Types and Variables

# Before Class

1. From the Python tutorial:

<https://www.w3schools.com/python/default.asp>

familiarize yourself with all topics, from the first section until the Java Booleans section.

* **Many Values to Multiple Variables**x, y, z = "Orange", "Banana", "Cherry"  
    
  fruits = ["apple", "banana", "cherry"]  
  x, y, z = fruits
* **Output Variables**x = "Python"  
  y = "is"  
  z = "awesome"  
  print(x, y, z)

x = "Python "  
y = "is "  
z = "awesome"  
print(x + y + z)

x = 5  
y = "John"  
print(x, y)

* **Global variables**   
  Create a variable outside of a function, and use it inside the function  
  If you use the global keyword, the variable belongs to the global scope
* **Data types**x = 5  
  print(type(x))
* **Numbers**x = 1    # int  
  y = 2.8  # float  
  z = 1j   # complex  
    
  Python does not have a random() function to make a random number, but Python has a built-in module called random that can be used to make random numbers
* **Python castings**x = int(1)   # x will be 1  
  y = int(2.8) # y will be 2  
  z = int("3") # z will be 3  
    
  x = float(1)     # x will be 1.0  
  y = float(2.8)   # y will be 2.8  
  z = float("3")   # z will be 3.0  
  w = float("4.2") # w will be 4.2x = str("s1") # x will be 's1'  
  y = str(2)    # y will be '2'  
  z = str(3.0)  # z will be '3.0'
* **Strings**You can assign a multiline string to a variable by using three quotes.  
    
  Get the character at position 1 (remember that the first character has the position 0):   
  a = "Hello, World!"  
  print(a[1])

Looping Through a String:  
for x in "banana":  
  print(x)  
  
String Length:  
a = "Hello, World!"  
print(len(a))  
  
Check string:   
txt = "The best things in life are free!"  
print("free" in txt)

Print only if "free" is present:  
txt = "The best things in life are free!"  
if "free" in txt:  
  print("Yes, 'free' is present.")

Check if "expensive" is NOT present in the following text:  
txt = "The best things in life are free!"  
print("expensive" not in txt)

print only if "expensive" is NOT present:  
txt = "The best things in life are free!"  
if "expensive" not in txt:  
  print("No, 'expensive' is NOT present.")

Get the characters from position 2 to position 5 (not included):  
b = "Hello, World!"  
print(b[2:5])

Get the characters from the start to position 5 (not included):  
b = "Hello, World!"  
print(b[:5])

Get the characters from position 2, and all the way to the end:  
b = "Hello, World!"  
print(b[2:])

Get the characters: From: "o" in "World!" (position -5) To, but not included: "d" in "World!" (position -2):  
b = "Hello, World!"  
print(b[-5:-2])

a = "Hello, World!"  
print(a.upper())

a = "Hello, World!"  
print(a.lower())

The strip() method removes any whitespace from the beginning or the end:  
a = " Hello, World! "  
print(a.strip()) # returns "Hello, World!"

a = "Hello, World!"  
print(a.replace("H", "J"))

a = "Hello, World!"  
print(a.split(",")) # returns ['Hello', ' World!']

a = "Hello"  
b = "World"  
c = a + b  
print(c)

a = "Hello"  
b = "World"  
c = a + " " + b  
print(c)

Create an f-string:  
age = 36  
txt = f"My name is John, I am {age}"  
print(txt)

Display the price with 2 decimals:  
price = 59  
txt = f"The price is {price:.2f} dollars"  
print(txt)

Perform a math operation in the placeholder, and return the result:  
txt = f"The price is {20 \* 59} dollars"  
print(txt)

The escape character allows you to use double quotes when you normally would not be allowed:  
txt = "We are the so-called \"Vikings\" from the north."

Escape Characters  
<https://www.w3schools.com/python/python_strings_escape.asp>

String Methods

<https://www.w3schools.com/python/python_strings_methods.asp>

1. Explain the concepts
   1. **data type**  
      Variables can store data of different types, and different types can do different things.
   2. **Variable**  
      Variables are containers for storing data values
   3. **Operator**  
      Variables are containers for storing data values
2. Check what data types and operators are available in Python.  
   <https://www.w3schools.com/python/python_operators.asp>  
   https://www.w3schools.com/python/python\_datatypes.asp
3. From the course textbook, available in your local repository, read chapters 1 and 2.

**CHAPTER 1**

*Debugowanie* to proces znajdowania w kodzie przyczyny błędu.

**Ćwiczenie 1.** Jaką rolę pełni w komputerze pamięć pomocnicza  
Przechowuje informacje przez dłuższy czas, nawet po wyłączeniu sprzętu.

**Ćwiczenie 2.** Czym jest program?  
to sekwencja instrukcji Pythona, która została stworzona po to, by coś zrobić.

**Ćwiczenie 3.** Jaka jest różnica między kompilatorem a interpreterem?  
Interpreter odczytuje kod źródłowy programu w postaci napisanej przez programistę, analizuje kod źródłowy i w locie interpretuje instrukcje.  
kompilator najpierw musi otrzymać w pliku cały program, potem uruchamia proces tłumaczenia kodu źródłowego wysokiego poziomu na język maszynowy, a następnie kompilator umieszcza w pliku

**Ćwiczenie 4.** Która z poniższych opcji zawiera “kod maszynowy”  
a**. Interpreter Pythona**. b. Klawiatura. c. Kod źródłowy Pythona. d. Dokument edytora tekstu

**Ćwiczenie 5.** Co jest nie tak w poniższym kodzie?  
>>> primt 'Witaj świecie!'

File "<stdin>", line 1  
 primt 'Witaj świecie!'  
 ^

SyntaxError: invalid syntax  
 >>>

**Ćwiczenie 6.** Gdzie w komputerze będzie przechowywana zmienna x po wykonaniu poniższej linijki kodu Pythona?  
x = 123  
**w pamięci glownej**

**Ćwiczenie 7**. Co wyświetli poniższy program?  
x = 43

x = x + 1

print ( x )

**44**

**Ćwiczenie 8.** Wyjaśnij każdy z poniższych elementów na przykładzie możliwości człowieka:  
a. procesor – mozg   
b. pamięć glowna – pamięć krotkotrwala   
c. pamięć pomocnicza – pamięć dlugotrwala   
d. urządzenia wejścia – zmysły   
e. urządzenia wyjścia – zmysły / mowa

**Ćwiczenie 9.** W jaki sposób naprawisz “Syntax Error” (błąd składniowy)?  
znaleźć linijke, sprawdzić typowe bledy składniowe

**CHAPTER 2**  
**Ćwiczenie 2.** Napisz program, który wykorzystuje funkcję input() do poproszenia użytkownika o jego imię, a następnie przywita go, używając jego imienia.  
imie = input('Podaj swoje imie: ')

print(f'Witaj {imie}!')

**Ćwiczenie 3.** Napisz program, który wyświetli użytkownikowi pytanie o liczbę godzin pracy i stawkę za godzinę w celu obliczenia wynagrodzenia  
godziny = float(input('Podaj liczbe godzin: '))

stawka = float(input('Podaj stawke godzinowa: '))

wynagrodzenie = godziny \* stawka

print(f'Wynagrodzenie: {wynagrodzenie}')

**Ćwiczenie 4.** Załóżmy, że wykonujemy następujące instrukcje przypisania:  
width = 17 height = 12.0  
Dla każdego z poniższych wyrażeń podaj wartość wyrażenia i oraz typ (wartości wyrażenia).

* 1. wartosc = width//2  
     print(f'wartosc = {wartosc}, typ = {type(wartosc)}')
  2. wartosc = width/2.0

print(f'wartosc = {wartosc}, typ = {type(wartosc)}')

* 1. wartosc = height/3

print(f'wartosc = {wartosc}, typ = {type(wartosc)}')

* 1. wartosc = 1 + 2 \* 5

print(f'wartosc = {wartosc}, typ = {type(wartosc)}')

**Ćwiczenie 5.** Napisz program, który prosi użytkownika o podanie temperatury w skali Celsjusza, przelicza ją na skalę Fahrenheita i wyświetla przeliczoną temperaturę.  
celsjusz = int(input('Podaj temperature: '))

far = (2 \* celsjusz) + 30

print(f'Temperatura w F: {far}')

1. From the chapter 2, complete all exercises in the Exercises section
   1. Use VSCode to create and run programs
   2. In VSCode, open the folder 02-TypesAndVariables available in your local repository
   3. Save every created program in that folder
   4. After completing all exercises, save changes to your local repository and then to your remote repository on Github. You can use the following Git commands:  
      git add .  
      git commit -m "My first Python programs from textbook"  
      git push  
      <https://youtu.be/eL_0Ok_Gkas>
   5. Check if your remote Github repository contains all the programs you created.

# During Class

## Types, operators and expressions

1. Consider what data type the following values represent. Then check your answers (run Python in interactive mode). You can use the available type(value) function.
   1. 50 int
   2. 149.17 float
   3. 4 \* 7 int
   4. 4.0 \* 7 float
   5. "Krakow University of Economics" str
   6. True bool  
      g. 2 > 5 bool

print(type(50))

print(type(149.17))

print(type(4\*7))

print(type(1.0\*7))

print(type('Krakow University of Economics'))

print(type(True))

print(type(2>5))

1. Count how many operators and arguments are included in each expression. Then, calculate values of the expressions. First, try to calculate every expression without using a computer. Then, check results on your computer (run Python in interactive mode).
   1. 3 – 2 + 1
   2. 5 + 10 \* 5
   3. 4 + 4 / 2 \*\* 2
   4. 4 % 3 % 2 % 1
   5. 1 + 2 % 3 \*\* 4 \* 5
   6. True != False

## Variables

1. Write a program that calculates the sum of two integer numbers contained in variables. Some result:

number1 = 71  
number2 = 14  
result = number1 + number2  
print('Number 1: ', number1)  
print('Number 2: ', number2)  
print('The result of summation: ', result)

number1 = int(input('Podaj pierwsza liczbe: '))

number2 = int(input('Podaj druga liczbe: '))

print(f'number1 = {number1} \nnumber2 = {number2}')

result = number1 + number2

print(f'The result of summation: {result}')

1. Natural values 15, 1, 8, 6, 31 have been assigned to variables named n1, n2, n3, n4, n5. Write a program that calculates and displays:
   1. sum of all variables
   2. sum of squared variables
   3. quotient of the variable three and five  
      d. message (True / False) indicating if the third variable is equal to the fourth variable

n1 = 15

n2 = 1

n3 = 8

n4 = 6

n5 = 31

numa = n1 + n2 + n3 + n4 + n5

numb = pow(n1, 2) + pow(n2, 2) + pow(n3, 2) + pow(n4, 2) + pow(n5, 2)

numc = n3 / n5

numd = n3 == n4

print(f'sum = {numa}, sum of squared variables = {numb}, quotient of the variable three and five = {numc}, message = {numd}')

1. One of the basic functionalities of a computer program is displaying results on the computer screen. To display the results in a readable form, string (text) formatting, available in programming languages, is often used. In Python, it is called f-strings.

<https://www.pythontutorial.net/python-basics/python-f-strings/>

<https://docs.python.org/3/tutorial/inputoutput.html>

Here is an example of displaying any text along with the value of a variable using f-strings:

name = "Adam"  
print(f"I am {name}, a student of this university")

Let three variables: name, age and height contain your personal data. Write a program that, using f-strings, displays the following sentence:

My name is ..., I am ... years old, and my height is ... cm. In 6 years I will be ... years old.

name = 'Julia'

age = 24

height = 173

age6 = 30

print(f'Ma name is {name}, I am {age} years old, and my height is {height} cm. In 6 years I will be {age6} years old.')

1. Write a program that calculates and displays the income of a family per person. To display the results in a readable form, use f-strings. Sample result:

father\_income = 5450  
mother\_income = 4920  
number\_of\_people = 5  
Total income: …  
Income per person: …

father\_income = 5450

mother\_income = 4920

number\_of\_people = 5

total = father\_income + mother\_income

per\_person = total / number\_of\_people

print(f'Total income: {total} \nIncome per person: {per\_person}')

## Data input

1. Write a program that reads your name and surname from the keyboard. Store this data in two separate variables. Then, display your first and last name separated by a single space. Sample result:

first\_name = input('Enter your first name: ')  
last\_name = input('Enter your last surname: ')  
full\_name = first\_name + ' ' + last\_name  
print(f'Your fullname is {full\_name}')

first\_name = input('Enter your first name: ')

last\_name = input('Enter your last name: ')

full\_name = first\_name + ' ' + last\_name

print(f'Your full name is {full\_name}')

1. Write a program that calculates the surface area of a cube. Read the length of the side of the cube from the keyboard. Take into account that the input() function returns a string, not a number. To convert a string to a number, use the int() function. Sample result:

Enter cube side: 6  
The surface area of a cube with side 6 is 216

side = int(input('Enter cube side: '))

surface = pow(side, 3)

print(f'The surface area of a cube with side {side} is {surface}')

## Algorithms

1. The radius of the circle has the value 5, stored in a variable. Write a program that calculates the area and circumference of the circle. Use the algorithm below.

#####  
# Calculation of circle area and circumference   
##  
  
# determine radius and PI  
# calculate area   
# calculate circumference   
# display results

radius = int(input('Determine radius:'))

pi = 3.14

area = pi \* pow(radius, 2)

circ = 2 \* pi \* radius

print(f'area = {area}\ncircumference = {circ}')

1. Write a program that reads temperature in degrees Celsius from the keyboard. Then, the program calculates and displays the temperature in Kelvin and Fahrenheit. Use comments to briefly describe the program's algorithm.

number1 = int(input('Podaj pierwsza liczbe: '))

number2 = int(input('Podaj druga liczbe: '))

print(f'number1 = {number1} \nnumber2 = {number2}')

result = number1 + number2

print(f'The result of summation: {result}')

# After Class

1. Evaluate the following expressions. First, try to do it without using a computer. Then, check the results on your computer (run Python in interactive mode).
   1. The product of the numbers 15 and 38
   2. The product of the sum of pairs of numbers 3 and 4, and 5 and 9
   3. Integer part of dividing the numbers 7 and 2
   4. The remainder of the division of 48 and 5
   5. Arithmetic mean of the numbers 8, 7, 4, 2
   6. 210
   7. Square root of 49 (do not use a function)
   8. 25% of 80

numa = 15 \* 38

numb = (3+4) \* (5\*9)

numc = int(7/2)

numd = int(48 % 5)

nume = (8+7+4+2)/4

numf = pow(2, 10)

numg = 49\*\*(1/2)

numh = 80 \* (1/4)

1. Calculate values of the following expressions. First, try to do it without using a computer. Then, check the results on your computer (run Python in interactive mode).
   1. 5 + 10 \* 5
   2. 3 – 2 + 1
   3. 2 + - 3
   4. 2 \*\* 8
   5. 4 + 4 / 2 \*\* 2
   6. 4 % 3 % 2 % 1
   7. 1 + 2 % 3 \*\* 4 \* 5
   8. True != False
   9. 2 <= 3 or False
   10. not True or not False and not True
   11. 2 < 3 and 4 < 5 or not 6 < 7
   12. 2 % 3 < 4 / 5 and 6 + 7 < 8 or not 9 + 10 == 19
   13. 0x11 + 0b11 + 11  
         
       55

2

-1

256

5.0

0

11

True

True

False

True

False

31

1. A variable x has a value of 7 and a variable y has a value of 34. Write a program that swaps variable values (x should be 34 and y should be 7). You can use one additional, auxiliary variable. Sample result:

Value of x: 7  
Value of y: 34  
Value of x after swapping: 34  
Value of y after swapping: 7

x = 7

y = 34

print(f'Value of x: {x}\nValue of y: {y}')

temp = x

x = y

y = temp

print(f'Value of x after swapping: {x}\nValue of y after swapping: {y}')

1. The length of the side of the cube is contained in the variable a. Write a program that calculates and displays the volume and surface area of the cube. Sample result:

Cube side: 4  
Cube volume: 64  
Cube surface area: 96

side = int(input('Cube side: '))

volume = pow(side, 3)

surface = pow(side, 2) \* 6

print(f'Cube volume: {volume}\nCube surface area: {surface}')

1. Variables a and b contain two integer numbers. Write a program that calculates and displays the result of their division, rounded down to the nearest whole number. Display also the remainder of the division. Sample result:

Number one: 17  
Number two: 5  
Division result: 3  
Remainder: 2

one = int(input('Number one: '))

two = int(input('Number two: '))

division = int(one/two)

remainder = one % two

print(f'Division result: {division}\nRemainder: {remainder}')

1. A variable contains your height in cm. Write a program that displays your height both in cm and in feet and inches. Sample result:

I am 170cm tall, i.e. 5 feet and 7 inches

height = int(input('Enter height in cm: '))

feet = int(height/30.48)

remaining = height - (feet \* 30.48)

inches = int(remaining \* 0.3937)

1. The variables a and b contain two integers, 3 and 5, respectively. Write a program that displays the following expression containing the values of these variables and the value of the expression.

3 – 5 = -2

print(f'{a} - {b} = {a-b}')

1. The length of the sides of the triangle is a, b and c. Write a program that calculates the area of the triangle (using the Heron formula) and the triangle circumference. Read the values of the triangle sides from the keyboard. Sample result:

Enter a: …  
Enter b: …  
Enter c: …  
Triangle area: …  
Triangle circumference: …

s = (a+b+c)/2

formula = s \* (s-a) \* (s-b) \* (s-c)

area = int(pow(formula, (1/2)))

circumference = a + b + c

print(f'Triangle area: {area}\nTriangle circumference: {circumference}')

1. Vehicle registration numbers in Kraków start with the letters KR or KK. Write a program that checks whether the vehicle registration number entered from the keyboard means a vehicle from Krakow. Display True whether a car is from Kraków or False otherwise. Sample result:

Enter vehicle registration number: KR45091  
Car from Kraków: True

a = input('Enter vehicle registration number: ')

if a[0] == 'K' and (a[1] == 'R' or a[1] == 'K'):

    print(f'Car from Krakow: True')

else:

    print(f'Car from Krakow: False')

1. People up to and including 26 years of age are exempt from paying taxes in Poland. Write a program that, based on the person's age entered from the keyboard, displays True if the person is exempt from paying taxes and displays False otherwise. Sample result:

Enter age: 23  
Exemption from paying taxes: True

age = int(input('Enter age: '))

print(f'Exemption from paying taxes: {age<=26}')

1. Write a program that checks whether the number entered from the keyboard is even. Display True when the number is even and False when the number is odd. Sample result:

Enter number: 34  
Number is even: True

number = int(input('Enter number: '))

print(f'Number is even: {number%2 == 0}')

1. Write a program that checks whether the number entered from the keyboard is between -10 and 10. Display True if the number is in the given range, and False otherwise. Sample result:

Enter number: 17  
Number in the range <-10,10>: True

number = int(input('Enter number: '))

print(f'Number in the range <-10,10>: {-10<number and number<10}')

1. Correct body weight has a significant impact on health. Write a program that calculates the Body Mass Index (BMI) based on your height in cm and weight in kg. A user enters data from the keyboard. Find the formula on the Internet for calculating the BMI. Then, using your program, check that you have the correct weight. Display the calculated BMI and display True if the weight is within the valid range, or display False otherwise. Sample result:

Enter your height in cm: ...  
Enter your weight in kg: ...  
Your BMI index: ...  
Correct weight: True

height = int(input('Enter height in cm: '))

weight = int(input('Enter wight in kg: '))

BMI = weight / pow((height)/100, 2)

print(BMI)

print(f'Correct weight: {18.5 < BMI and 24.9 > BMI}')

1. Each programming language provides a set of functions that you can use in your programs. One of them is a function that returns a random number. Write a program that displays results of three dice rolls and the sum of dice rolled. Apply a random number generator:

<https://docs.python.org/3/library/random.html>

import random

list = [1, 2, 3, 4, 5, 6]

a = int(random.choice(list))

b = int(random.choice(list))

c = int(random.choice(list))

sum = a + b + c

print(f'{a}\n{b}\n{c}\nsum = {sum}')

1. In many games, the numbers one and six on dice have special meaning. Write a program that displays the number of dice rolled and the value True if the number rolled is 1 or 6. Sample result:

Dice rolled: 4  
Special number: False

import random

list = [1, 2, 3, 4, 5, 6]

a = int(random.choice(list))

print(f'Dice rolled: {a}\nSpecial number: {a==1 or a==6}')

1. Write a program that enables a user to challenge a computer. The computer throws dice. Then, the user then tries to guess the number from dice by entering a number from 1 to 6 from the keyboard. If the user has guessed the number from the dice, the computer displays True. Otherwise, the computer displays False.

import random

list = [1, 2, 3, 4, 5, 6]

a = int(random.choice(list))

b = int(input('Guess the number: '))

print(f'Guessed number: {a==b}')

1. 23% VAT was paid from an amount. Write a program that reads an amount from the keyboard. Then, it calculates and displays both the amount and its VAT. Apply formatting with two decimal places. Sample result:

Amount : 15.84  
VAT 23% : 3.64

amount = float(input('Amount: '))

vat = (23 \* amount) / 100

rounded\_vat = round(vat, 2)

print(f'VAT 23%: {rounded\_vat}')

1. The password is valid if it is at least 8 characters long. Write a program that checks whether the password read from the keyboard is correct. Sample result:

Enter password: qwertyXX  
Password is valid: True

a = input('Enter password: ')

print(f'Password is valid: {len(a)>=8}')

1. The speed of vehicles on a highway in Poland must be between 40 and 140 km/h. Write a program that checks whether the vehicle speed entered from the keyboard is correct. Sample result:

Enter vehicle speed: 158  
Speed is valid: False

speed = int(input('Enter vehicle speed: '))

print(f'Speed is valid: {speed >= 40 and speed <= 140}')

1. A tree may be cut down if its diameter is at least 50 cm. Write a program that, based on the circumference of the tree entered from the keyboard, calculates and displays the value True if the tree can be cut down, or display the value False otherwise. Sample result:

Enter tree circumference: …  
Tree can be cut down: …

circ = int(input('Enter tree circumference: '))

diameter = circ / 3.14

print(f'Tree can be cut down: {diameter >= 50}')

1. A bank buys and sells Euro. Write a program that, based on the Euro buying and selling rates entered from the keyboard, calculates the difference between the buying and selling rates (spread). Display result with 4 decimal places. Sample result:

Bank buys EUR: 4.5940  
Bank sells EUR: 4.6250  
Spread: 0.0310

buy = float(input('Bank buys EUR: '))

sell = float(input('Bank sells EUR: '))

spread = sell - buy

print(f'Spread: {round(spread, 4)}')

1. In Python, to read a range of characters from the string, a slicing method can be used.

<https://www.w3schools.com/python/python_strings_slicing.asp>

The employee file contains the employee's data in a descriptive form. Write a program in which the personal\_data variable contains employee data:

"Mr. John May, born on 1998-02-16"

Display the employee's name, surname, initials and date of birth on separate lines. Sample result:

Description: Mr. John May, born on 1998-02-16  
Name: John  
Surname: May  
Initials: JM  
Born: 1998-02-16

personal\_data = 'Mr. John May, born on 1998-02-16'

print(f'Description: {personal\_data}')

print(f'Name: {personal\_data[4:8]}')

print(f'Surname: {personal\_data[9:12]}')

print(f'Initials: {personal\_data[4]}{personal\_data[9]}')

print(f'Born: {personal\_data[22:]}')

1. To improve readability, telephone numbers are often presented with a dash or space separating some digits. Write a program that displays a 9-digit telephone number entered from the keyboard as three groups of 3 digits each, separated by a dash character. Sample result:

Enter phone number: 543097329  
Phone number: 543-097-329

number = input('Enter phone number: ')

print(f'Phone number: {number[:3]}-{number[3:6]}-{number[6:]}')

1. The price of the product on the price tag is given before and after the discount is applied. Write a program that allows you to enter the product price and discount. Display the product price, discount, discounted product price, and the difference between the product price before and after the discount. Display all prices with two decimal places. Sample result:

Enter price: 24.72  
Enter discount %: 15  
Price with discount: 21.01  
Reduction: 3.71

price = float(input('Enter price: '))

discount = float(input('Enter discount %: '))

with\_discount = price - (discount \* price) / 100

reduction = (discount \* price) / 100

print(f'Price with discount: {round(with\_discount, 2)}')

print(f'Reduction: {round(reduction, 2)}')

1. The credit card number consists of 16 digits. Write a program that allows you to enter a credit card number from the keyboard. Display the credit card number in groups of 4 digits, separating the groups with a space character. Sample result:

Enter credit card number: 5020312109004442  
Card number: 5020 3121 0900 4442

number = input('Enter credit card number: ')

print(f'Card number: {number[:4]} {number[4:8]} {number[8:12]} {number[12:]}')

1. The binary numeral system is a positional numeral system that requires only two digits to write numbers: 0 and 1. The hexadecimal numeral system is a positional numeral system that uses sixteen distinct symbols, most often the symbols "0"–"9" to represent values 0 to 9, and "A"–"F" (or alternatively "a"–"f") to represent values from ten to fifteen. Both are widely used in mathematics, computer science and digital electronics. Write a program that reads an integer number from the keyboard and displays that value as a binary and hexadecimal number. To convert a decimal number to binary or hexadecimal value, use the available Python functions. Sample result:

Enter number: 125  
Binary number: 0b1111101  
Hexadecimal number: 0x7d

dec = int(input('Enter number: '))

print(f'Binary number: {bin(dec)}')

print(f'Hexadecimal number: {hex(dec)}')

1. Write a program that allows you to enter a binary, four-digit number. Convert the entered number from binary to decimal value. Do not use available Python functions. Sample result:

Enter binary number: 0110  
Binary number in decimal notation: 6

binary = input('Enter a binary number: ')

if binary[0] == '1':

    a = 8

else:

    a = 0

if binary[1] == '1':

    b = 4

else:

    b = 0

if binary[2] == '1':

    c = 2

else:

    c = 0

if binary[3] == '1':

    d = 1

else:

    d = 0

decimal = a + b + c + d

print(f'Binary number in decimal notation: {decimal}')

1. In computer science, every written (graphic) character of human language (letters, digits, symbols, etc.) is encoded by assigning it a number. This allows characters to be stored, transmitted, and transformed using digital computers. Write a program that displays a numerical representation of each letter of your name. To convert a character to its numerical representation, use the Python ord() function. Sample result:

Name: John  
J(74) o(111) h(104) n(110)

name = input('Name: ')

for char in name:

    print(f"'{char}': {ord(char)}")