Data Types:

Numbers:

int 45, 67, 1290

float 45.67, 89.1

complex

bool (Boolean) True or False

strings stream of characters

text field

Functions code which act upon data

modules (library which can have both data & code)

(for us to use in our programs, or applications)

Operators:

Arithmetic:

add +

sub -

mul \*

div /

floor div //

modulo % (remainder)

Comparative (Relational):

<

>

<=

>=

==

!=

Functions used:

print()

help()

type()

len()

time.sleep()

Strings  
created using double quotes or single quotes

indexed

ordered

immutable

sliced

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| s | p | a | n | l | a | b | s |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| -8 | -7 | -6 | -5 | -4 | -3 | -2 | -1 |

Escape sequences:

\n new line

\t tab space

\r return (new line)

\f form feed

"he said "he will come", so let us wait for him"

str.upper()

str.lower()

Data structures

list

tuple

set

dictionary

Lists:

diff types of data types

indexed

mutable

ordered

sliced

Tuples:

diff types of data types

indexed

immutable

ordered

sliced

Sets:

diff types of data (members should be immutable)

not indexed

no duplicates

unordered

members can be added (set itself is mutable)

dictionary:

key:value pair

values:

diff types of data

immutable/mutable

keys:

immutable

no duplicates

not indexed

unordered

members can be added (set itself is mutable)

Control Constructs:

if

else

RegEx  
pattern = ^a..a$

asia

area

africa

alia

Meta charaters

^ $ [] () . + {} \

[a-d] 🡪 a,b,c,d

[A-G] 🡪 ABCDEFG

[0-5] 🡪 0,1,2,3,4,5

[a-dt] 🡪 abcdt

[0-39] 🡪01239

[^0-5] 🡪 except 012345

[0-9]{2,4}🡪 123

2347

789012 X

1 X

a|d

\d digits (similar [0-9])

\D any non-digit (similar [^0-9])

\s whitespace

(space, tab, newline…)

\S

\w alphanumeric (digits & alphabets) and underscore

string =