



Python Training.....

-- Jeetendra Bhattad



Agenda

- Data-Types & Containers
- Operators
- Control flow
- main ?
- Functions
- Command line arguments
- Modules & Packages



Data Types & Containers

- Dynamically Typed
- Numbers
 - Integer
 - Float
 - Complex
- Data Model
 - Value
 - Reference
 - Garbage Collection using reference count mechanism
- Containers
 - String
 - List
 - Tuple
 - Set
 - Dictionary



Operators

- Arithmetic : `**`, `//`, `+`, `-`, `*`, `/`, `%`
- Relational : `<>`, `!=`, `==`, `>`, `>=`, `<`, `<=`
- Logical : `and`, `or`, `not`
- Bitwise : `&`, `|`, `^`, `>>`, `<<`, `~`
- Membership : `in`, `not in`
- Identity : `is`, `is not`



Control Flow

- Selection
 - if
 - if-elseif
- Iteration
 - while
 - while-else
 - for
 - for - else
- Jump
 - break
 - continue
 - pass
 - return



main

```
#!/usr/bin/python
```

```
def Add(a, b):
```

```
    return a+b
```

```
if __name__ == "__main__":
```

```
    print(Add(10, 20))
```



Functions

```
#!/usr/bin/python
```

```
def Add(a, b):
```

```
    return a+b
```

```
def Sub(a, b=1):
```

```
    return a-b
```

```
if __name__ == "__main__":
```

```
    print(Add(10, 20))
```

```
    print(Add(b=20, a=30))
```

```
    print(Sub(20, 10))
```

```
    print(Sub(10))
```

Continued.....



Variable Arguments

```
#!/usr/bin/python
```

```
def VariableArgumentDemo(*args):  
    ''' Variable Number of Arguments of Demo '''  
    for x in args:  
        print x  
    else:  
        print 'Executing else'
```

```
VariableArgumentDemo(1,2,3,4)
```

```
VariableArgumentDemo(1,2,3,4,5,6,7,8)
```

Continued.....



Variable Arguments & Dictionary

```
#!/usr/bin/python
```

```
def VariableArgumentDictionaryDemo(a,b,*args,**xargs):
```

```
    """ Variable number of arguments & dictionary """
```

```
    print 'Value of a is %d'%a
```

```
    print 'Value of b is %d'%b
```

```
    for x in args:
```

```
        print x
```

```
    for x in xargs:
```

```
        print x, xargs[x]
```

```
VariableArgumentDictionaryDemo(1,2,1,2,3,name="Jeetendra", Surname="Bhattad")
```

```
VariableArgumentDictionaryDemo(1,2,name="Jeetendra", Surname="Bhattad")
```



Command Line Arguments

```
#!/usr/bin/python
```

```
import sys
```

```
if __name__ == "__main__":
```

```
    print(sys.argv)
```



Modules & Packages

- Any .py file is a module
- `import <module-name>`
- `from <module-name> import <specific-function/class-name>`
- `import` v/s `from import`
- `from <module-name> import *`
- Packages are just folders of modules with a special `init.py` file which indicates to python that this folder is special because it contains python modules.
- `dir(module-name)`



Built-in functions

- enumerate
- zip
- max
- min
- globals
- any – true if iterable
- Functional support :-
 - map
 - reduce
 - filter