Chapter 1 notes

Python has two modes – interactive and script

Interactive mode – executes each line when you enter it. Interpreted means that each line in converted and is run

>>> print ('hello')

Hello

Syntax – rules of the language

Python is case sensitive. If you make a syntax mistake, you will get a syntax error

Psuedocode – write out what you want to do – don’t worry about syntax

Write a program to calculate the price of an item including the tax. We will use 6% sales tax

1. Enter the price
2. Enter the tax percent
3. Calculate the tax by taking the price times the tax percent
4. Add the price and tax together
5. Print the total

Variables in Python must be on the left side of an =sign

Variables are case sensitive. Price is not the same as price

Correct: price = 100

To be able to reuse your code – script code

Saves your commands and executes all of them

File >New File then enter the commands. We can save and run all of the commands.

Hardware

CPU – Central Processing Unit – main brain of the pc

Programs must be copied from the hard drive into RAM

RAM is volatile – if you lose power, RAM is gone

DATA BUS – binary digits transferred at the same time.

* Front Door – how much data can get in and out

Machine language – off or on, we use binary code, 0 and 1

We use high level language that is translated in binary for computer to run

Python is interpreted – translated and runs line by line

Compiled programs – the entire program is translated into binary, and then it runs

Base 10 place value – 10 numbers – 0 through 9

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | 10 squared | 10 ot the first power | 10 to the 0 power |
|  |  |  |  | 1 |
|  |  |  | 1 | 0 |
|  |  | 1 | 0 | 0 |

Base 2 is used for programs because computer circuits can be on or off. 0 is off and 1 is on

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2^7  128 | 2^6  64 | 2^5  32 | 2^4  16 | 2 cubed  8 | 2 squared  4 | 2 to the first power | 2 to the 0 power |
|  |  |  |  |  |  |  | 1 |
|  |  |  |  |  |  | 1 | 0 |
|  |  |  |  |  | 1 | 0 | 0 |
|  |  |  |  | 1 | 0 | 0 | 0 |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

Binary to base 10

|  |  |
| --- | --- |
| Binary | Base 10 |
| 1 | 1 |
| 10 | 2 |
| 11 | 3 |
| 100 | 4 |
| 101 | 5 |
| 111 | 6 |
| 11111111 |  |

Chapter 2

Do it right the first time!

Program development life cycle

Talk to the user – MIS – the bridge between the user and technology

Write pseudocode

Write actual code

Test the program

Correction( from user feedback)

Flow chart

Input

Processing

Output

GIGO -Garbage in, Garbage out

Function performs a task

Print will display the output

Arguments – provide information for a function

Comments – a note that you can put into a program – does not execute

# - beginning of a line # print (‘hello’)

Endline comment is the # after the code. Everything before executes, everything after is ignored

Print (‘helllo’) # this will print output

Age = 25 – this is ok

25 = age – this will not work

Varuables

No spaces in vatiable names

Can use Camel Case – StudentName, Student\_name

* Variables are case sensitive
* First character must be a letter or \_
* After the first, you can use lettes, numbers, and more characters

Garbage collection

* If you use age = 25 and later age = 40, if there is no variable attached, the 25 is deleted

DATA TYPES

String

Integer (no decimal)

Float (Number with decimal)

INPUT STATEMENTS

Instead of saying a = 25

A =input (‘enter a number: ’) – this becomes the value of a

Input statements, the data is a string, even if you type a number

A = 25 – this is an interger

A = 25.0 – this is a float

A =input (‘enter a number: ’) – this is a string

a = 25

b = 3

c = a\*b

print (c)

e=3.0

f = 4.0

g= e\*f

print (g)

h = input ('Enter a number: ')

i = input ('Enter a number: ')

j =h\*i

print (j)

will not work without int defining the input

h = int (input ('Enter a number (integer): '))

i = int (input ('Enter a number (integer): '))

j =h\*i

print (j)

print type (jj)

The type command will tell you the datatype of a variable

If you specify int in the input statement – you can enter an integer (You can’t enter a float)

If you specify a float in the input statement - you can enter a float or integer

k = float (input ('Enter a number: '))

l = float (input ('Enter a number: '))

m =k\*l

print (m)

print (type(m))

If the data type in the input statement is int – specify that the user must enter and integer

#division

B =7

C =2

D = b/c

Print (d)

b = 7

c = 2

d = b/c

print (d)

#if you sue ?? you will get the answer without the remainder

e = b//c

print (e)

#if you just want the remainder

f = b % c

# follows the order of operations from algebra

# \*\* exponents

g = 50\*\*50

a = .01

b = .7

c = a\*b

print (c)

#formatting to a set number of places

#modify yhe print function - arguments - seperate them with commas

# in the quotes - comma format - two fixed decimal places

print (format (c,'.2f'))

print (format (c,'.4f'))

format displays the….but not the underlying value