



- V
- V
- V

```
VAR_CID = 100;
```

The diagram illustrates the components of a variable assignment in Python. A code box at the top contains the text 'VAR\_CID = 100;'. Two blue arrows originate from this box: one points from 'VAR\_CID' to a label 'Actual variable name' in a blue arrow, and another points from '100' to a label 'Value for the variable' in a blue arrow.

Value for the variable

Actual variable name

In Python, the data type of a variable is automatically determined by its values

```
VAR_CID = 100
```

```
INT
```

```
VAR_f = 100.13
```

```
FLOAT
```

```
VAR_s = "JOTHI"
```

```
STRING
```

```
VAR_b = True
```

```
Boolean
```

```
VAR_li = 12345678
```

```
LONG INT
```

```
Create table EMP
```

```
(
```

```
  EID INT
```

```
  ENAME VARCHAR(50)
```

```
)
```

```
vAR_a = range(3, 20, 2)
```

This variable will have range of value

Start

Increment

End

3, 5, 7, 9, ..., 19

```
vAR_str = "JOTHI"
```

```
Print (vAR_str)
```

This prints the whole string

How string stored in Python

J	O	T	H	I
---	---	---	---	---

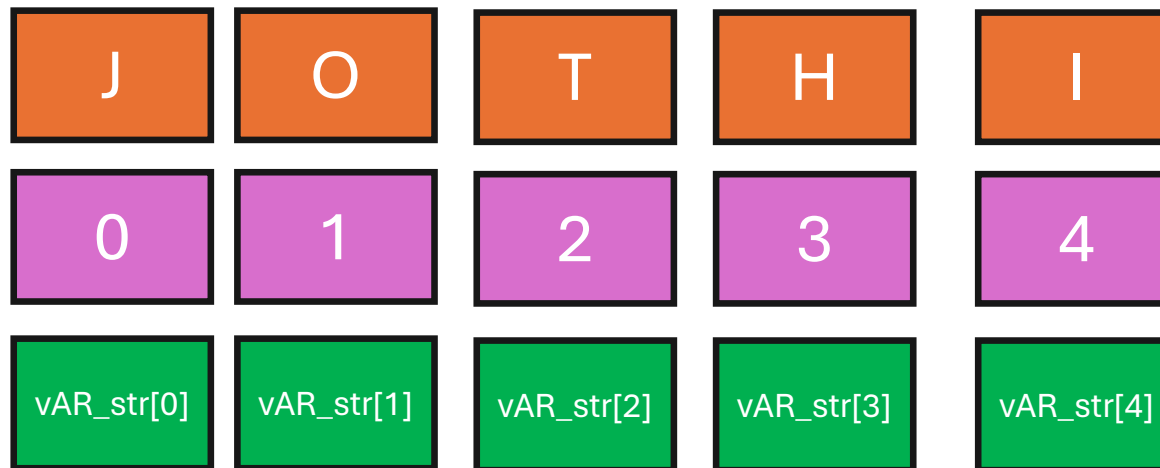
How each position identified in Python

0	1	2	3	4
---	---	---	---	---

```
vAR_str = "JOTHI"
```

```
Print (vAR_str)
```

This prints the whole string



```
vAR_str[0]
```

```
vAR_str[1]
```

This is without any range

```
vAR_str[:2]
```

This is with range

From the 0<sup>th</sup> position, we need to go to 2 characters



# Python Data Structure

```
# () - Tuple  
# [] - List  
# {} - Dictionary  
# {} - Set
```

Variable is  
different and  
data structure  
is different

A data structure allows us to store and process data in memory with defined characteristics

Ex: I want to store 100 employee record for payroll processing

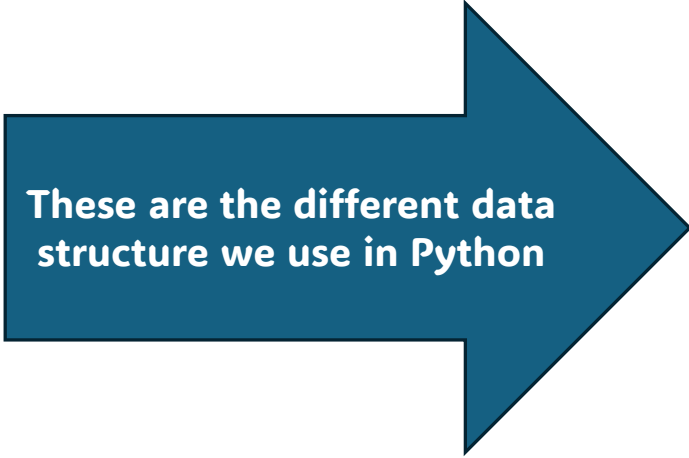
Variable

A column in a table

Data structure

A record in a table

Collection of columns



**These are the different data  
structure we use in Python**

# () – Tuple

# [] – List

# {} – Dictionary

# {} – Set

# What is the difference between each one and when to use what?

	Tuple	List	Set	Dictionary
Order	YES	NO	YES	NO
Duplicate				
Change				YES

Characteristics of the data structure

- ✓ Is this data in the data structure ordered
- ✓ Can this data structure have duplicates values
- ✓ Can we change the data structure values

```
# () - Tuple  
# [] - List  
# {} - Dictionary  
# {} - Set
```

When to use what, is driven based on the business requirements and the characteristics of the data structure

```
VAR_list = [ 'abcd', 786 , 2.23, 'Jothi', 70.2 ]
```

A data structure of  
type LIST

This is a list of values of type anything (int, float, str, boolen)

```
var_list = [ 'abcd', 786 , 2.23, 'Jothi', 70.2 ]
```

0

1

3

5

4

List Index

You can access the each members  
using the Index position



```
vAR_list = [ 'abcd', 786 , 2.23, 'Jothi', 70.2 ]
```

0

1

3

5

4

List Index

You can access the list members one at a time (0, 1, 2, etc.)  
access them as a range

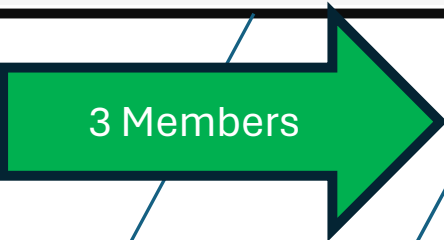
```
vARList=[[1, 2, 3, 4, 5], [12, 13, 23], [10, 20, 30], [11, 22, 33], [12, 24, 36]]
```



## A List of List

- The members in each list could be any type and any numbers

```
vARList=[[1, 2, 3, 4, 5], [12, 13, 23], [10, 20, 30], [11, 22, 33], [12, 24, 36]]
```



HW:

Check this out , in the List is there any option to sort the list members in DESC order , by default it's in ASC order

```
vAR_List = [ 10,20, "JOTHI", 11.11 ]
```

```
vAR_tuple = ( 'abcd', 786 , 2.23, 'Jothi', 70.2, "apple", "banana", "cherry" )
```

A tuple can also holds any type of data like  
a list

```
vAR_tuple = ( 'abcd', 786 , 2.23, 'Jothi', 70.2, "apple", "banana", "cherry" )
```

The Tuple members indexed like a LIST, it's starts from  
“0”, and then it goes from there