Programming with Data

**Variables, Arithmetic operators, Expressions**



Week 3: Sept. 20<sup>th</sup> – Sept 26<sup>th</sup>

# This week in glance

- Variables, Arithmetic operators, Expressions
  - Variables and assignments
  - Arithmetic expressions
  - Python expressions
  - Zylab examples
- In-lab Exercises with Zylab (activated at last 75 minutes of the last session in this week)
- Portfolio Assignment 1

# Tutorial 3.1 – House Real Estate Summary

- **Task:** Write a program with two inputs, current price and last month's price (both integers). Then, output a summary listing the price, the change since last month, and the estimated monthly mortgage computed as  $(\text{current\_price} * 0.051) / 12$ .
- **Note:** Output each floating-point value with two digits after the decimal point, which can be achieved as follows:  
`print(f'{your_value:.2f}')`

# Tutorial 3.1 – House Real Estate Summary

- **Step 1: Inputs**
  - What are the inputs?
  - How many inputs do I have?
  - Where do the inputs come from?
  - Type of the variable.
  - Choose the meaningful name for variables.
- `current_price = int(input())`
- `last_months_price = int(input())`

# Tutorial 3.1 – House Real Estate Summary

- Step 2: Math operations or Expression
- **What formula I need?**
  - Price change
  - Calculate the mortgage  $((\text{current\_price} * 0.051) / 12)$

Example:

$\text{Change\_price} = \text{current\_price} - \text{last\_months\_price}$

$\text{Monthly\_Mortgage} = (\text{current\_price} * 0.051) / 12$

# Tutorial 3.1 – House Real Estate Summary

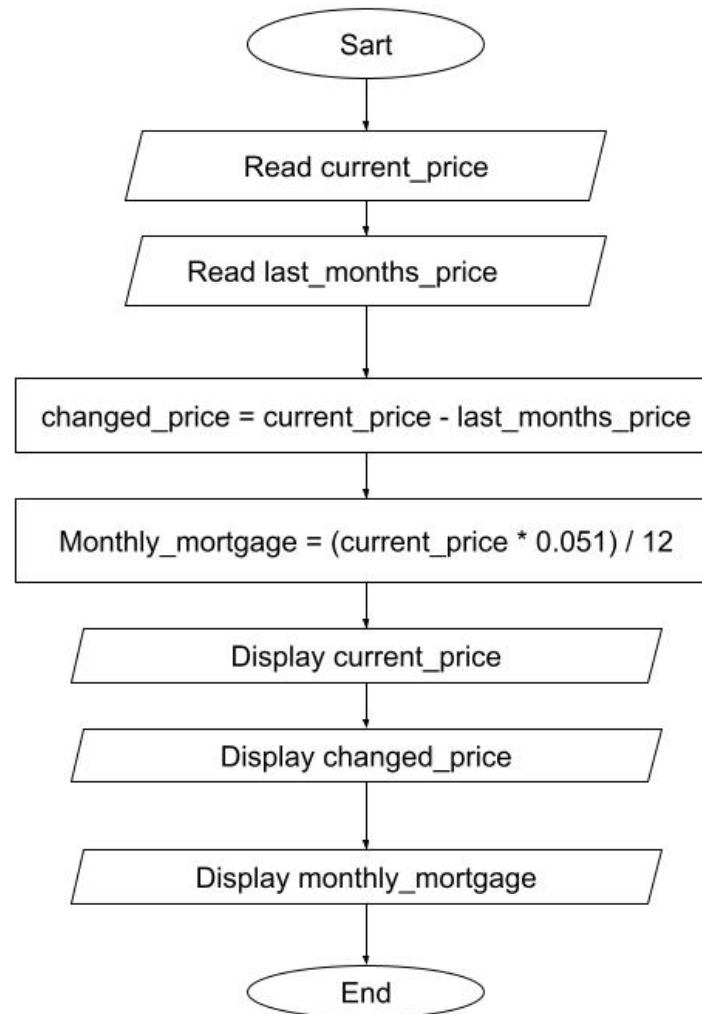
- Step 3: Outputs
- **What are the outputs?**
  - Listing the price,
  - The change since last month
  - The estimated monthly mortgage

```
print(f'This house is ${current_price}. The change is ${Change_price } since last month.')
```

```
print(f'The estimated monthly mortgage is ${Monthly_Mortgage :.2f}.')
```

# Tutorial 3.1 – House Real Estate Summary

## Flow Chart





# Tutorial 3.1 – House Real Estate Summary

## #Step 1: Inputs

```
current_price = int(input())  
last_months_price = int(input())
```

## #Step 2: Math operation (expressions)

```
Change_price = current_price - last_months_price  
Monthly_Mortgage=(current_price * 0.051) / 12
```

## #Step 3: Outputs

```
print(f'This house is ${current_price}. The change is ${current_price - last_months_price} since last month.')  
print(f'The estimated monthly mortgage is ${(current_price * 0.051) / 12 :.2f}.')
```



## Tutorial 3.2 – Volume and area of cylinder

- The volume and area of a cylinder are calculated as:  
Volume =  $\pi r^2 h$   
Area =  $2\pi r h + 2\pi r^2$
- **Task** : Given the radius and height of a cylinder as floating-point numbers, output the volume and area of the cylinder.

## Tutorial 3.2 – Volume and area of cylinder

- **Hint:** Use the constant pi from the math module as (math.pi) in your calculations.
- Output each floating-point value with one digit after the decimal point, which can be achieved as follows:  
`print(f'Volume: {yourValue:.1f} cubic inches')`

# Tutorial 3.2 – Volume and area of cylinder

```
import math
```

```
Radius_Cylinder = float(input())    # Read radius
```

```
Height_Cylinder = float(input())    # Read height
```

```
# calculate volume
```

```
Volume_Cylinder = math.pi * (Radius_Cylinder **2) * Height_Cylinder ;
```

```
# caculate area of cylinder
```

```
Area_Cylander = (2 * math.pi * Radius_Cylinder * Height_Cylinder ) + (2 * math.pi *  
(Radius_Cylinder **2))
```

```
print(f'Volume: {Volume_Cylinder :.1f} cubic inches')
```

```
# display volume
```

```
print(f'Surface area: {Area_Cylander :.1f} square inches')
```

```
# diplay surface area
```

## Tutorial 3.3 – Expression for Calories Burned

- The following equation estimates the average calories burned for a person when exercising, which is based on a scientific journal article ([source](#)):
- $$\text{Calories} = ( (\text{Age} \times 0.2757) + (\text{Weight} \times 0.03295) + (\text{Heart Rate} \times 1.0781) - 75.4991 ) \times \text{Time} / 8.368$$
- Task - Write a program using inputs age (years), weight (pounds), heart rate (beats per minute), and time (minutes), respectively. Output the average calories burned for a person.

## Tutorial 3.3 – Expression for Calories Burned

- **Note:** Output each floating-point value with two digits after the decimal point, which can be achieved as follows:  
`print(f'Calories: {calories:.2f} calories')`
- Input:  
49  
155  
148  
60
- Output:  
Calories: 736.21 calories

# Tutorial 3.3 – Expression for Calories Burned

## Step1:

```
age_years = int(input())  
weight_pounds = int(input())  
heart_bpm = int(input())  
time_minutes = int(input())
```

## Step 2:

```
calories = ((age_years * 0.2757) + (weight_pounds * 0.03295) +  
(heart_bpm * 1.0781) - 75.4991) * time_minutes / 8.368
```

## Step 3:

```
print(f'Calories: {calories:.2f} calories')
```

## Tutorial 3.4 – Measurement's converter

- Output each floating-point value with two digits after the decimal point, which can be achieved as follows:  
`print(f'{your_value:.2f}')`
- (1) Prompt the user for the number of cups of lemon juice, water, and agave nectar needed to make lemonade. Prompt the user to specify the number of servings the recipe yields. Output the ingredients and serving size. (Submit for 2 points).





## Tutorial 3.4 – Cooking Measurement's converter

- ***Note:** This zyLab outputs a newline after each user-input prompt. For convenience in the examples below, the user's input value is shown on the next line, but such values don't actually appear as output when the program runs.*

# Tutorial 3.4 – Cooking Measurement's converter

## Task:

**Subtask 1:** Prompt the user for the number of cups of lemon juice, water, and agave nectar needed to make lemonade. Prompt the user to specify the number of servings the recipe yields. Output the ingredients and serving size.

**Subtask 2:** Prompt the user to specify the desired number of servings. Adjust the amounts of each ingredient accordingly, and then output the ingredients and serving size.

**Subtask 3:** Convert the ingredient measurements from (2) to gallons. Output the ingredients and serving size. Note: There are 16 cups in a gallon.

# Tutorial 3.4 – Cooking Measurement's converter

## Subtask 1

```
lemon_juice_cups = float(input('Enter amount of lemon juice (in cups):\n'))

# FIXME (1): Finish reading other items into variables, then output the three
ingredients
water_cups = float(input('Enter amount of water (in cups):\n'))
sugar_cups = float(input('Enter amount of agave nectar (in cups):\n'))
serving_size = float(input('How many servings does this make?\n'))

print(f'\nLemonade ingredients - yields {serving_size:.2f} servings')
print(f'{lemon_juice_cups:.2f} cup(s) lemon juice')
print(f'{water_cups:.2f} cup(s) water')
print(f'{sugar_cups:.2f} cup(s) agave nectar')
```

# Tutorial 3.4 – Cooking Measurement's converter

## Subtask 2:

# FIXME (2): Prompt user for desired number of servings. Convert and output the ingredients

```
new_serving_size = float(input('\nHow many servings would you like to make?\n'))
```

```
div_num = new_serving_size / serving_size
```

```
print(f'\nLemonade ingredients - yields {serving_size * div_num:.2f} servings')
```

```
print(f'{lemon_juice_cups * div_num:.2f} cup(s) lemon juice')
```

```
print(f'{water_cups * div_num:.2f} cup(s) water')
```

```
print(f'{sugar_cups * div_num:.2f} cup(s) agave nectar')
```

# Tutorial 3.4 – Cooking Measurement's converter

## Subtask 3:

# FIXME (3): Convert and output the ingredients from (2) to gallons

cups\_to\_gallons = 1/16

```
print(f'\nLemonade ingredients - yields {serving_size * div_num:.2f} servings')
print(f'{lemon_juice_cups * div_num * cups_to_gallons:.2f} gallon(s) lemon
juice')
print(f'{water_cups * div_num * cups_to_gallons:.2f} gallon(s) water')
print(f'{sugar_cups * div_num * cups_to_gallons:.2f} gallon(s) agave nectar')
```

# Exercise 3.5 - Working with Variables and Expressions

This exercise is graded. You must submit your answer within 24 hours of your last active learning session in Week 3

- Ali is a mechatronics engineer working on a new type of high-performance vehicle. She needs a terminal-based program that will automatically compute the velocity and acceleration of her test vehicles. The available input is as follows:
- Starting position (in meters)  
Ending position (in meters)  
Starting time (in seconds)  
Ending time (in seconds)

# Exercise 3.5 - Working with Variables and Expressions

- **Task** : Write a program that accepts the four required float values (each on a newline with no user prompt) and calculates the required information. Your output must be rounded to one decimal place and must match the example output shown.
- You may assume that the ending values will always be larger than the starting values.



# Portfolio Assignment

- D2L>Portfolio Assignments > Assignment#1