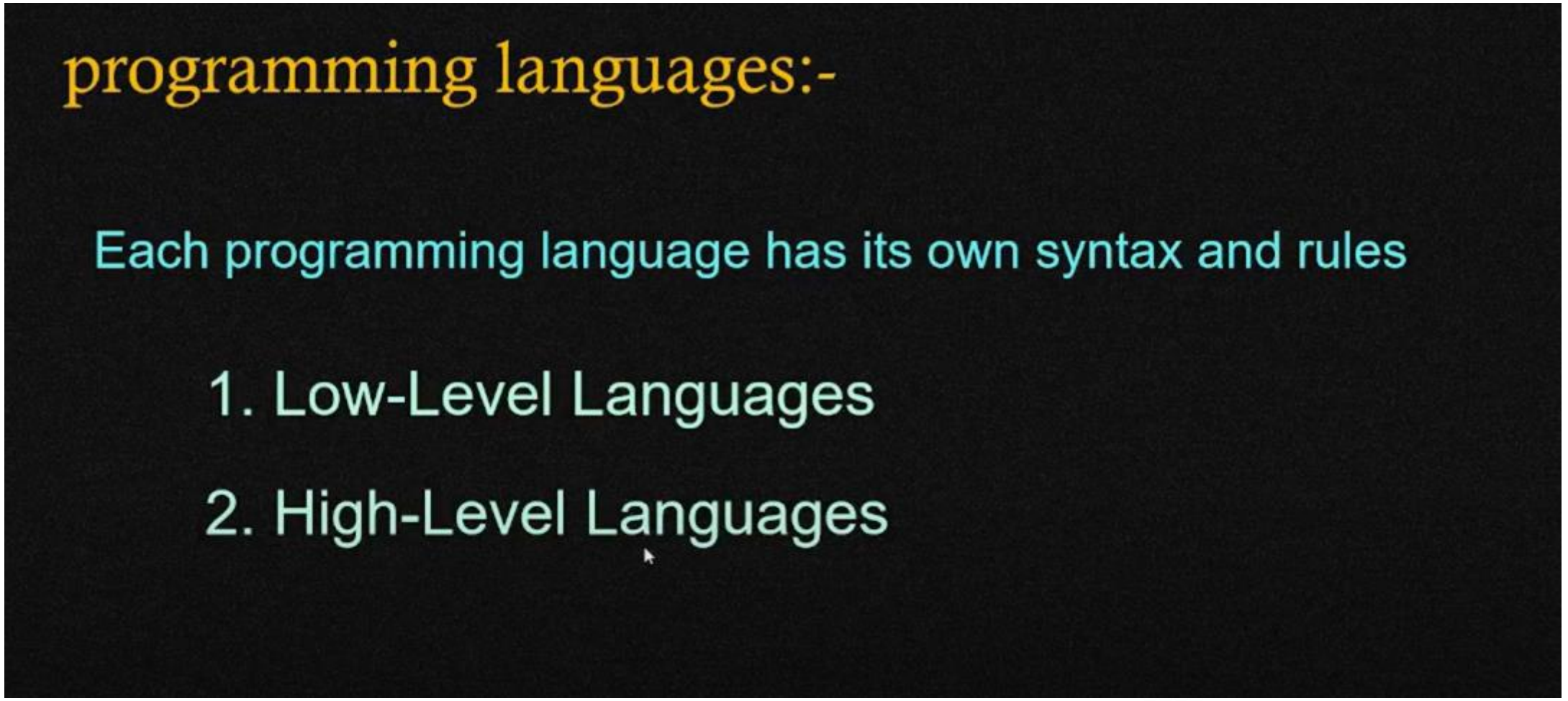
PYTHON- BASICS

Dr Sivabalan,

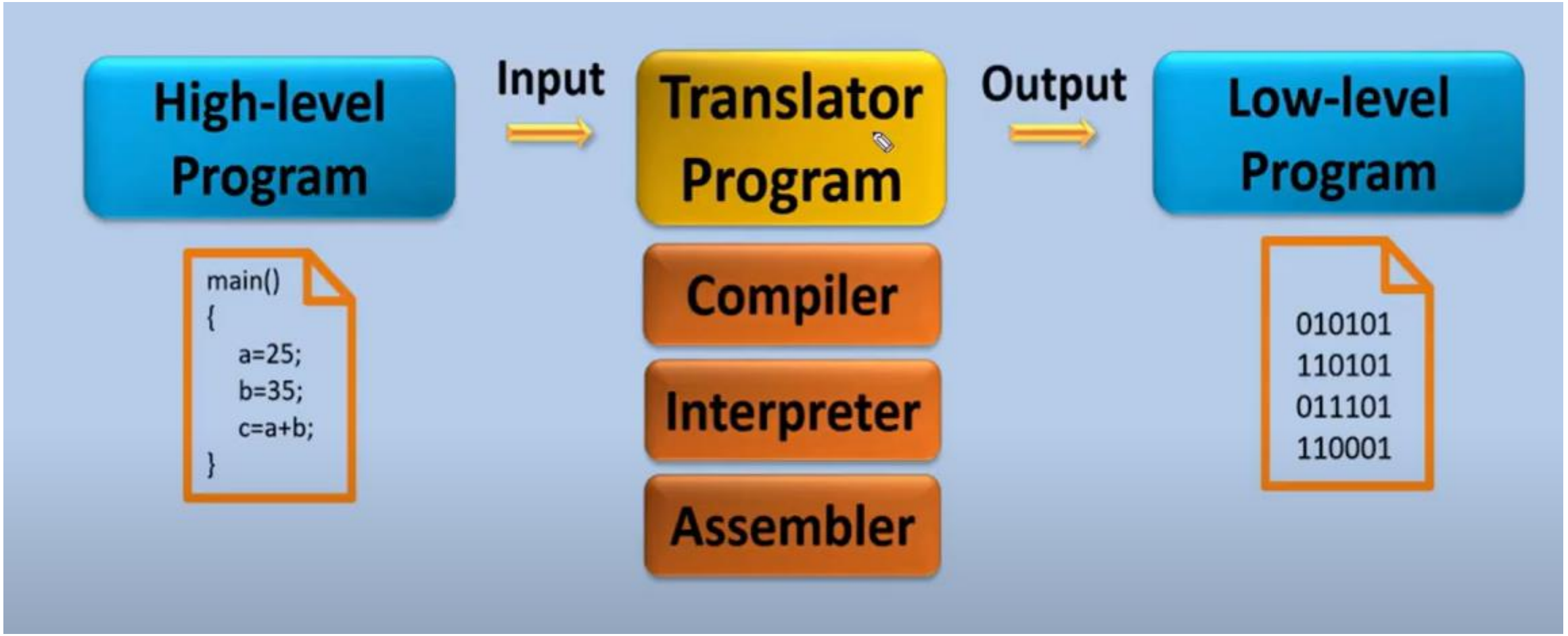
Technical Training Advisor

Sivabalan.n@nttdata.com

Python-Basics



Python-Basics



Python-Basics

What is Python?

Python is a high level programming language like C,C++

Python is a simple, high level, interpreted, general purpose, dynamically typed and object oriented programming language created by Guido Van Rossum in 1991.

Python-Basics

Python-Basics

Hello World

Java:

```
// Hello World in Java
class HelloWorld {
    static public void main(String args[]) {
        System.out.println("Hello World!");
}   }
```

C++:

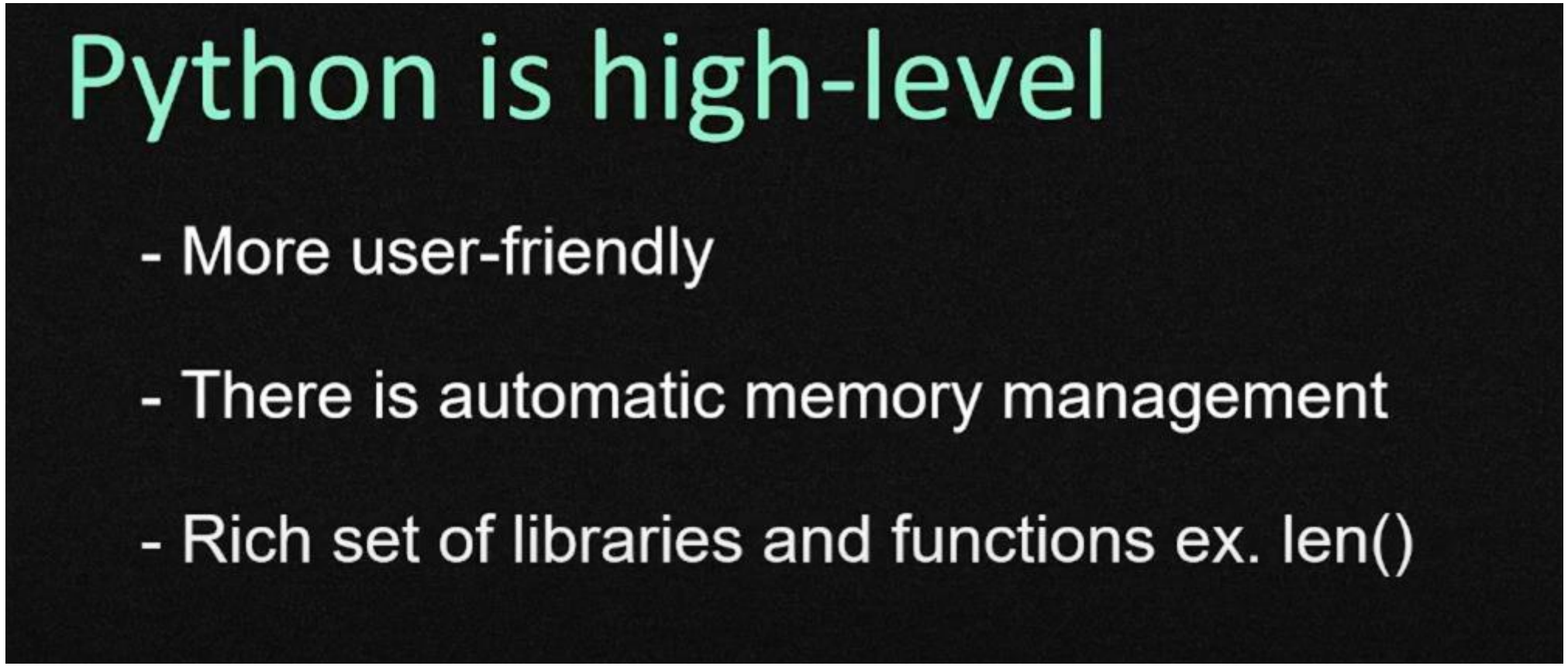
```
// Hello World in C++
#include <iostream.h>
Main() {
    cout << "Hello World!" << endl;
    return 0;
}
```

Python:

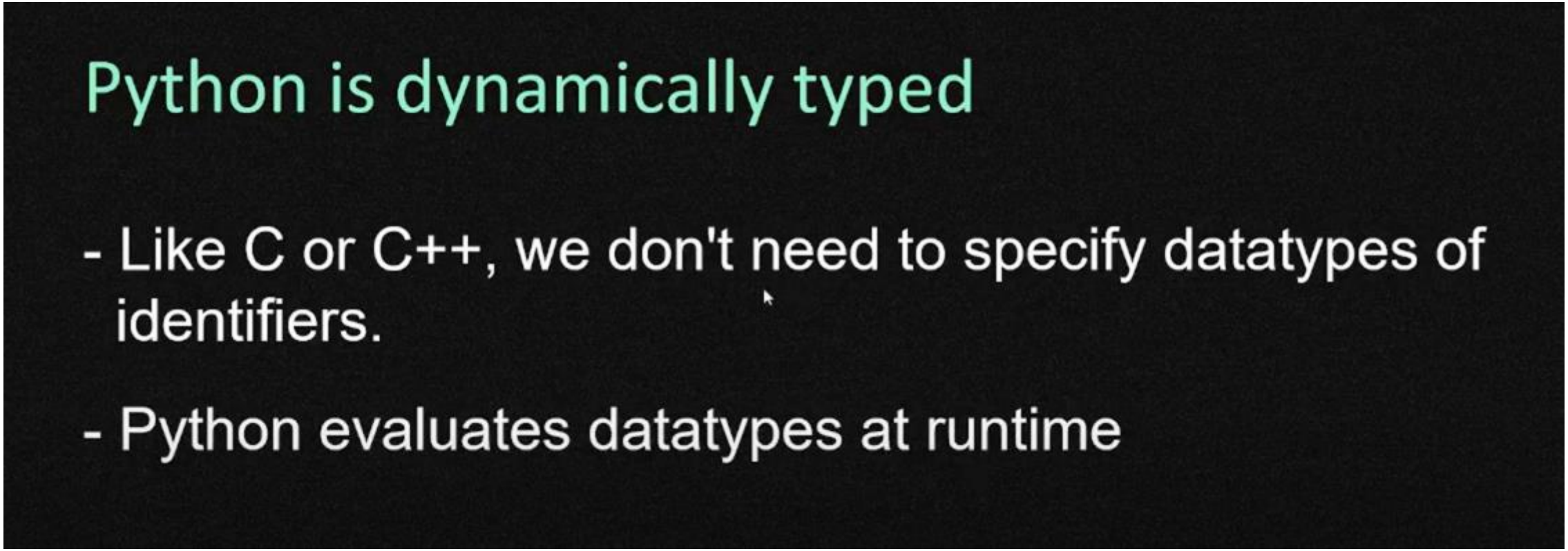
```
# Hello World in Python
print("Hello World!")
```

Python combines remarkable power with very clean, simple, and compact syntax.

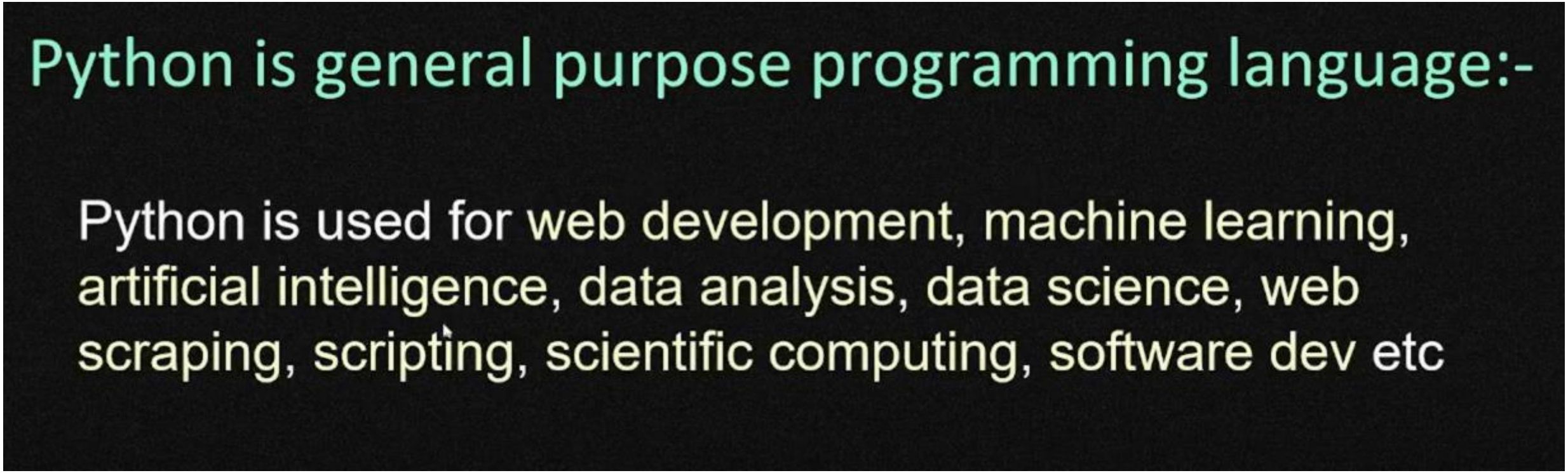
Python-Basics



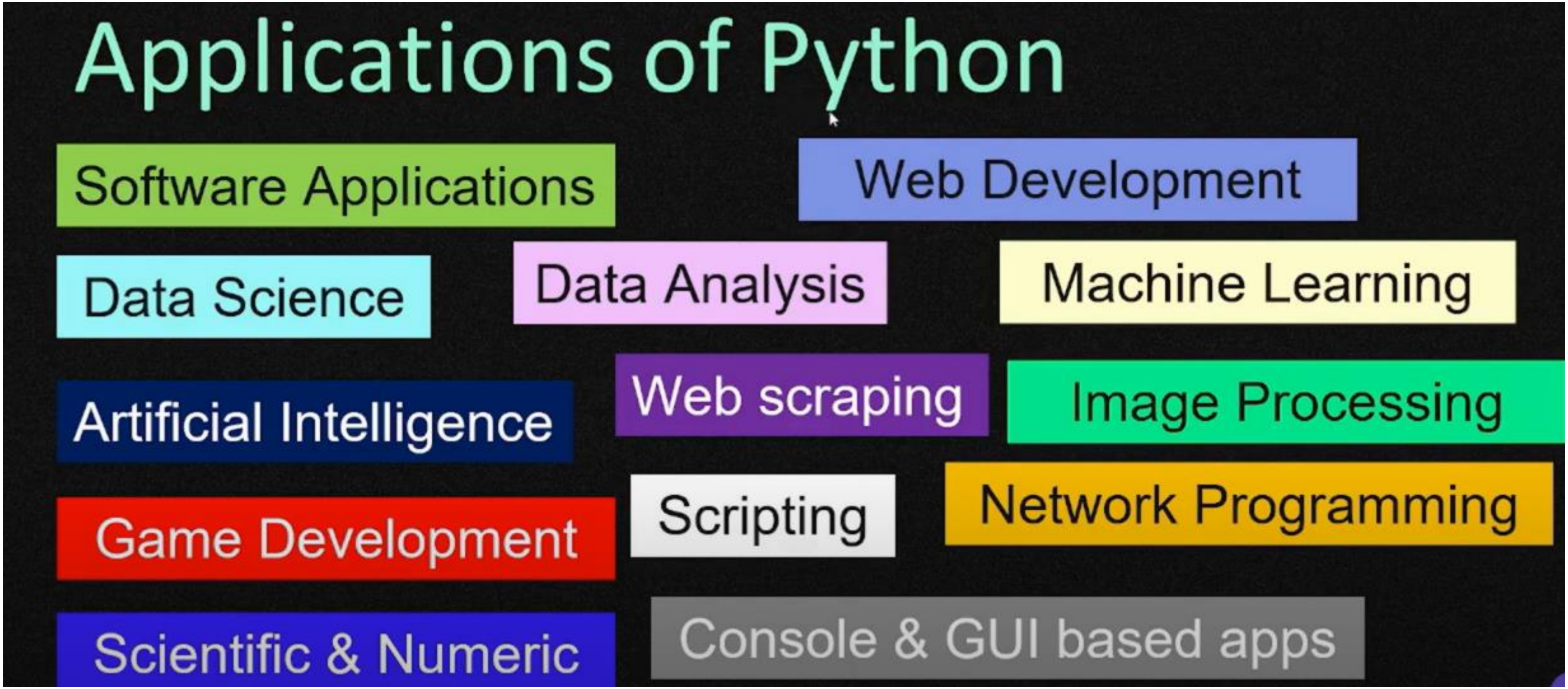
Python-Basics



Python-Basics



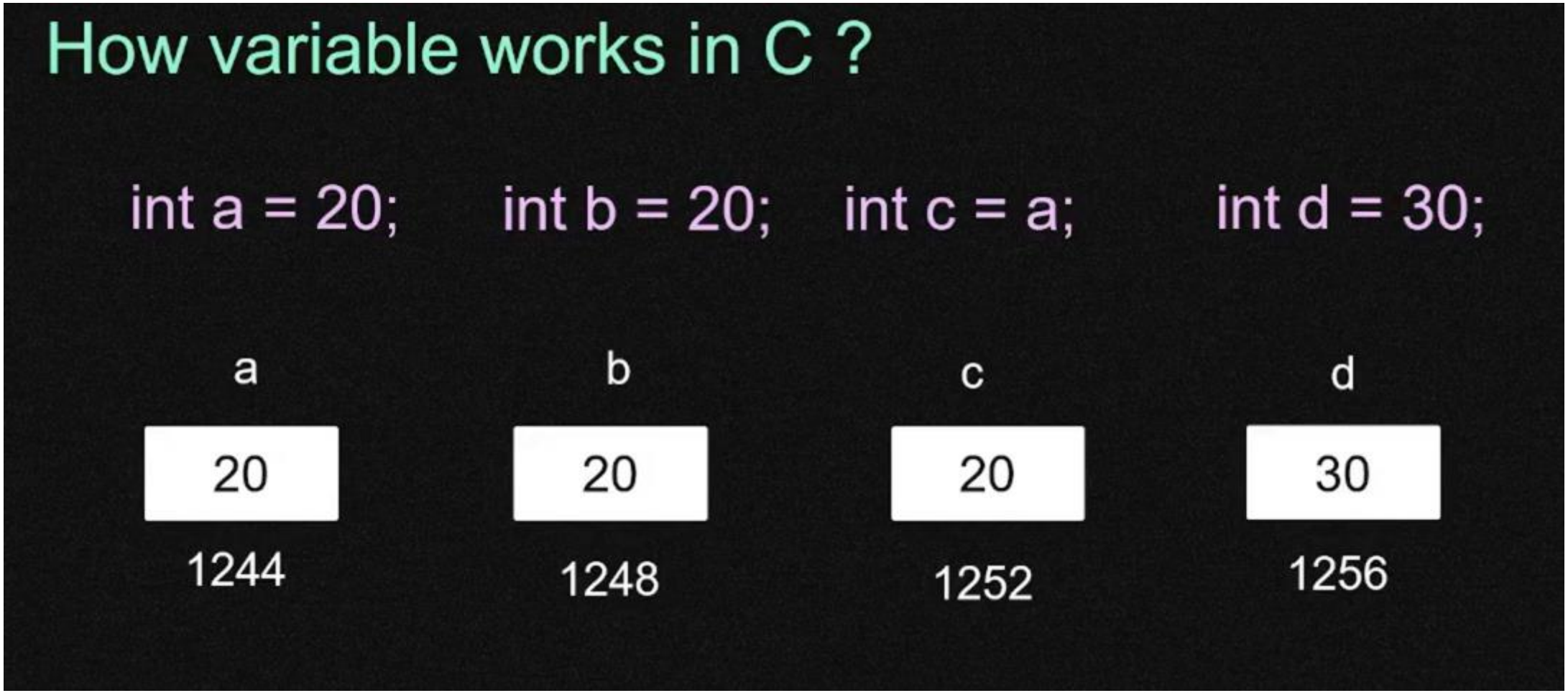
Python-Basics



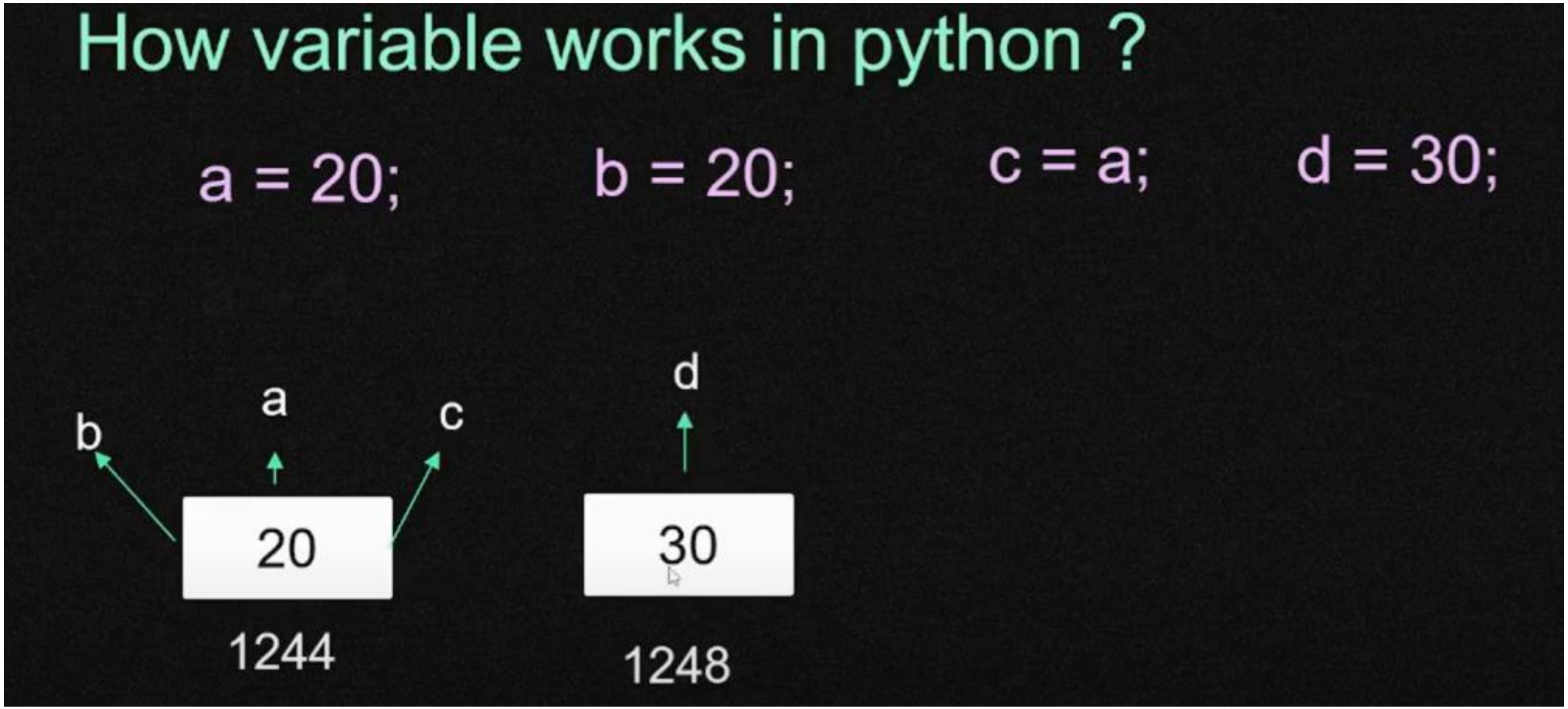
Python-Basics

Python-Basics

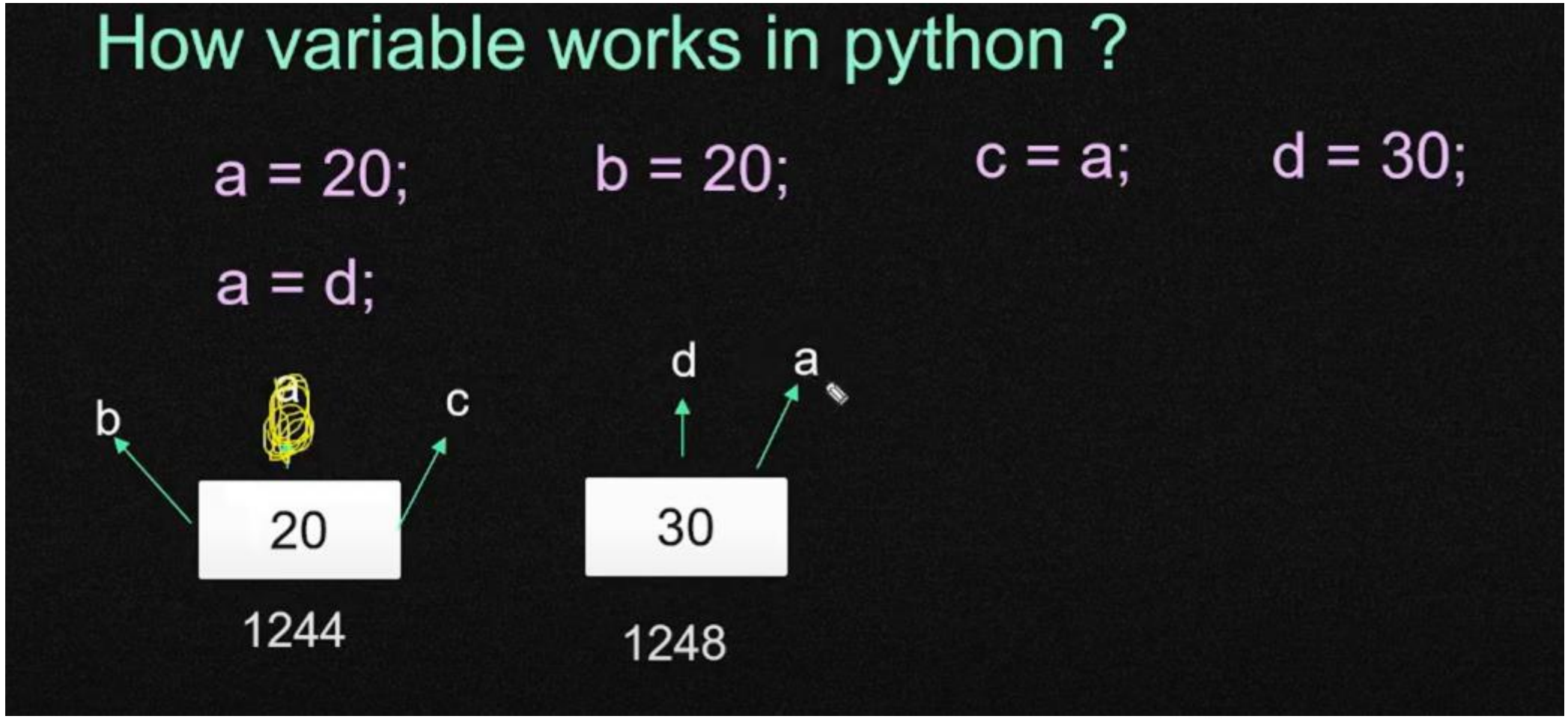
- What are variables and need of variables ?
- How to create variables & working of variables ?
- Rules for creating variables (Naming conventions)
- do’s & dont’s
- Keywords in python



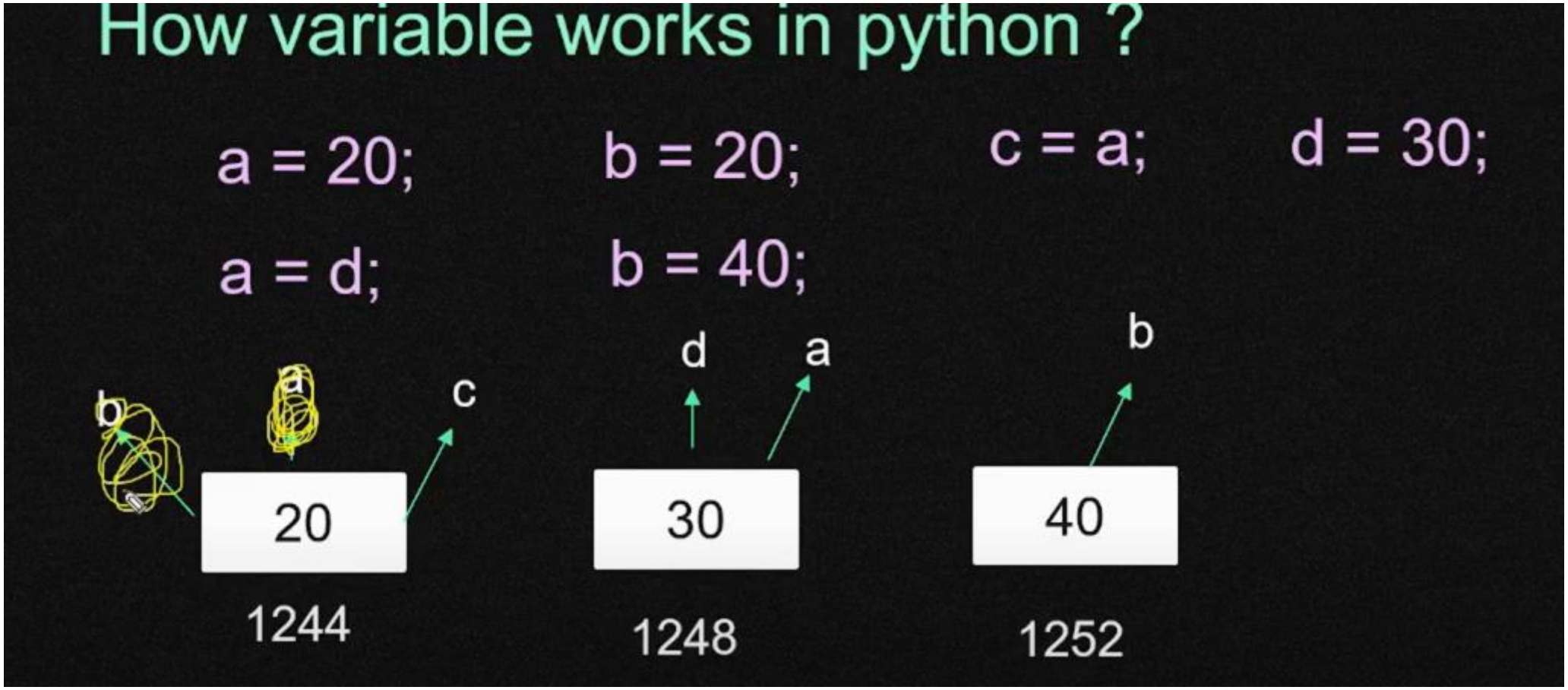
Python-Basics



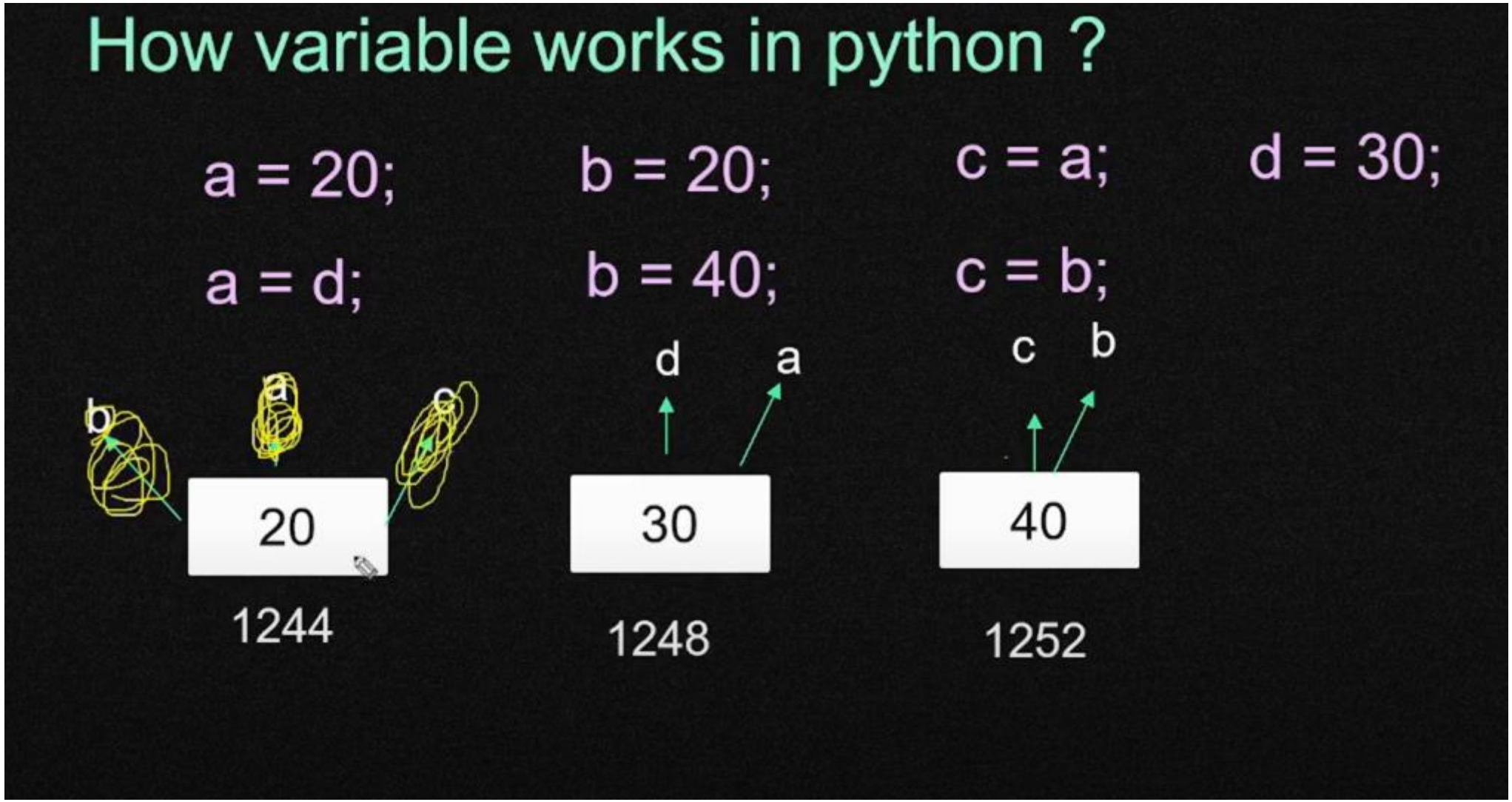
Python-Basics

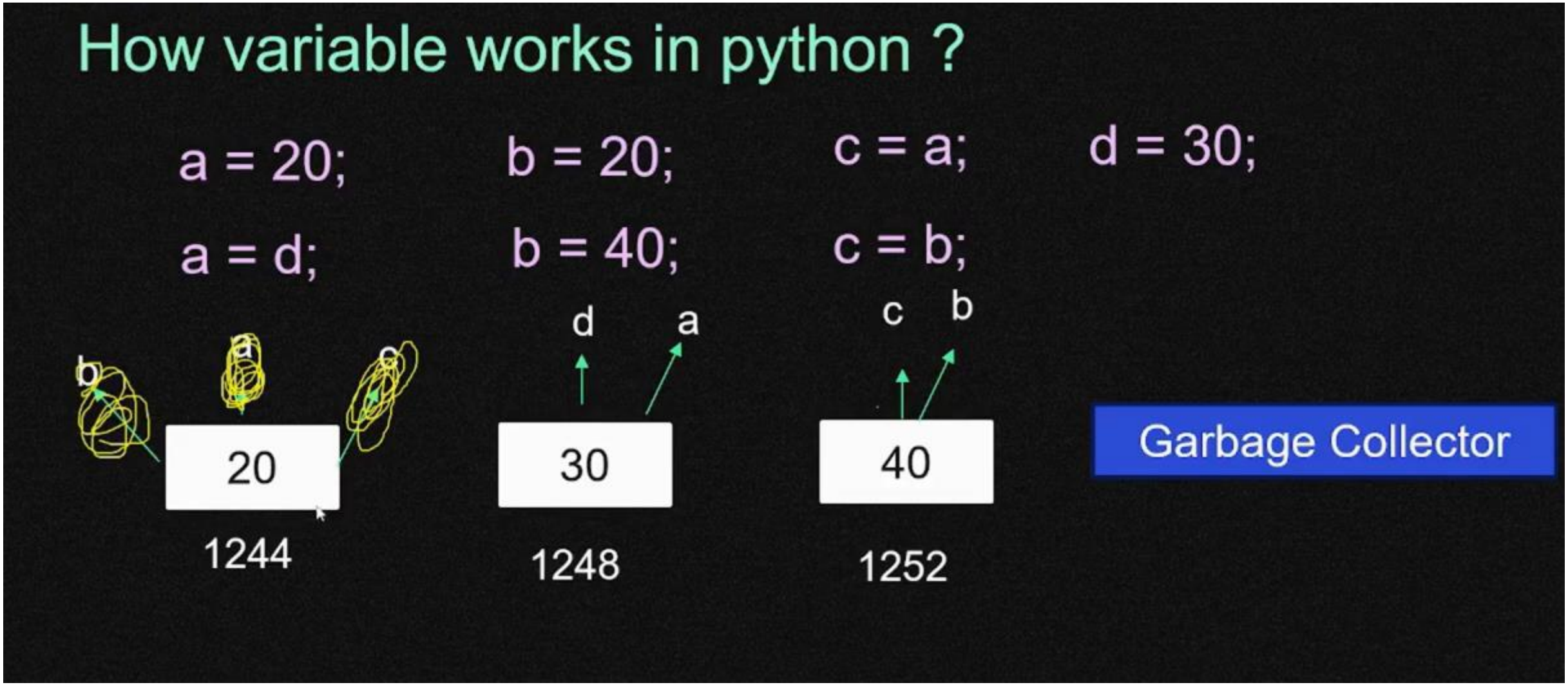


Python-Basics

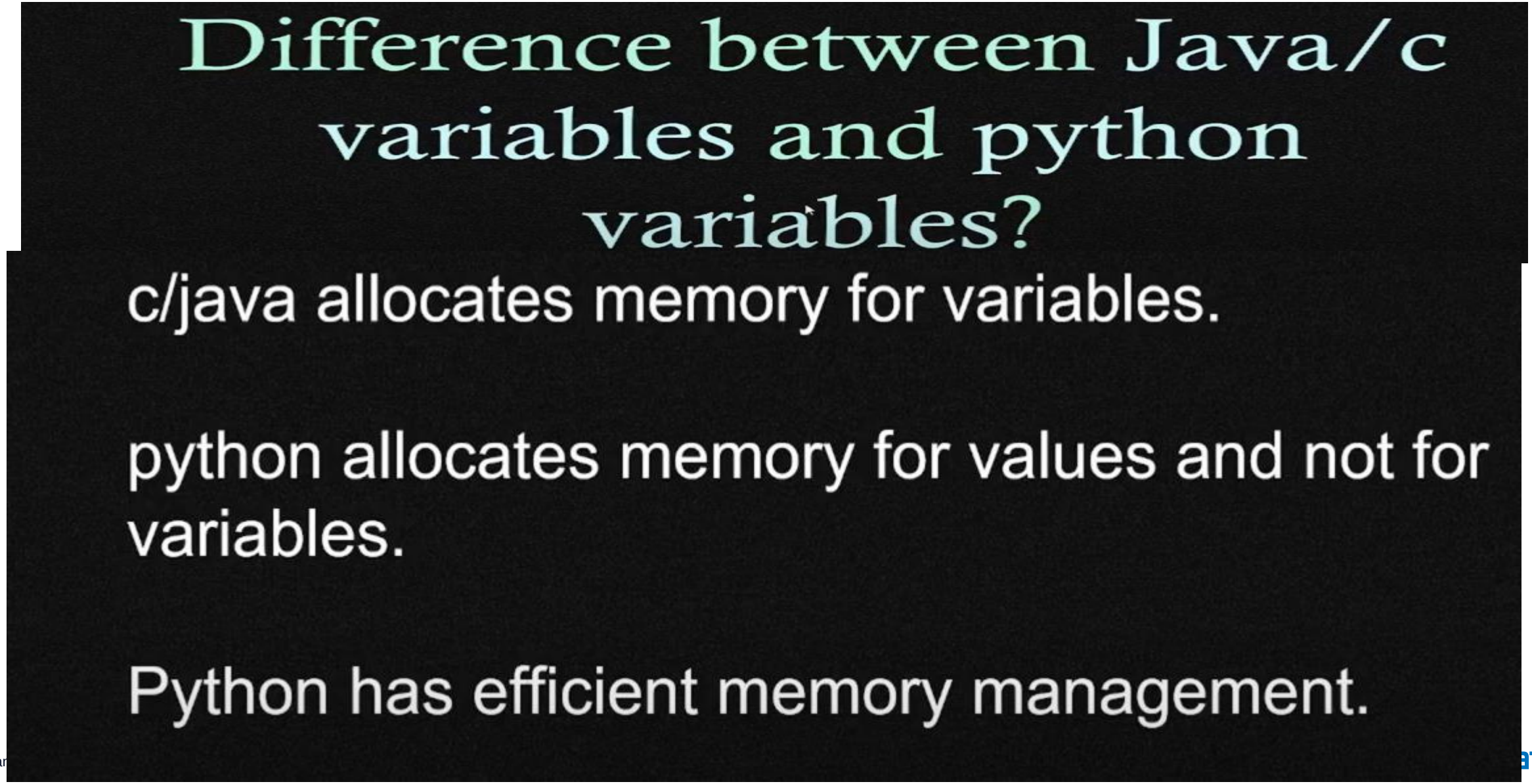


Python-Basics

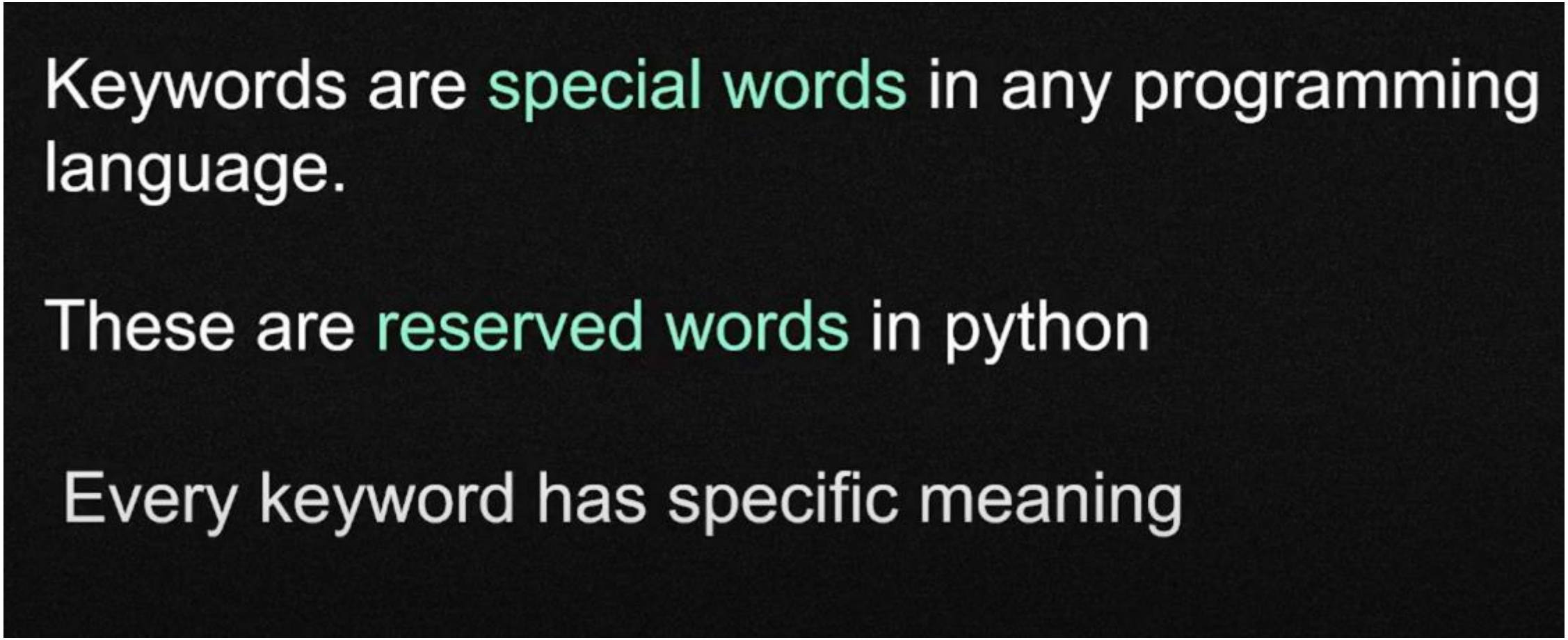




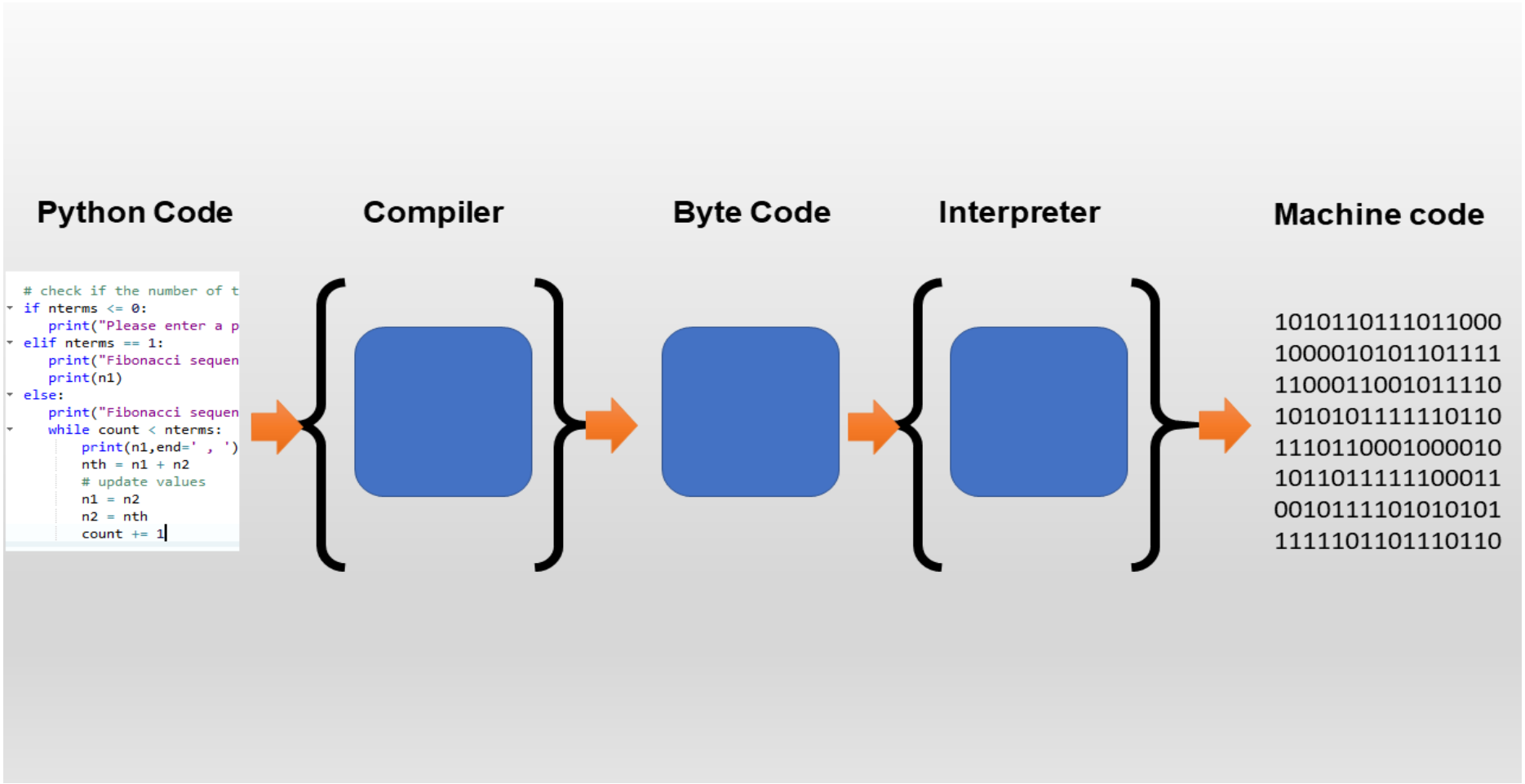
Python-Basics



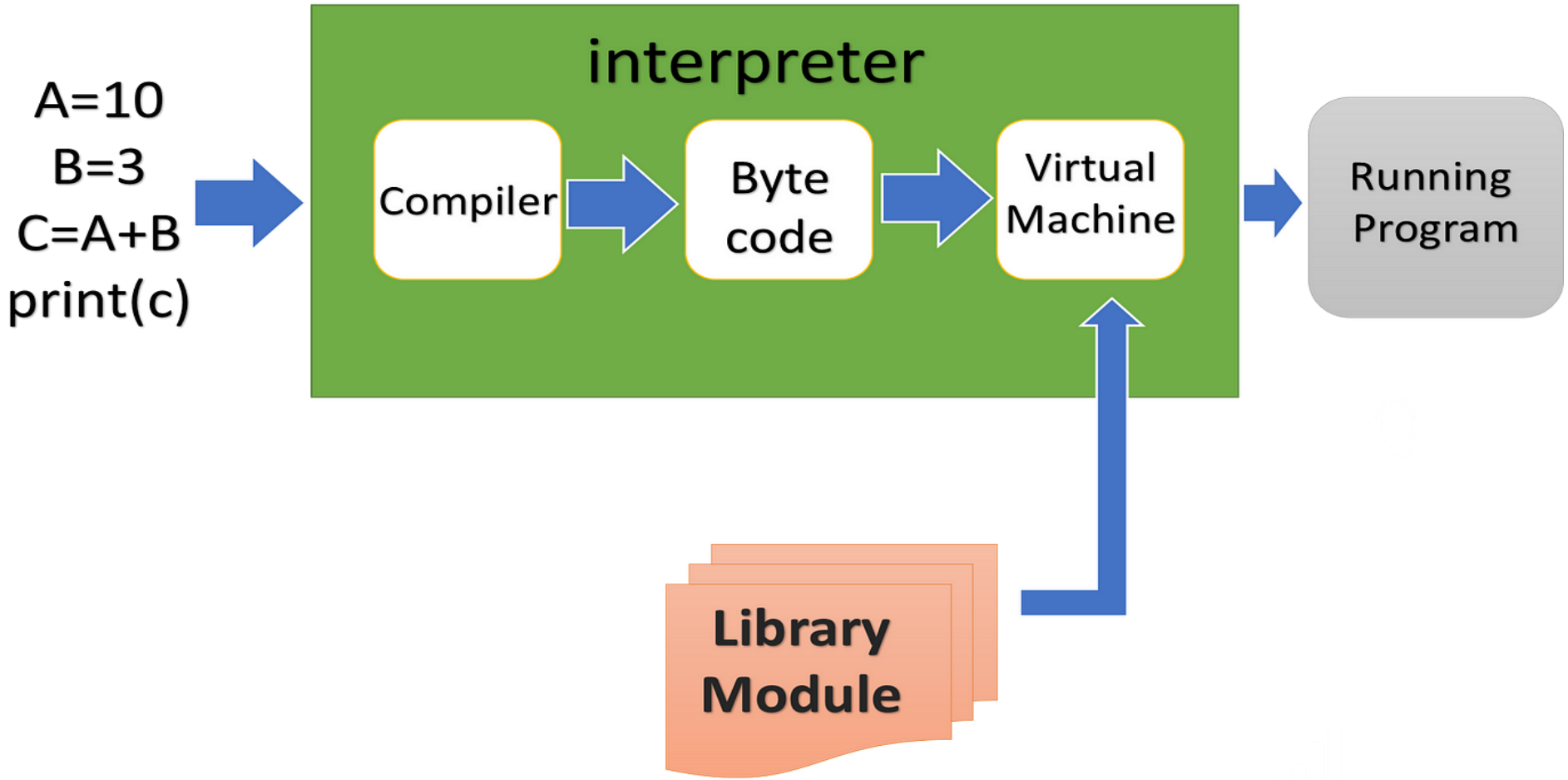
Python-Basics



Python-Basics



Python-Basics



Python-Basics

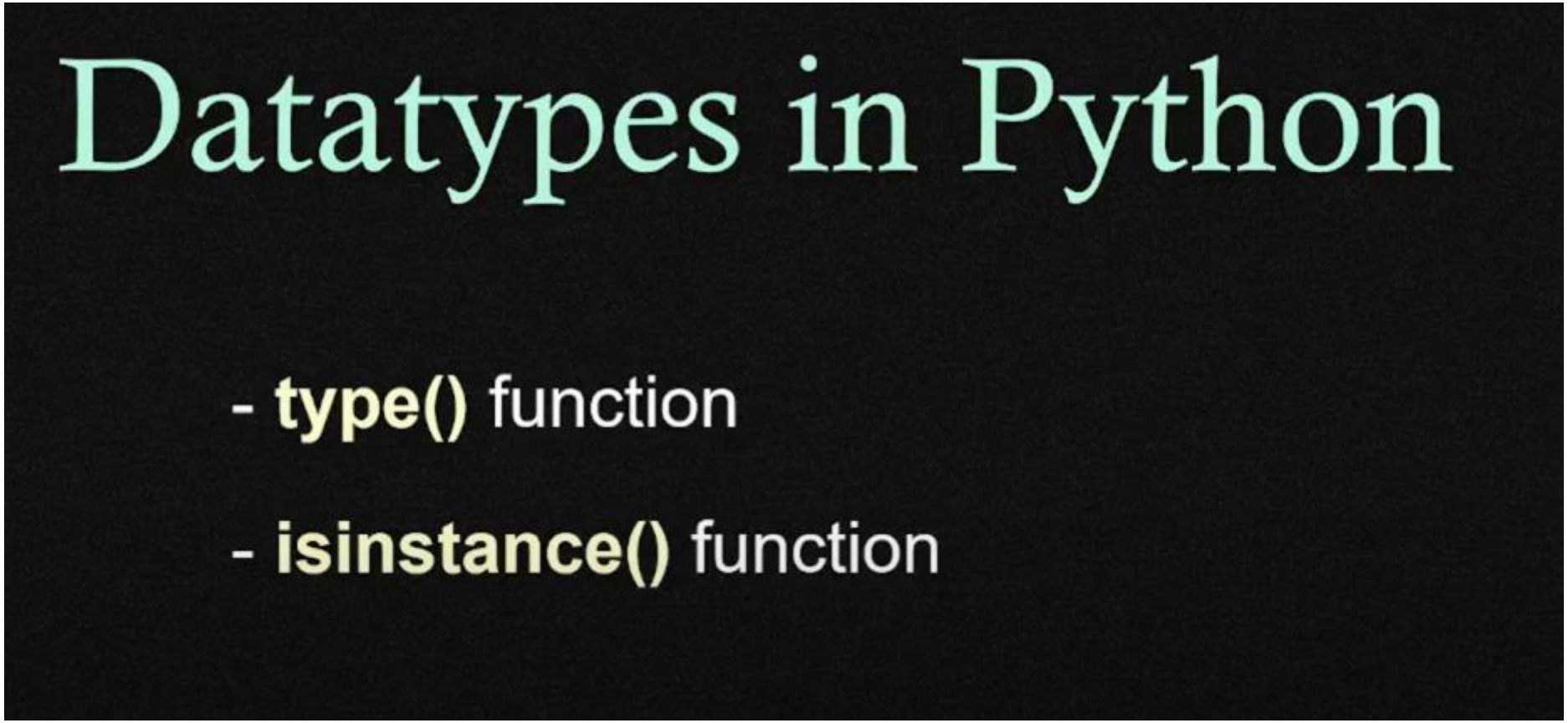
Python-Basics

Need of comments

helps you and others to understand later on the intention of your code.

This allows you to more easily find errors, to fix them, to improve the code later on, and to reuse it in other applications as well.

Python-Basics

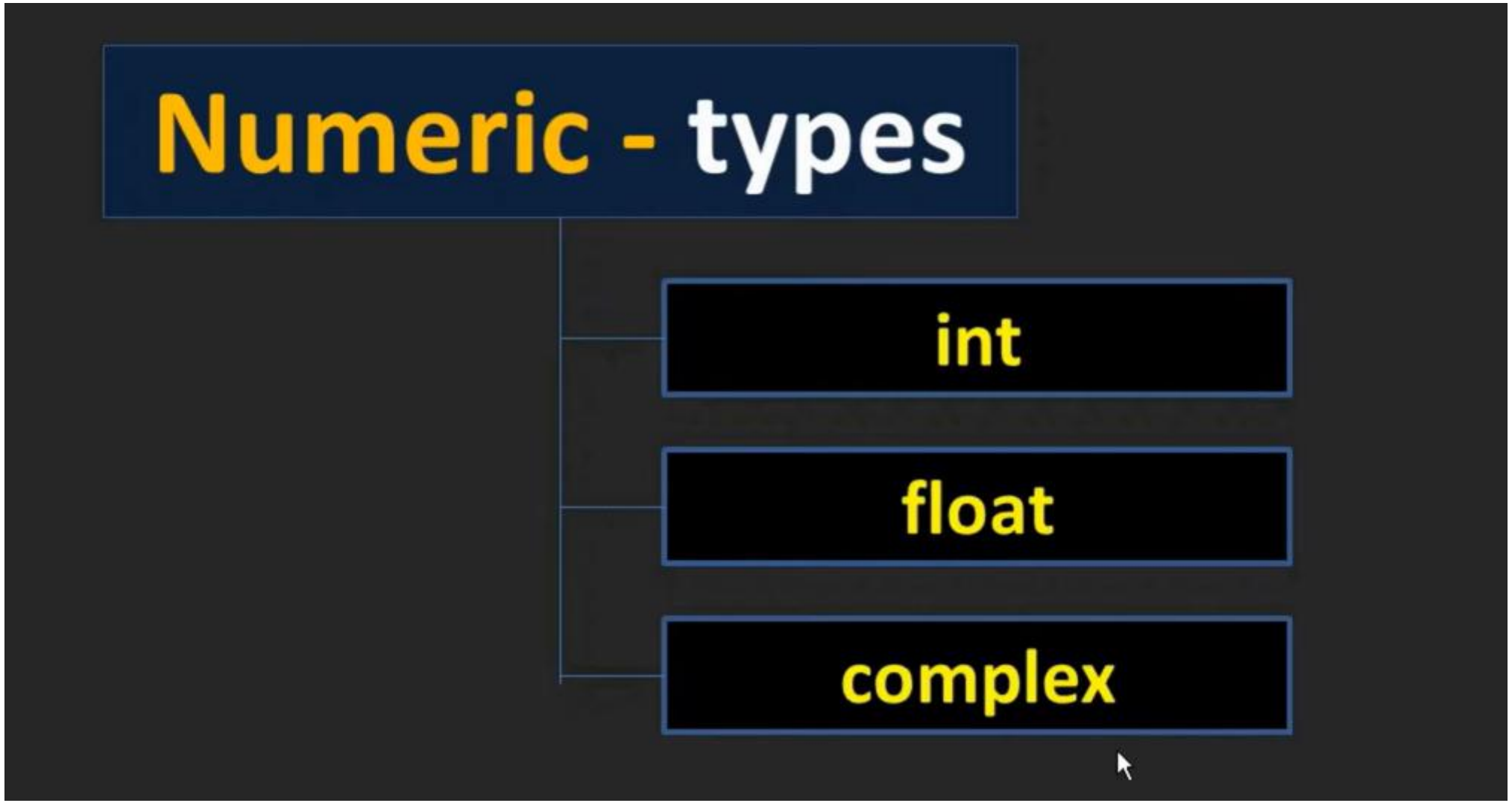


Python-Basics

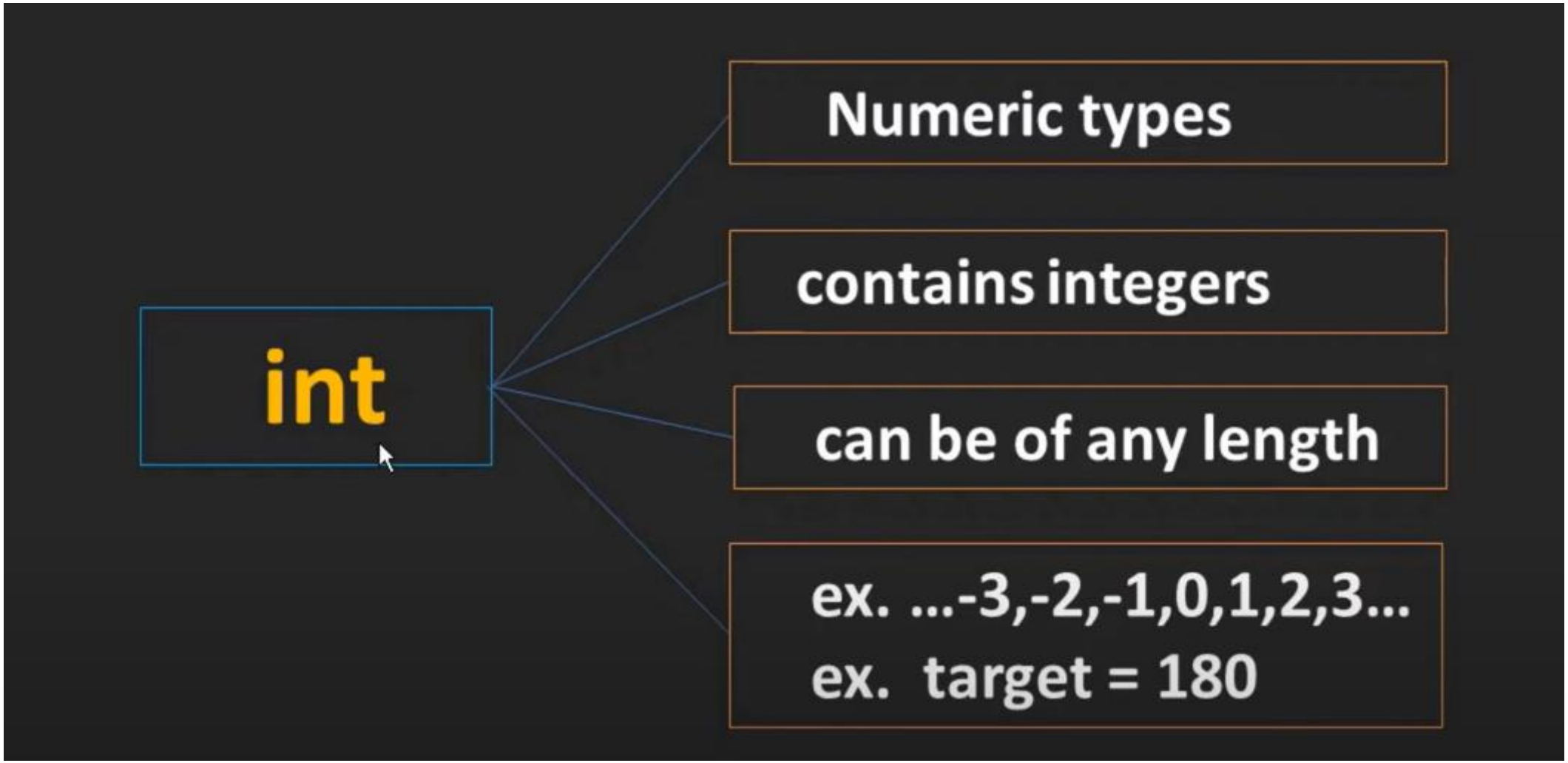
Below are datatypes available in python:-

- Numeric types:- int, float, complex (special type)
- Sequence types:- str, list, tuple
- Mapping types:- dict
- Set types:- set, frozenset
- Boolean Type:- Bool
- Binary Types:- bytes, bytearray
- None type:- NoneType

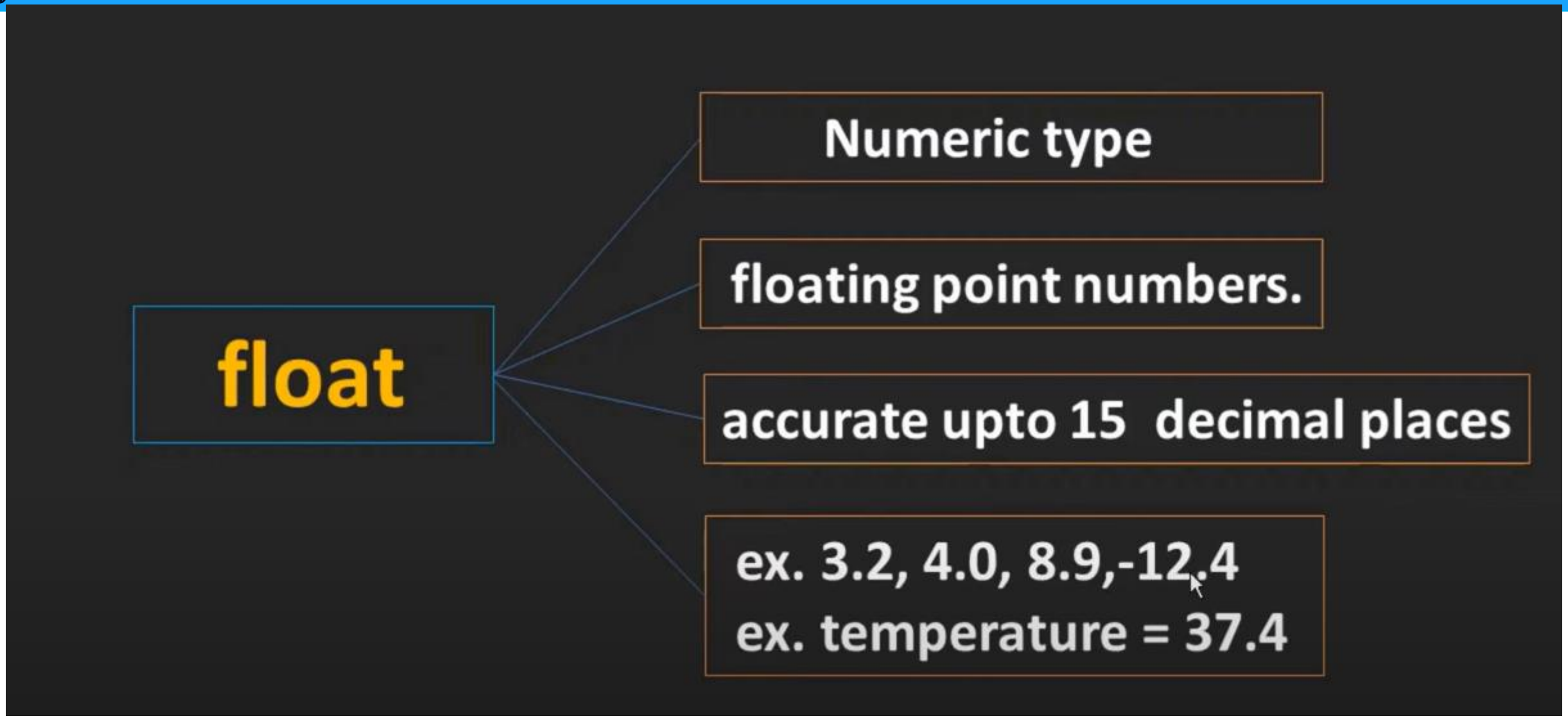
Python-Basics



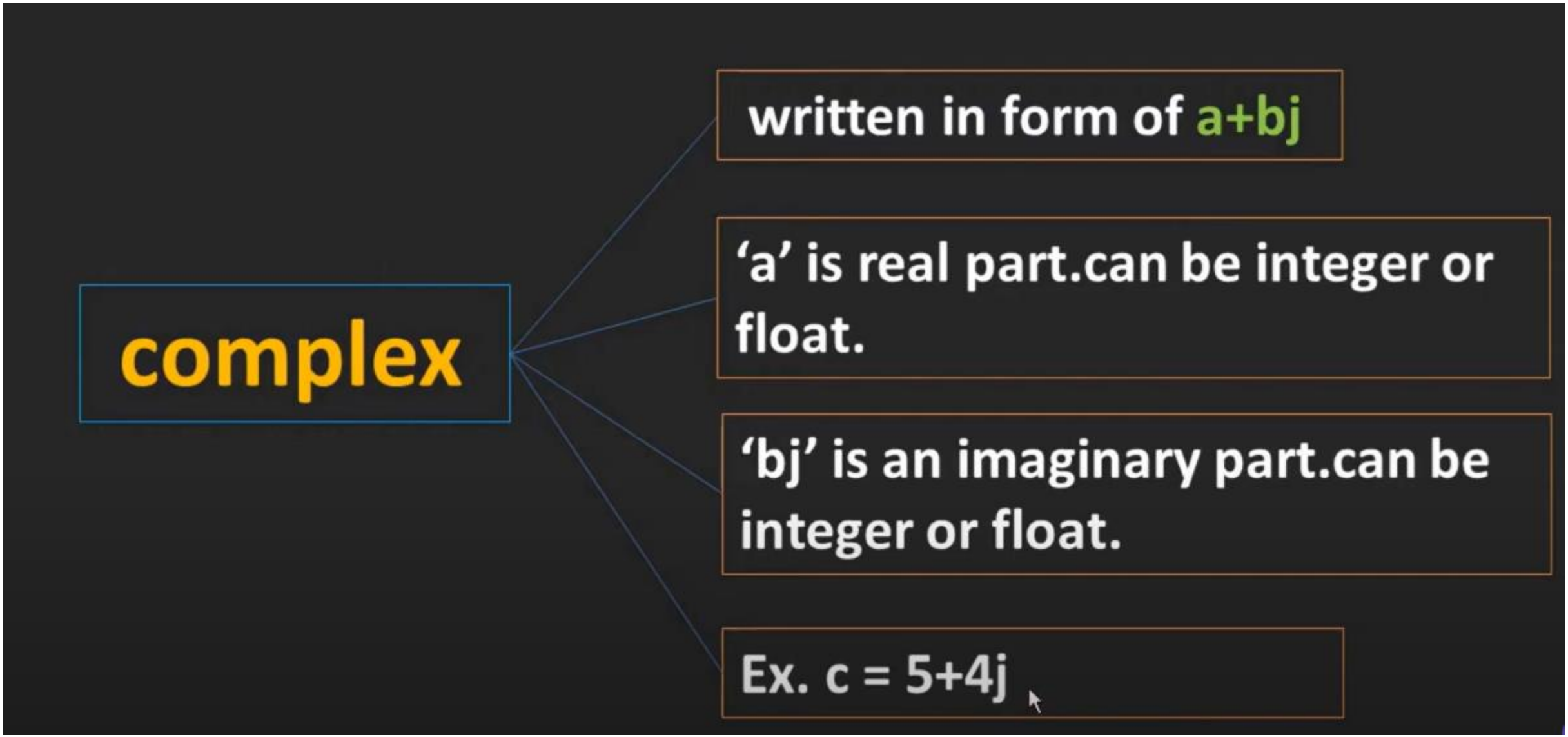
Python-Basics



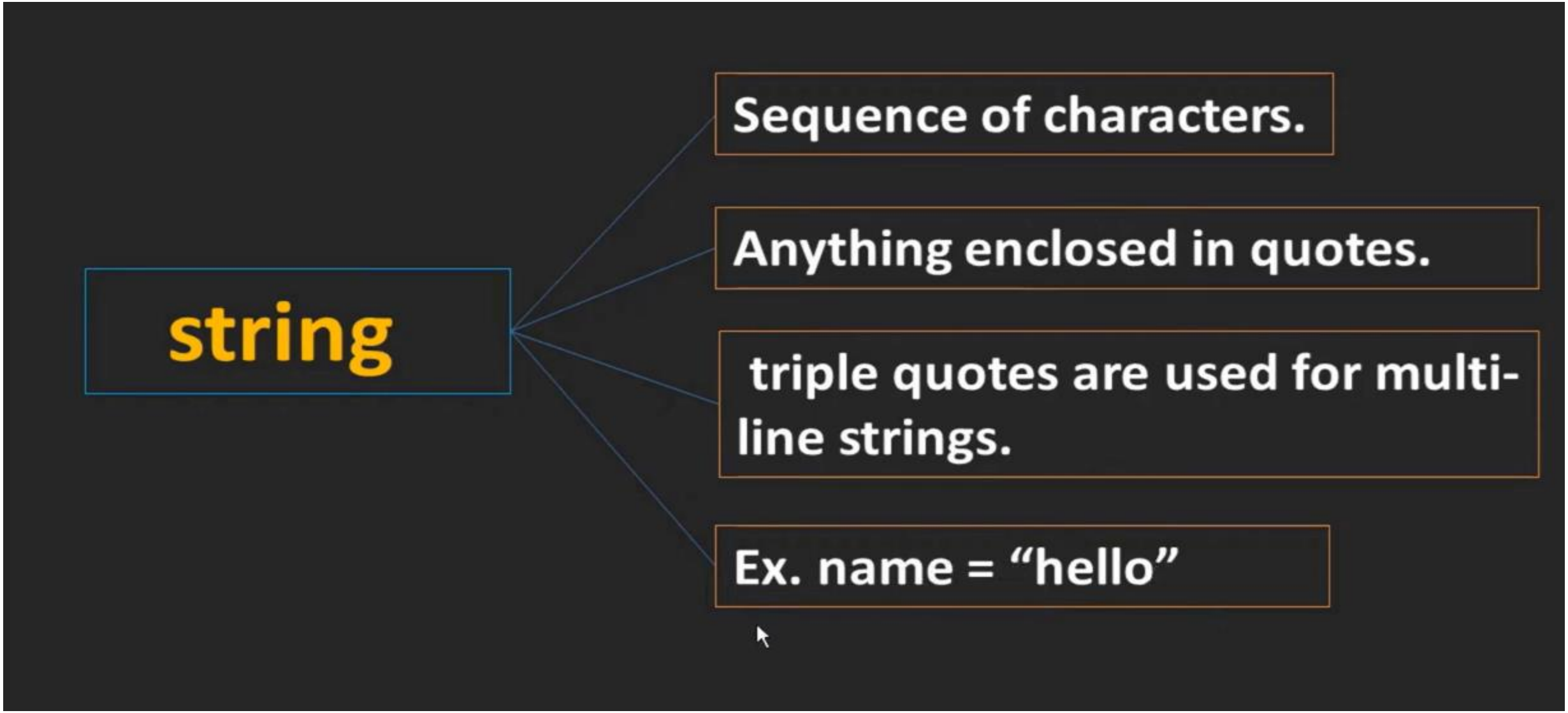
Python-Basics



Python-Basics

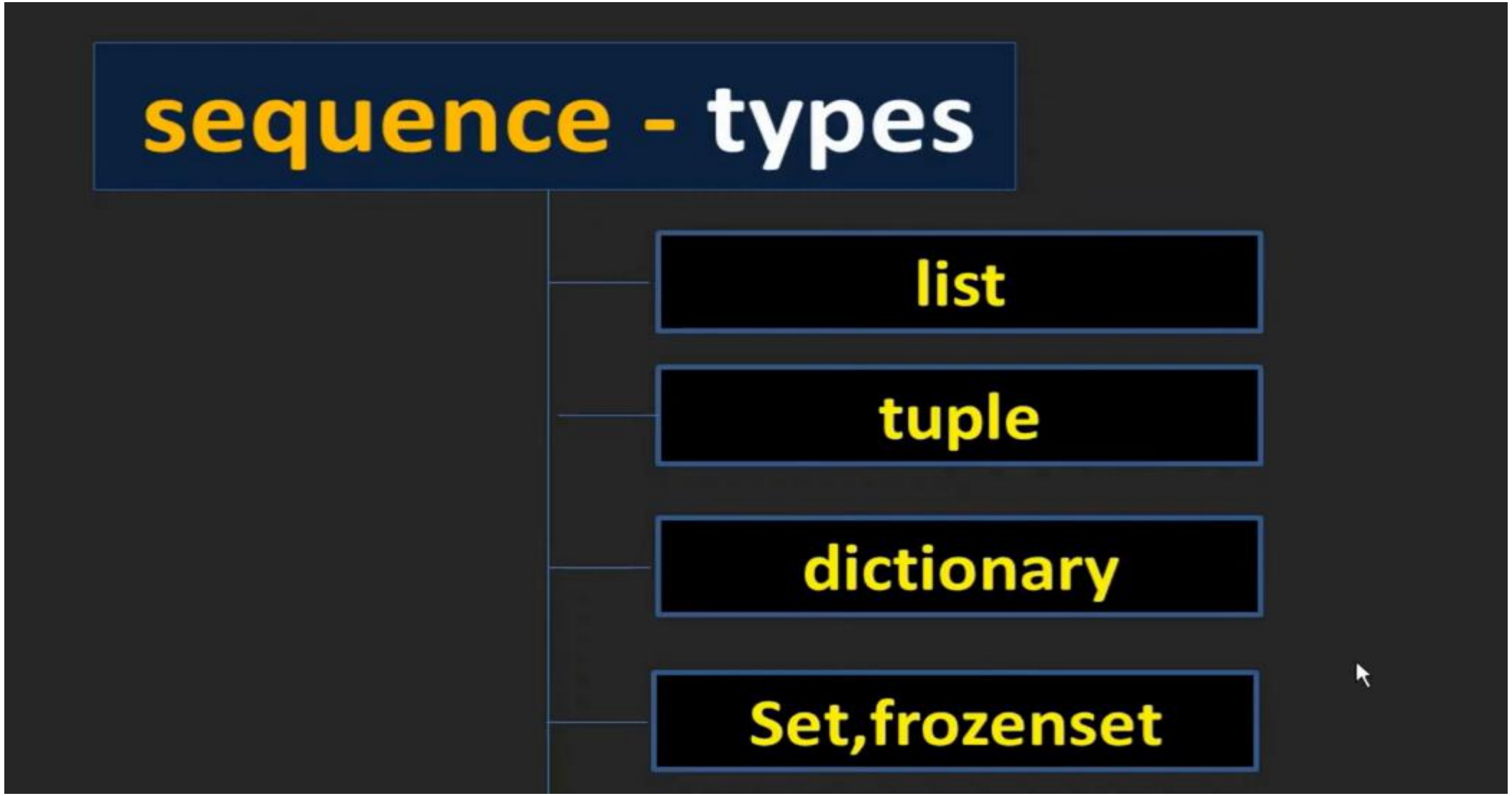


Python-Basics

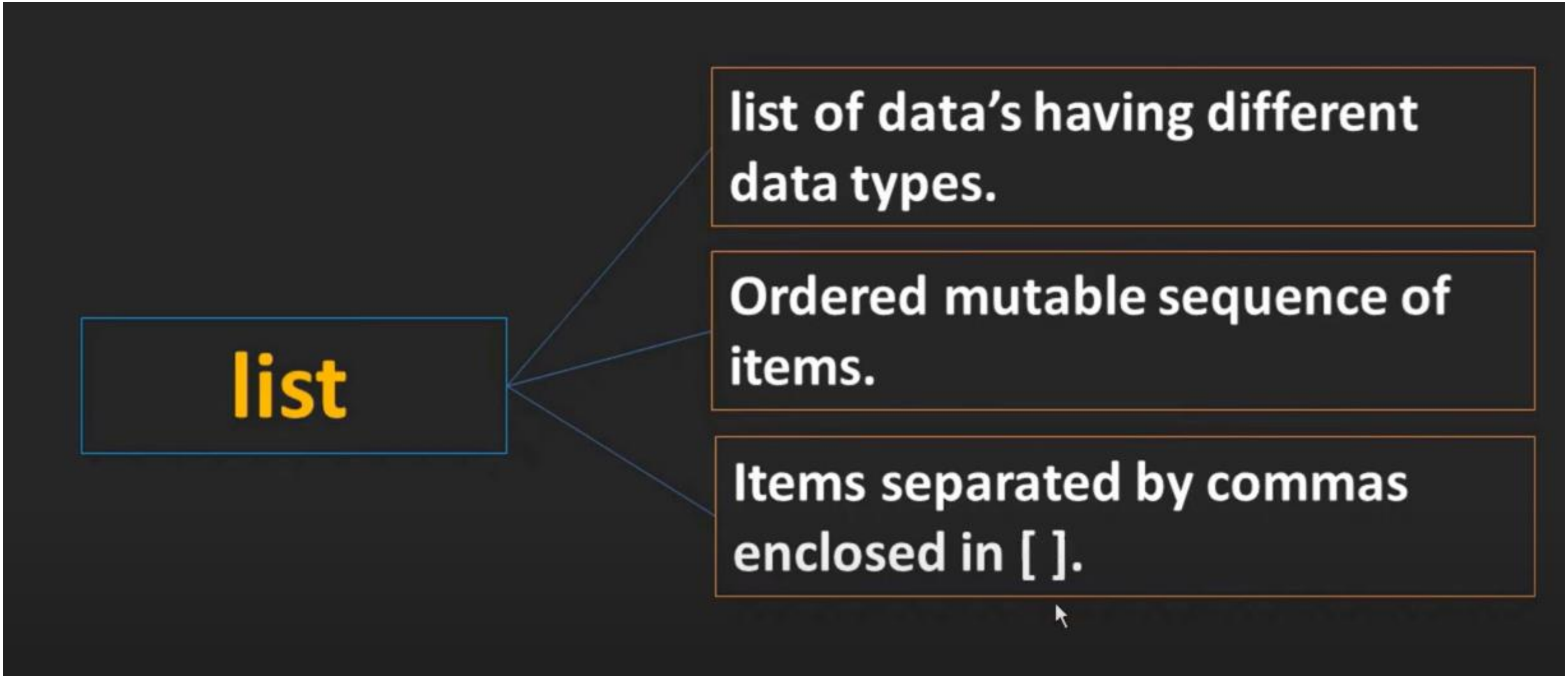


Python-Basics

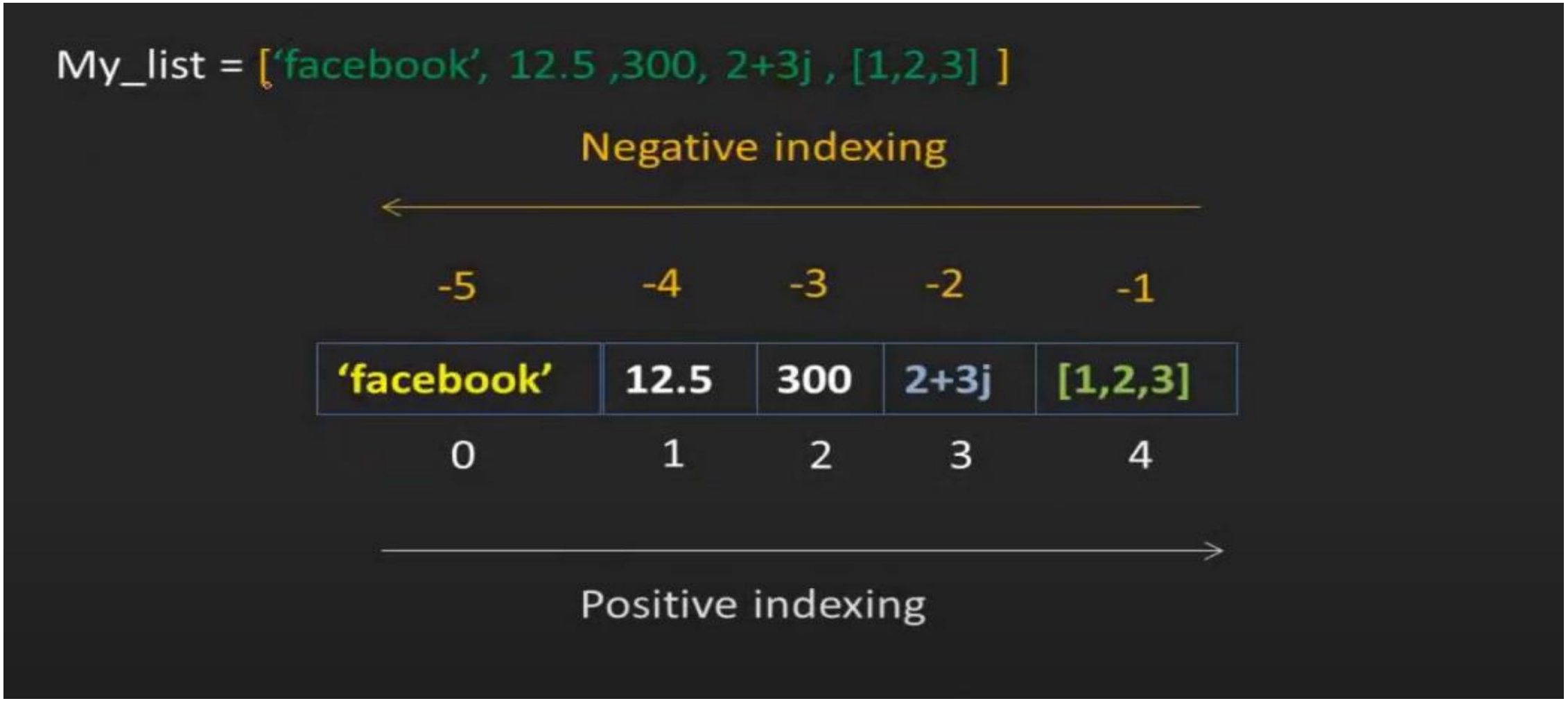
Python-Basics



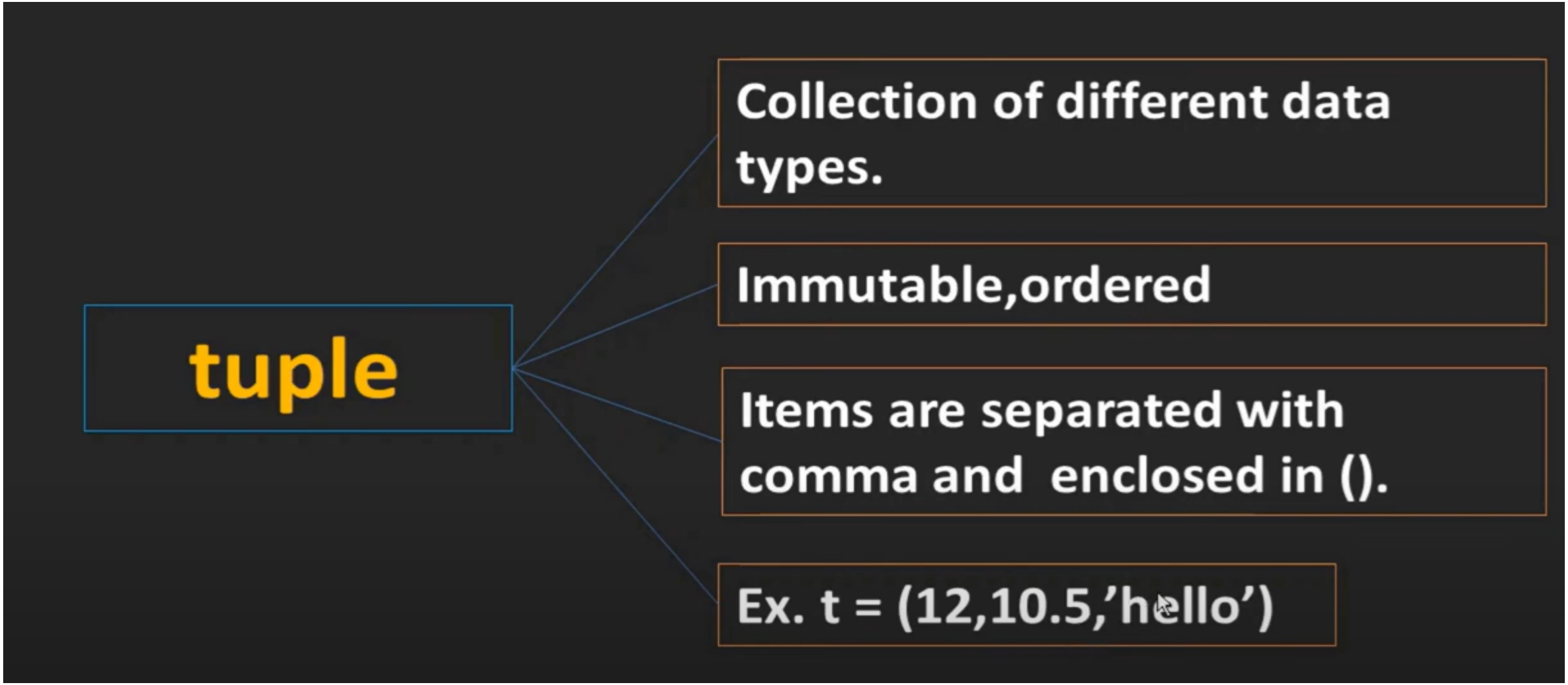
Python-Basics



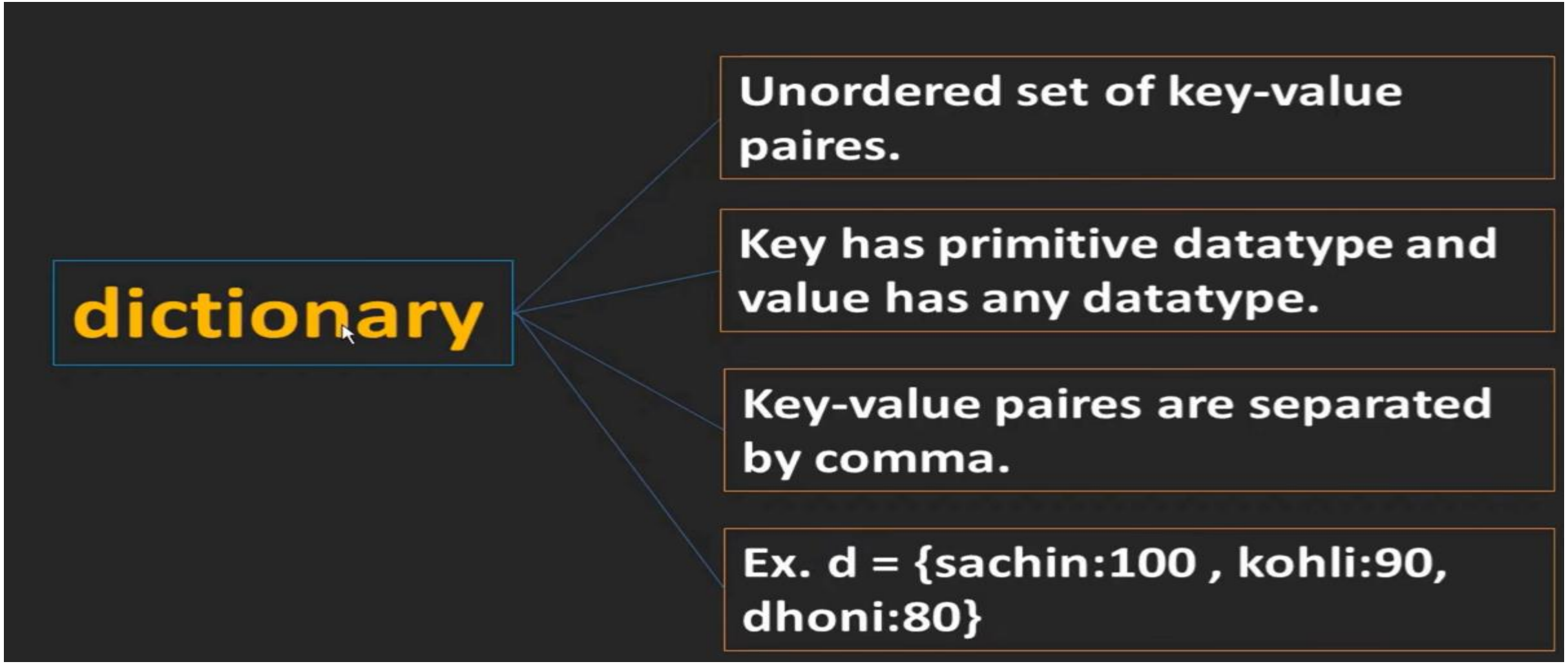
Python-Basics

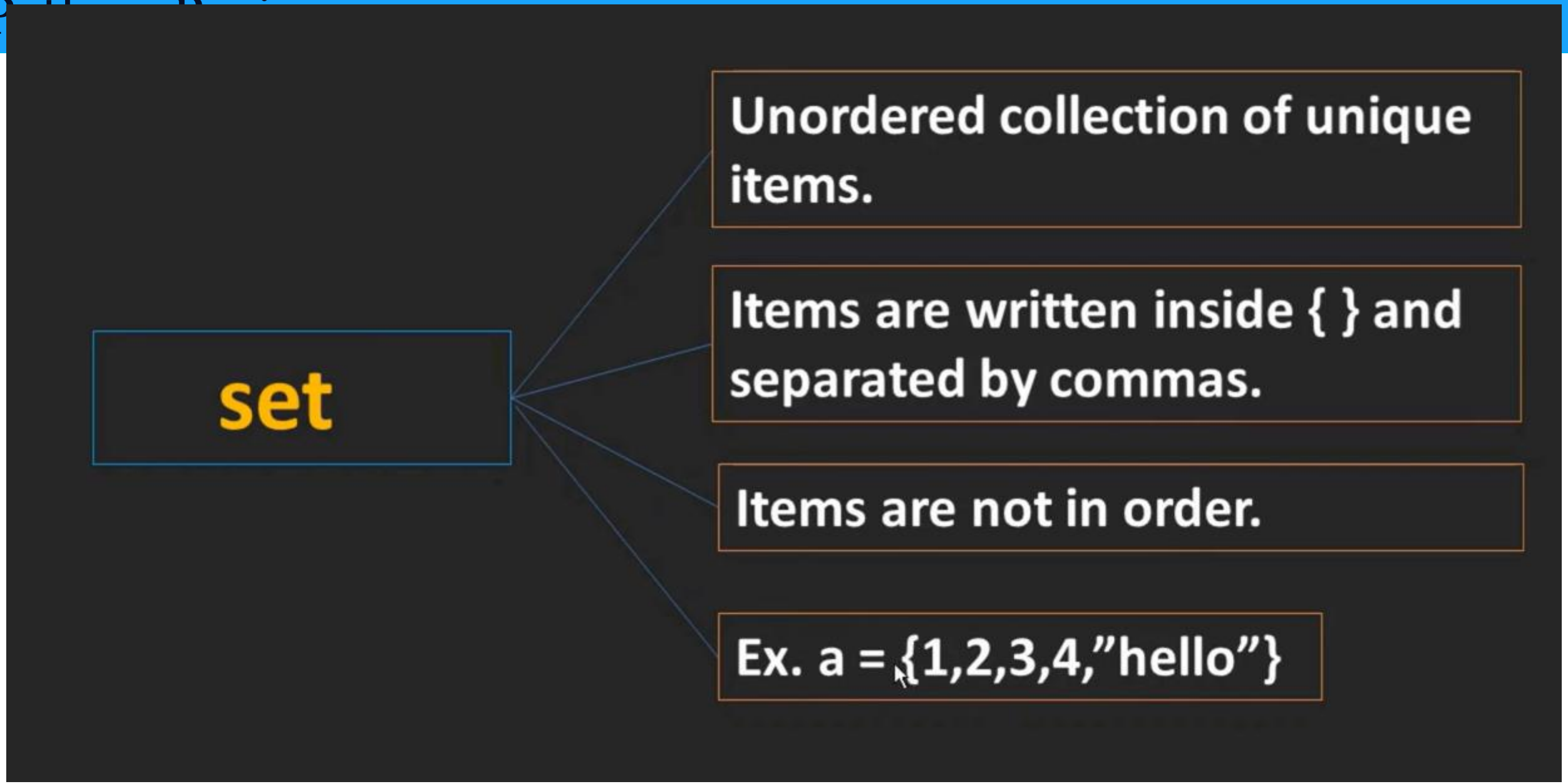


Python-Basics

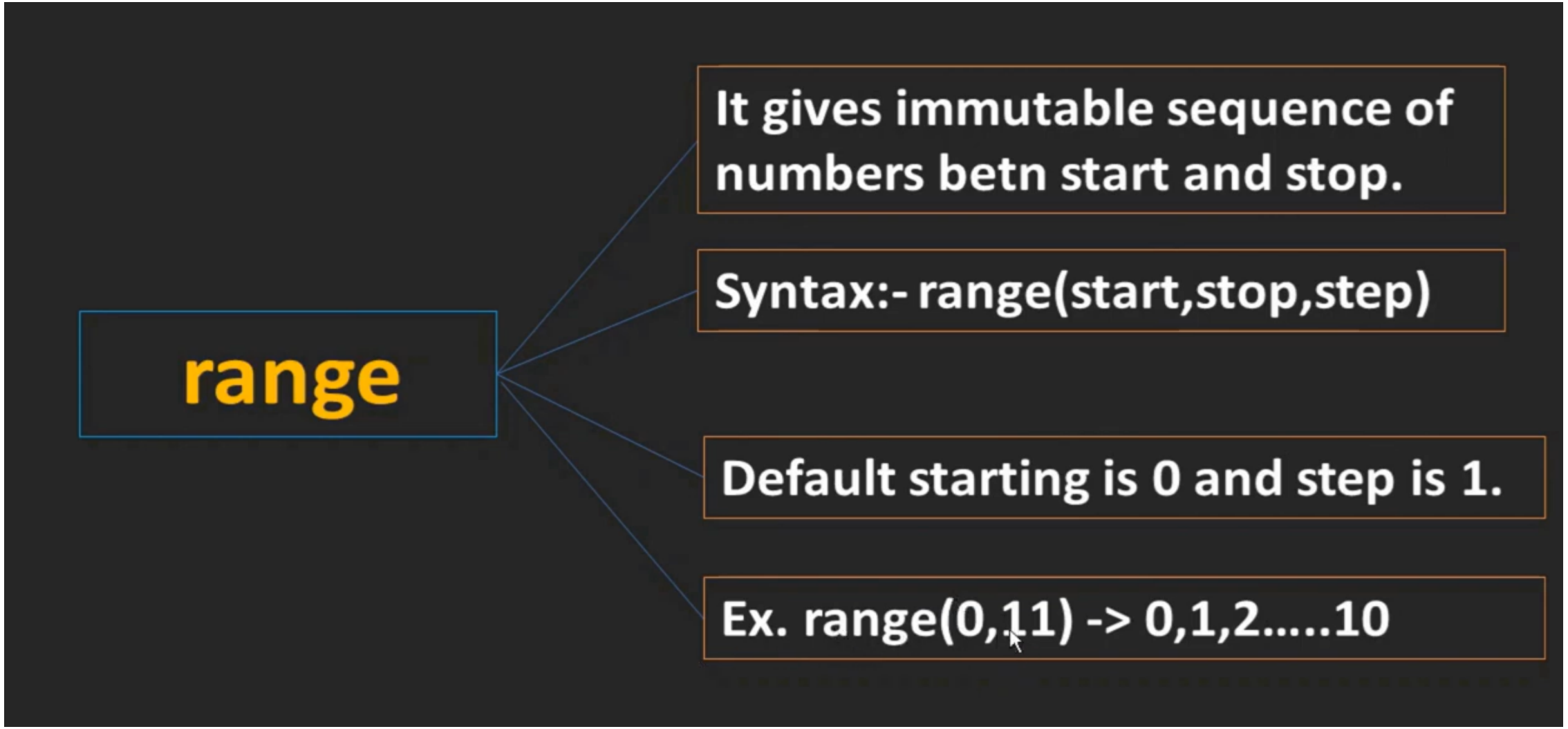


Python-Basics

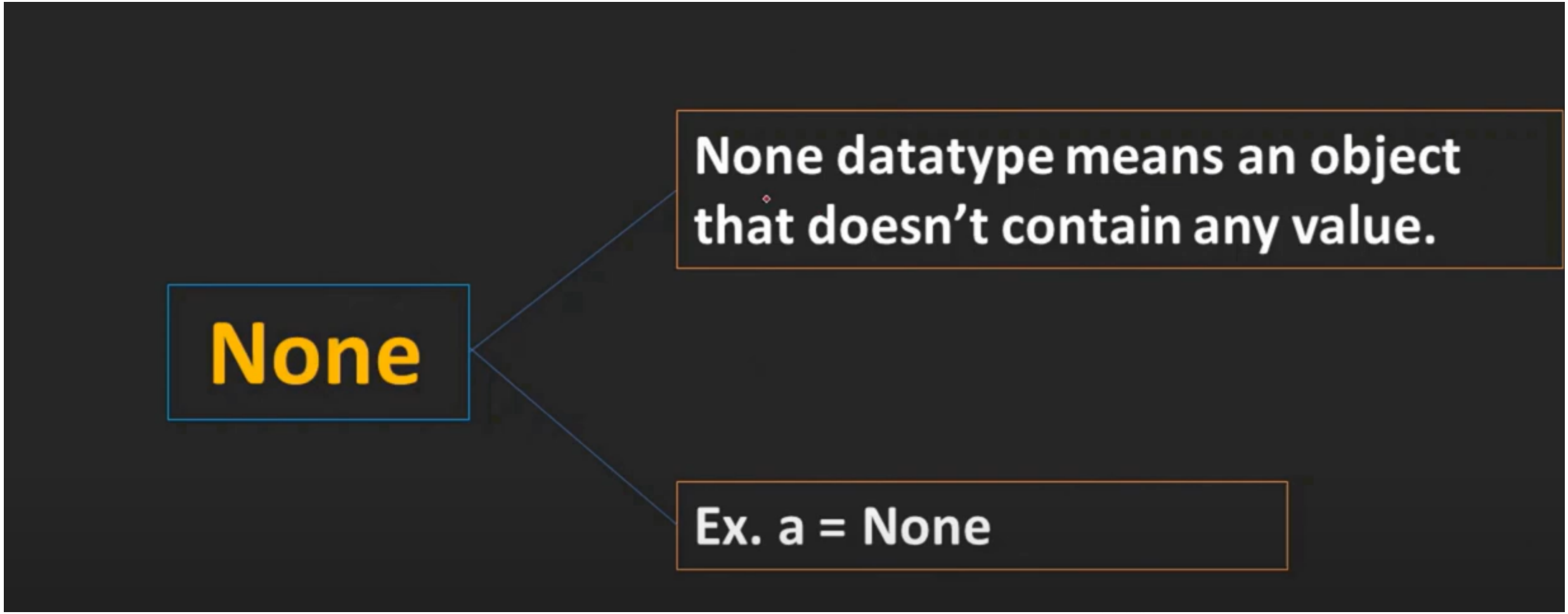




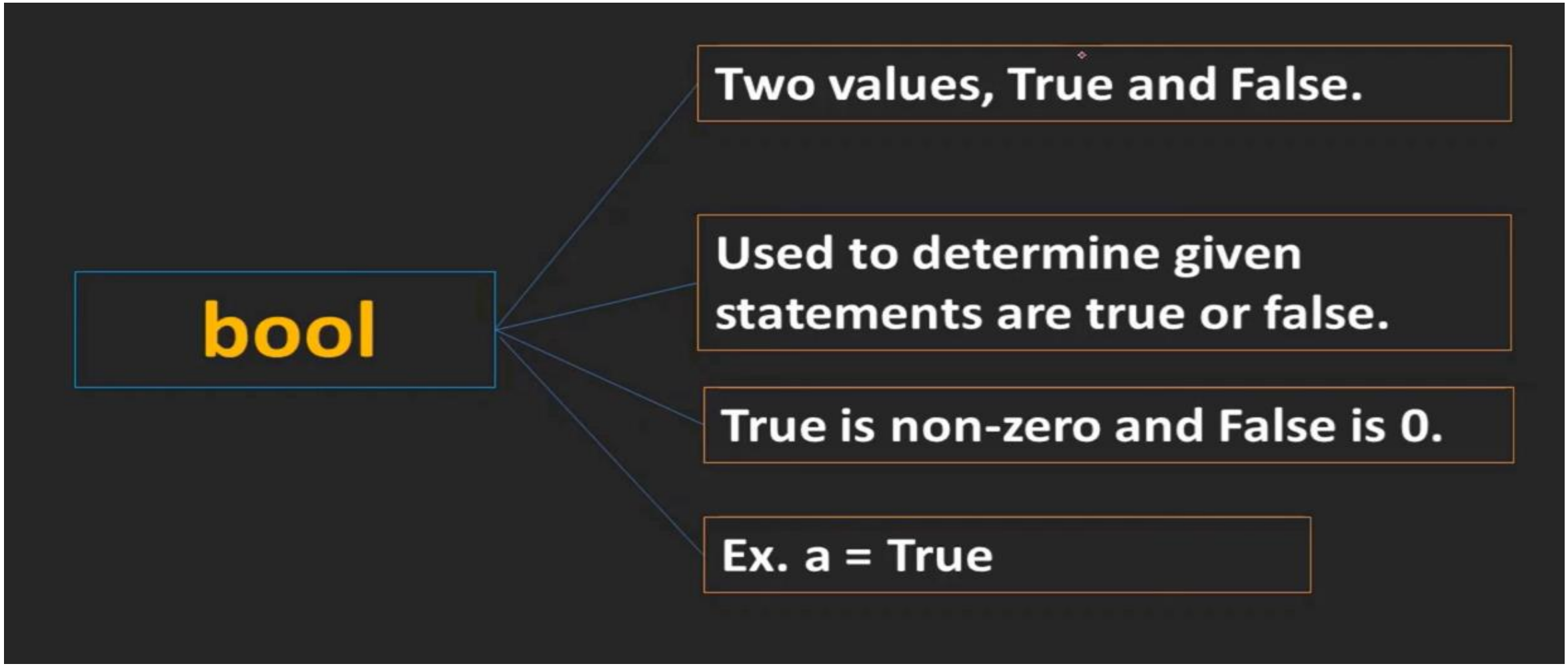
Python-Basics

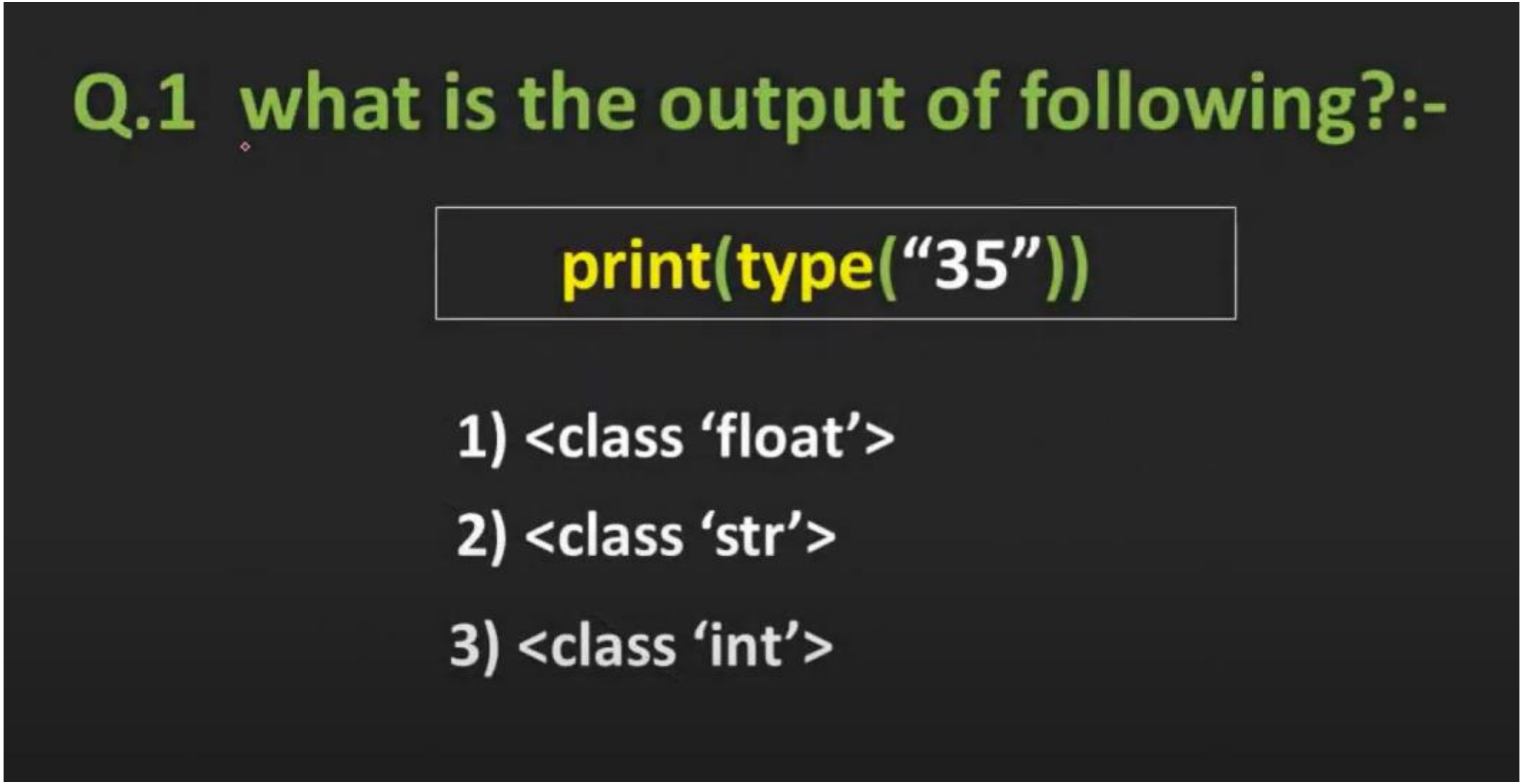


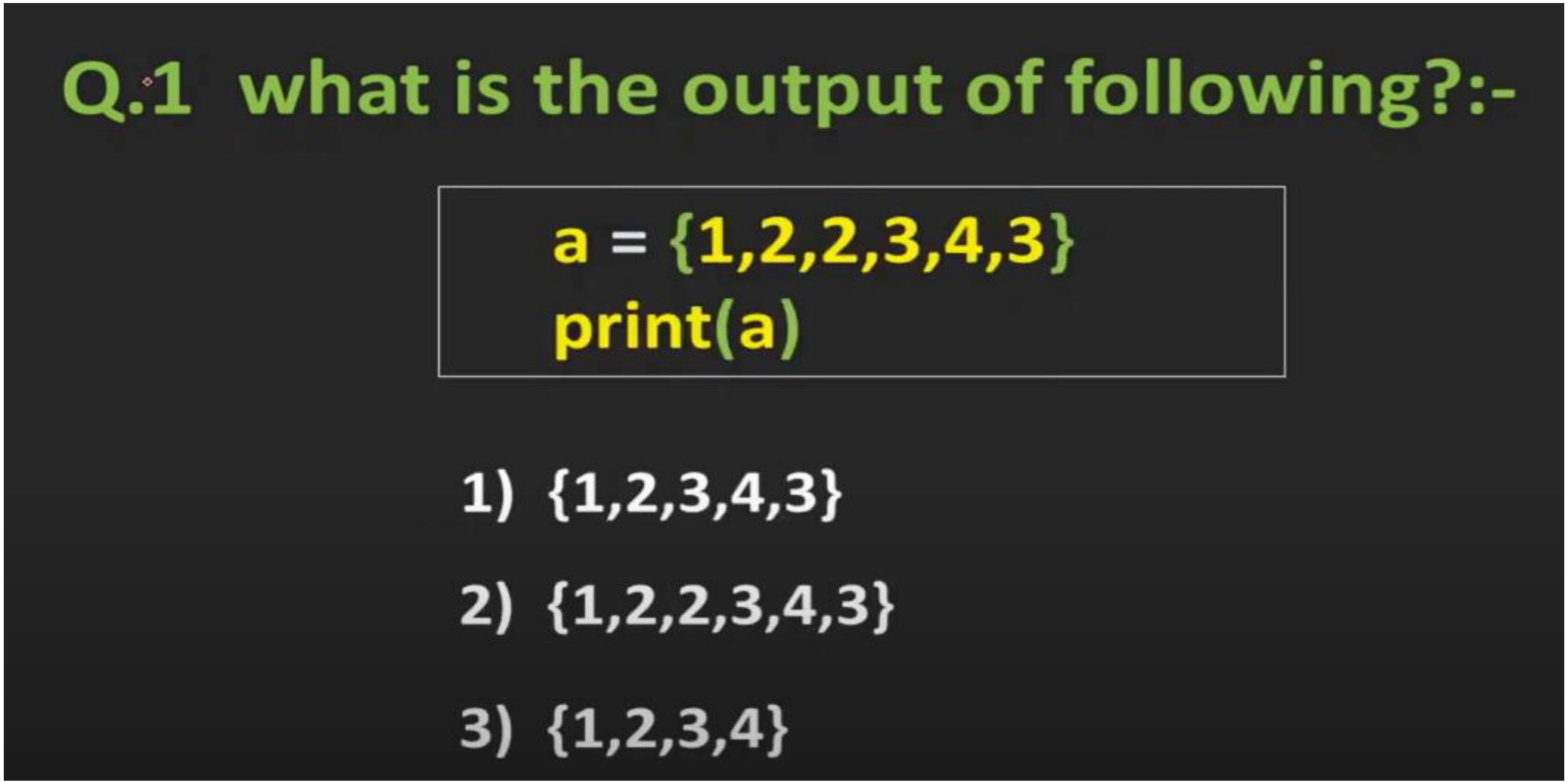
Python-Basics

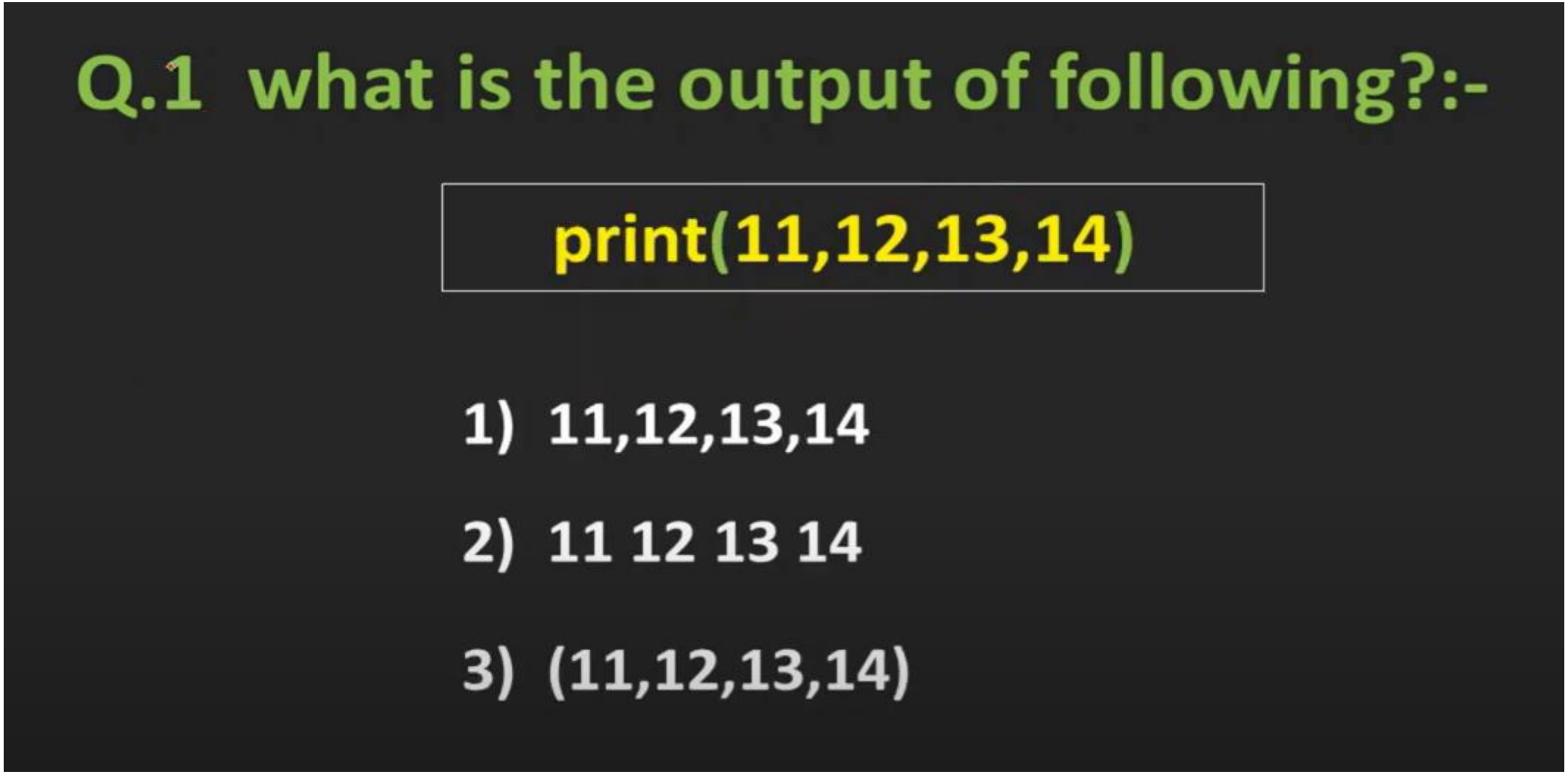


Python-Basics





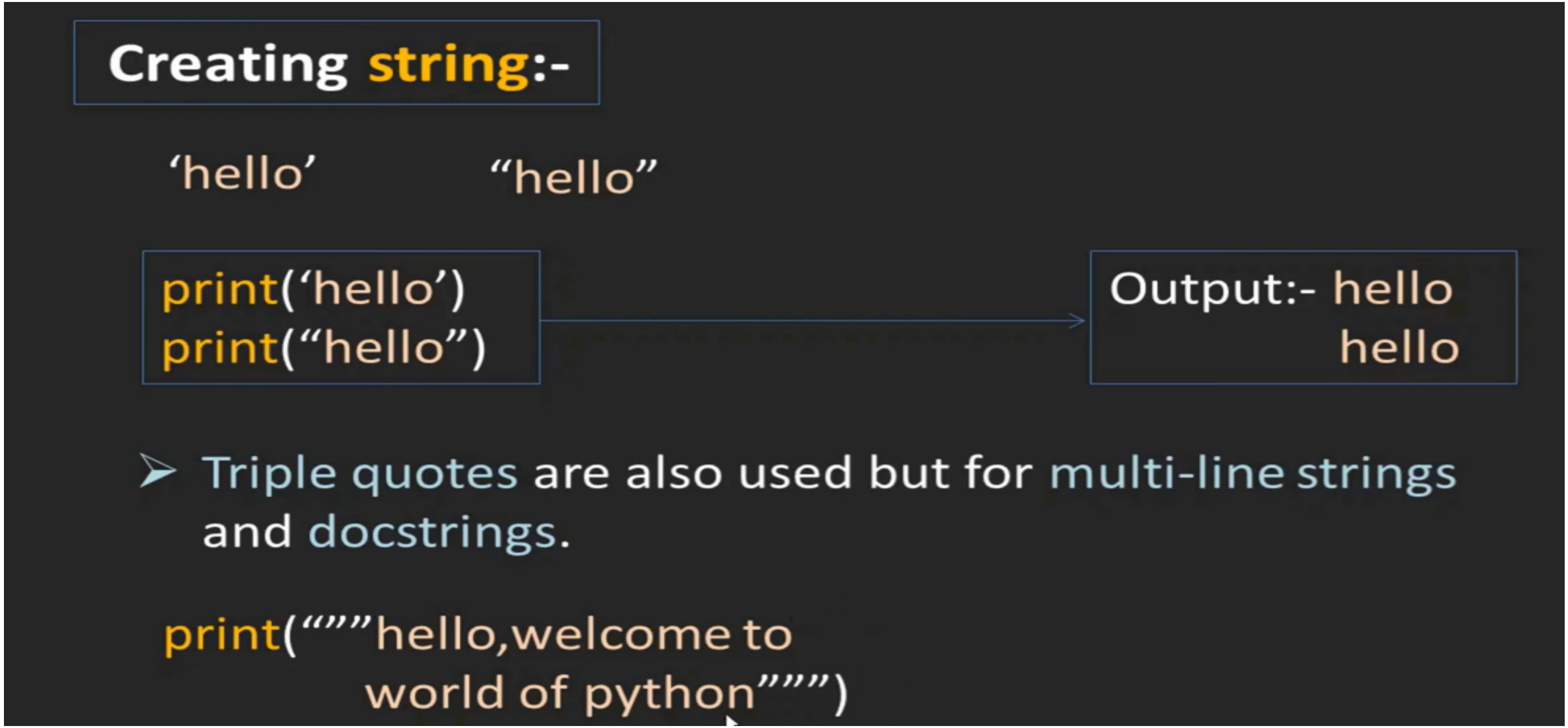




String in Python:-

- A string is a sequence of characters. Ex. “shantanu”
- Anything enclosed inside single quotes ,double quotes except escape sequences is string.
- String is a sequence of unicode characters.

Python-Basics



Accessing substring (characters) in string:-

Two ways:-

1) Indexing
2) Slicing

Indexing:- used to access individual(single) element of string.

slicing:- used to access group of elements of a string. We can access range of characaters using slicing.

Accessing substring (characters) in string:-

Two ways:-

1) Indexing
2) Slicing

Indexing:- used to access individual(single) element of string.

slicing:- used to access group of elements of a string. We can access range of characaters using slicing.

Python-Basics

Index:-

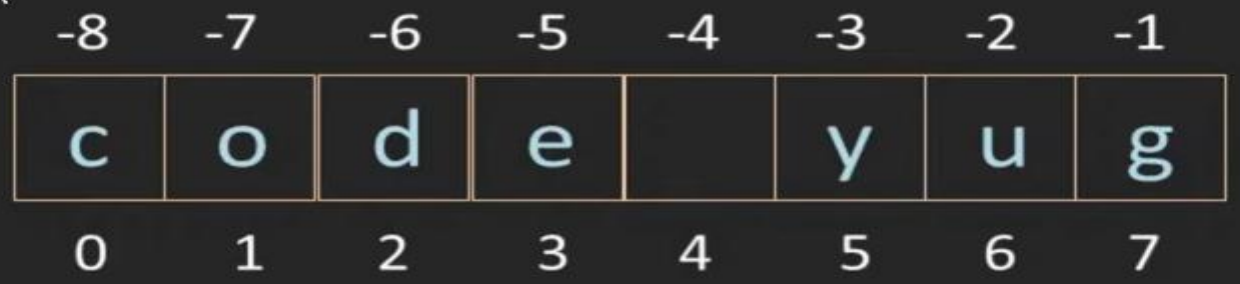
- In python,Every element is represented by index.

Name = “code yug”

Accessing Single Element:-

- To access single element,python uses indexing.

Name = “code yug”



Syntax:- **var_name[index]**

Name[0] ---> c Name[2] ---> d Name[5] ---> y

Name[-1] ---> g Name[-4] ---> Name[-8] ---> c

Python-Basics

slicing:-

- To access a range of items (sub-string),we will use 'slicing'.
- You can return range of characters using slicing.
- [:] is a slice operator.

Syntax:-

```
Var_name[start:stop:step]
```

- start :- starting point of substring.
- stop :- It indicated end of string.
- step :- difference between indexing.

Python-Basics

important:-

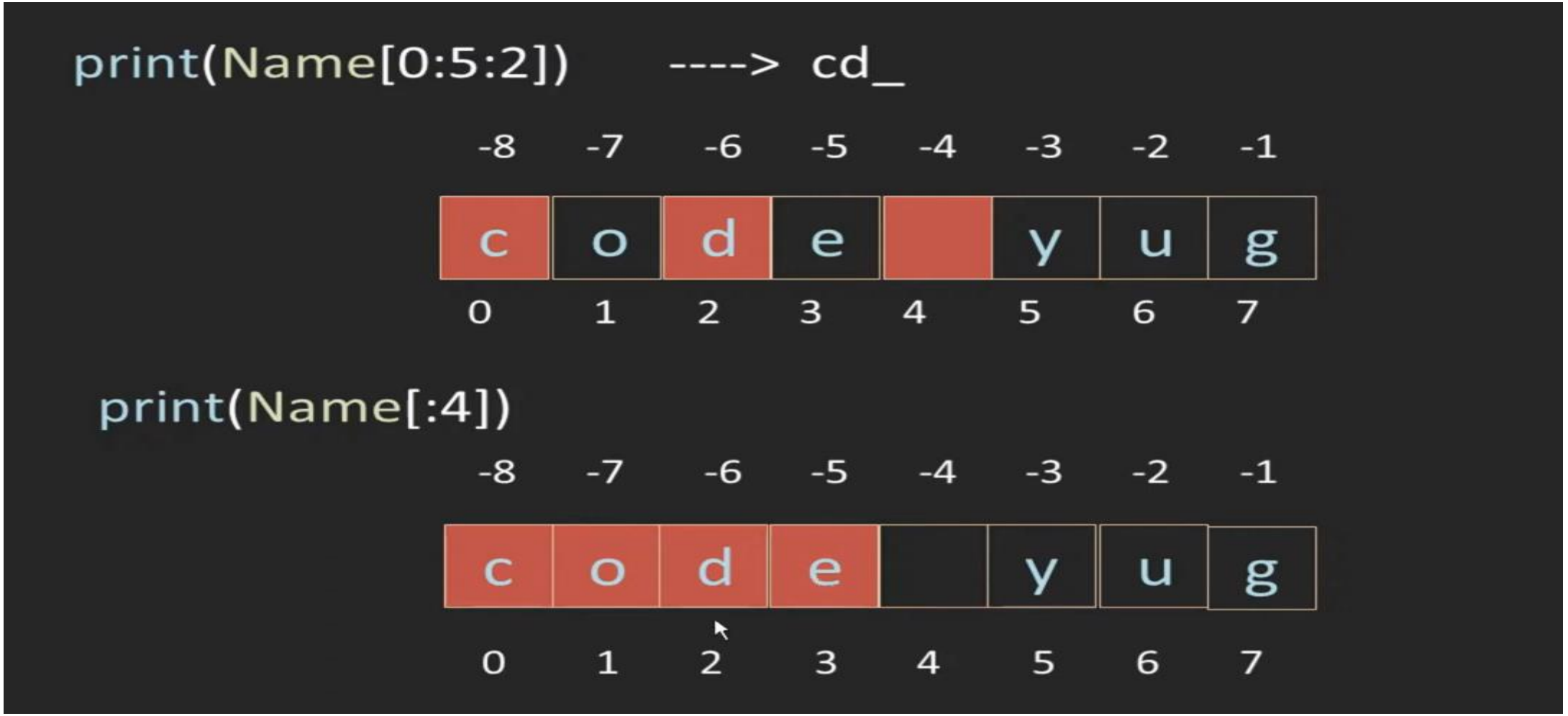
- start,stop,step are optional.
- Default start is index 0.
- Default step is 1 .
- end = stop -1
- Character at stop index is not printed.
- Positive index :- Go in forward direction
  Negative index :- Go in Negative direction
- If stop not found,nothing is returned.

Python-Basics

print(Name[0:3:1]) ----> cod

print(Name[2:4]) ----> de

Python-Basics



Python-Basics

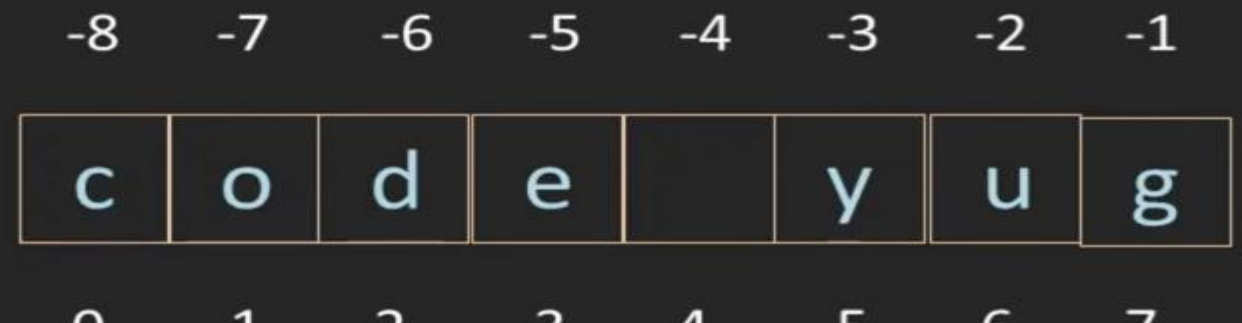
print(Name[::]) ----> code yug

print(Name[0::3]) ----> ceu

Python-Basics

print(Name[-1:-4:-1]) ----> guy

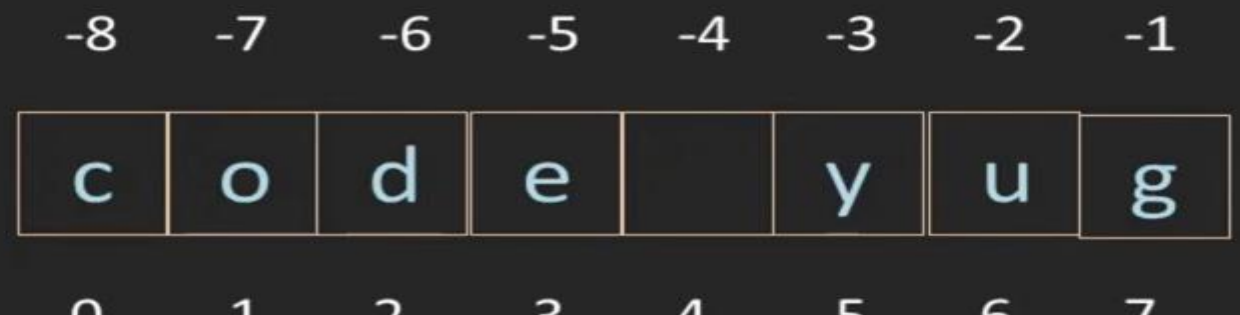
print(Name[-1:-5:-2]) ----> gy



Python-Basics

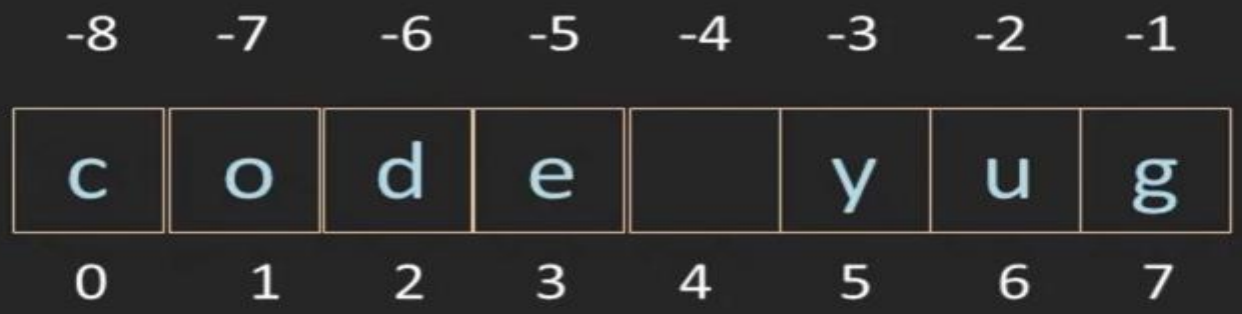
```
print(Name[-2:-6:2])
```


```
print(Name[-1: :-1])
```



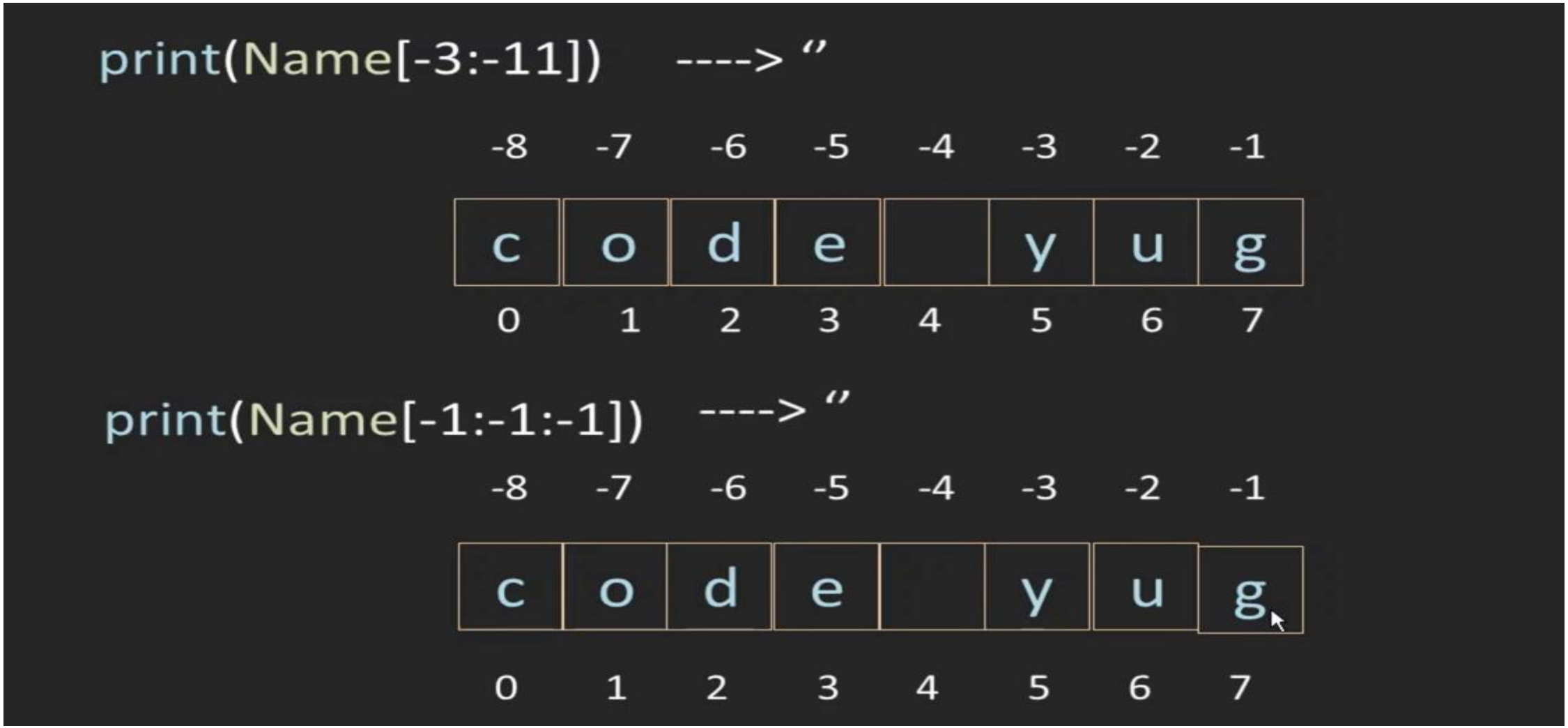
Python-Basics

print(Name[-2:-6:2]) ----> ''



print(Name[-1: :-1]) ----> guy edoc

Python-Basics



Python-Basics- Q1

```
string = "Hello, world!"
print(string[7:12])
```

a) world b) world! c) , worl d) , world

Python-Basics- Q1

```
string = "Hello, world!"
print(string[7:12])
```

a) world b) world! c) , worl d) , world

Python-Basics- Q2

```
string = "Hello, world!"
print(string[:5])
```

a) Hello b) Hello, c) , world! d) world!

Python-Basics- Q2

```
string = "Hello, world!"
print(string[:5])
```

a) Hello b) Hello, c) , world! d) world!

Python-Basics- Q3

```
string = "Hello, world!"
print(string[-6:-1])
```

a) world b) world! c) , worl d) , world

Python-Basics- Q3

```
string = "Hello, world!"
print(string[-6:-1])
```

a) world b) world! c) , worl d) , world

Python-Basics- Q4

Which of the following is the correct way to slice a string to get the last 3 characters?

a) string[-3:] b) string[3:] c) string[:-3]
b) d) string[-3:-1]

Python-Basics- Q4

Which of the following is the correct way to slice a string to get the last 3 characters?

a) string[-3:] b) string[3:] c) string[:-3]
b) d) string[-3:-1]

Python-Basics- Q5

```
string = "Hello, world!"
print(string[-5:2:-2])
```

a) ol b) ow c) ,ol d) dlro

Python-Basics- Q6

```
string = "Python is fun!"
print(string[::2][::-1])
```

Python-Basics- Q7

Which of the following slices would extract the substring "world" from the string "Hello, world!"?

a) string[7:12] b) string[-6:-1] c) string[7:-1] d) All of the above

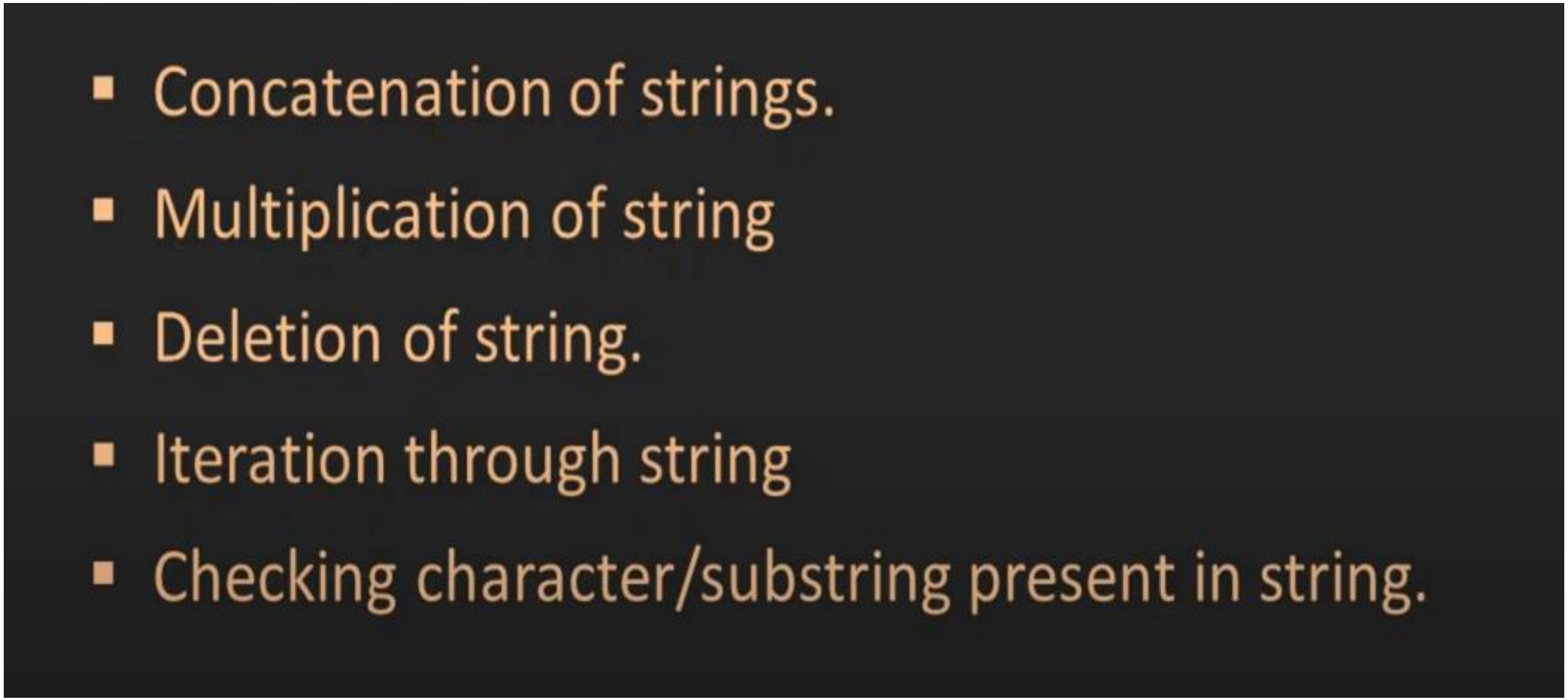
Python-Basics- Q8

```
string = "Programming“

print(string[3:len(string) - 4])
```

Python-Basics- Q9

```
S = [1,2,3,4,5,6,7,8,9]

print(s[:-4:-1])
```

Python-Basics- Q9



Python-Basics- Q9

concatenation:-

- Joining two or more strings into a single one.
- It adds strings together.
- The '+' operator does this.
- Example:-

```
str1 = 'Python is'
str2 = 'awesome'
str3 = str1 + str2
print(str3)
```

Output:-

Python isaweson

Python-Basics- Q9

.

Multiplication:-

- Repeat same string for multiple times.
- Using '*'
- Example:-

```
str1 = 'codeyug'
str2 = str1*2
print(str2)
```

Python-Basics- Q9

Deletion:-

- We cannot delete characters of string.But,deleting entire string is possible using ‘del’ keyword.
  - Example:-

```
str1 = ‘codeyug’
del str1
prints (str1)
```

Python-Basics- Q9

- Iteration on string
- Finding length of string
- len() function
- String membership
- Finding number of vowels in entered string.