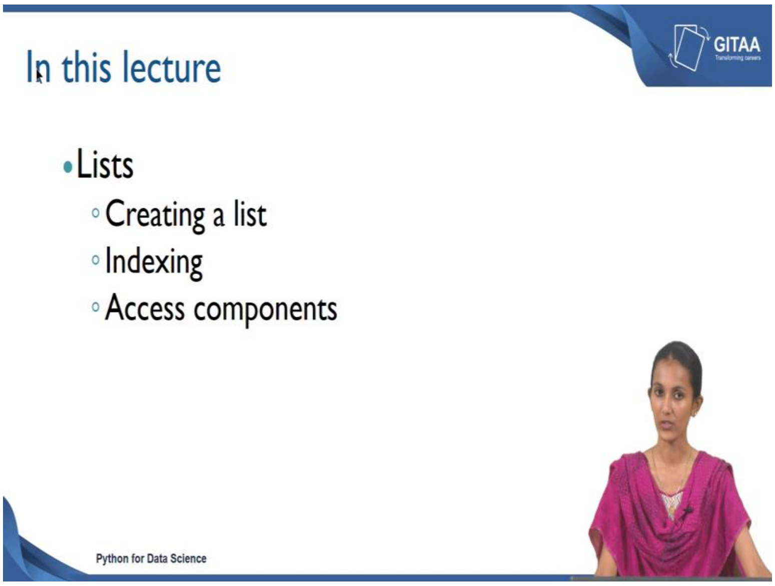
Python for Data Science
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture: 07
Lists Part – 1

(Refer Slide Time: 00:17)

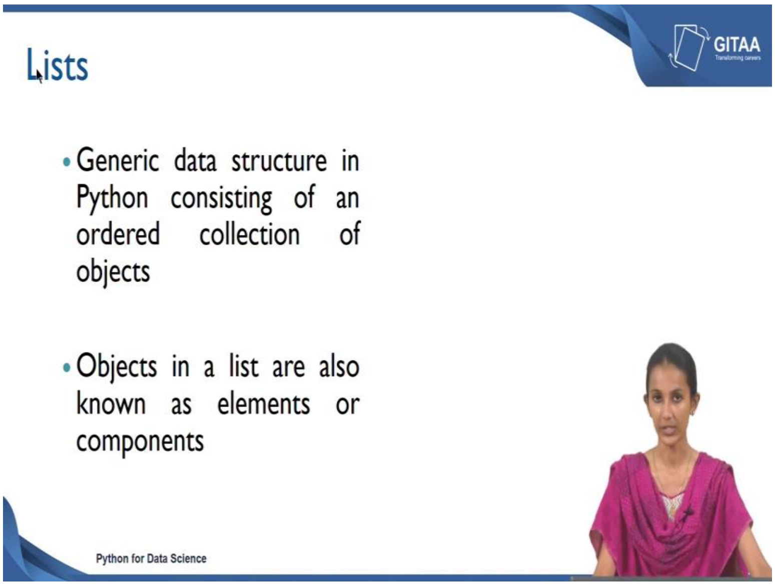


In this lecture

- Lists
  - Creating a list
  - Indexing
  - Access components

Welcome to the lecture. In this lecture, we will see what is mean by Lists, how to create a list, indexing and also the accessing components of the list.

(Refer Slide Time: 00:27)

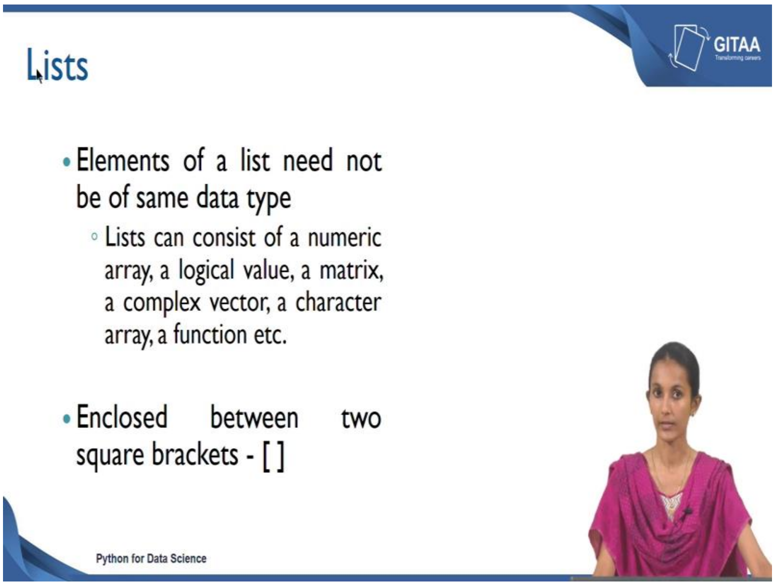


Lists

- Generic data structure in Python consisting of an ordered collection of objects
- Objects in a list are also known as elements or components

First let us look what is mean by list? List is one of the data structure, consisting of ordered collection of the objects. Objects in a list it can also be called as an elements or also components of the list.

(Refer Slide Time: 00:41)



Lists

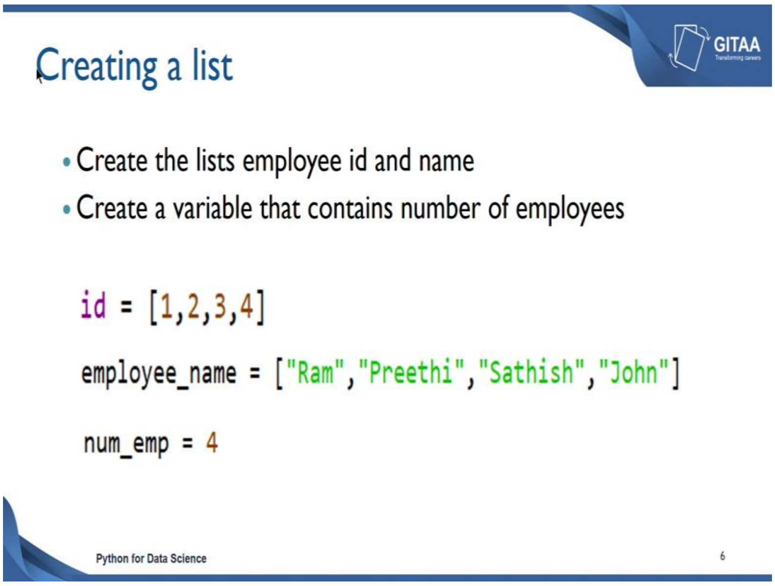
- Elements of a list need not be of same data type
  - Lists can consist of a numeric array, a logical value, a matrix, a complex vector, a character array, a function etc.
- Enclosed between two square brackets - [ ]

Python for Data Science

GITAA
Transforming careers

So, elements it need not be a same data type, it can also be a different data type. So, it can be of an array, vector, numeric matrix, logical value, a complex vector etc. Elements inside the list is enclosed between two square brackets.

(Refer Slide Time: 00:59)



Creating a list

- Create the lists employee id and name
- Create a variable that contains number of employees

```
id = [1,2,3,4]

employee_name = ["Ram","Preethi","Sathish","John"]

num_emp = 4
```

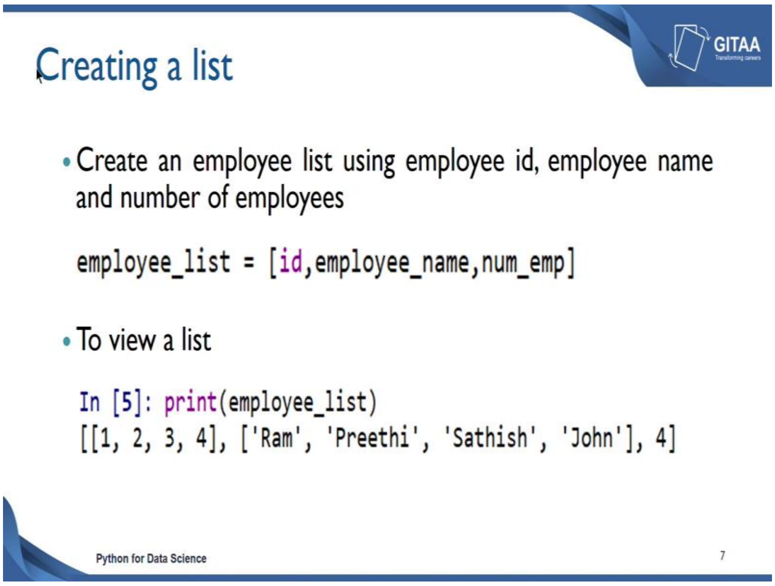
Python for Data Science 6

GITAA
Transforming careers

Let us see how to create a list. So, we will create the list for employee id and also for employee name and also we will create a variable that contains the number of employees.

I am creating a list call id which has a set of values 1, 2, 3, 4 within inside the square bracket separated by commas. We will also create an employee_name as a lists, so which consists of values Ram, Preethi, Sathish and John which is separated by commas. And also we will create a variable call num_emp which has a values of the number of employees.

(Refer Slide Time: 01:39)



Creating a list

- Create an employee list using employee id, employee name and number of employees

```
employee_list = [id,employee_name,num_emp]
```

- To view a list

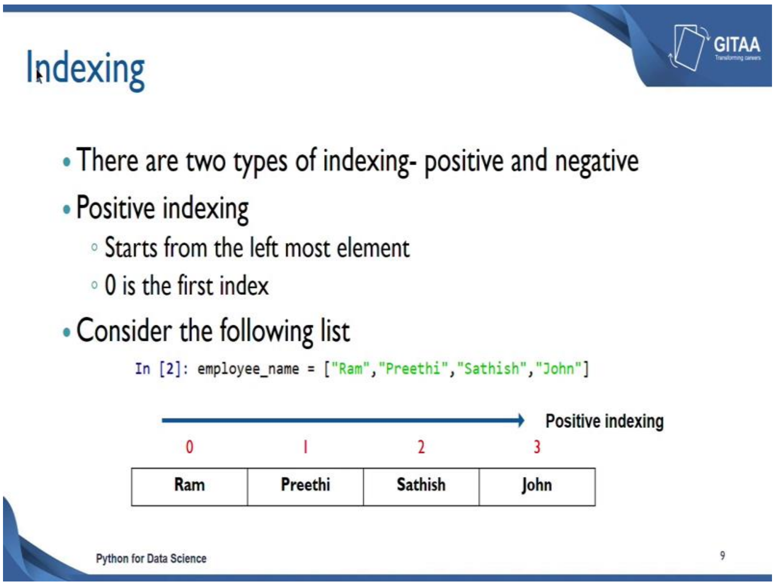
```
In [5]: print(employee_list)
[[1, 2, 3, 4], ['Ram', 'Preethi', 'Sathish', 'John'], 4]
```

Python for Data Science

7

Now, let us see how to create an employee list using this employee id, employee name and also the number of employees. So, I will create a list employee_list which has values, id, employee_name and also the number of employees inside a square brackets. Let us see how to view the list; we will use the command; print(employee_list) it gives us a list which has a values; the id values and also the employee names and also the number of employees under the employee_list.

(Refer Slide Time: 02:19)



Indexing

- There are two types of indexing- positive and negative
- Positive indexing
  - Starts from the left most element
  - 0 is the first index
- Consider the following list

```
In [2]: employee_name = ["Ram","Preethi","Sathish","John"]
```

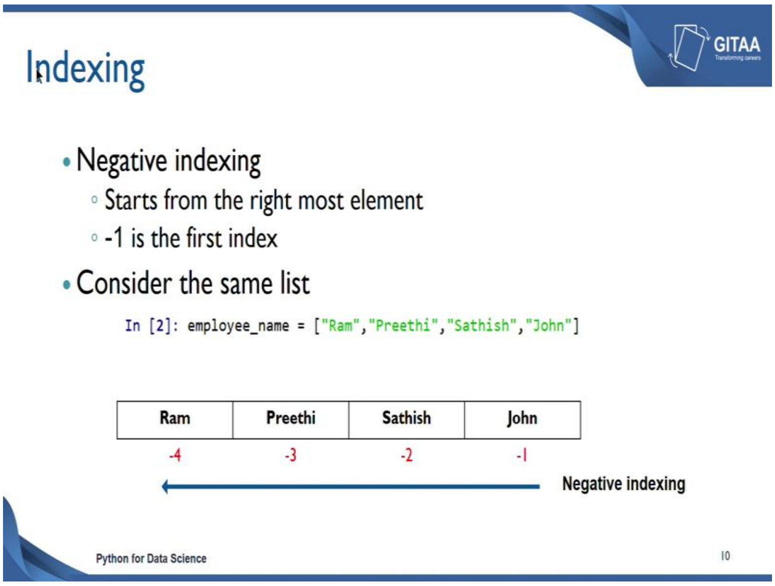
Positive indexing

0	1	2	3
Ram	Preethi	Sathish	John

Now, let us see the indexing in python; there are two types of indexing one is positive and another one is negative indexing. So, first let us look at the positive indexing. So, it always starts from the left most element to the right most and 0 is the first index and it ranges till n -1. So, let us take an example; so we will considered the list employee_name which has a employee name which is Ram, Preethi, Sathish and John. Ram will have the index value of 0, Preethi will have the indexing value of 1, Sathish will have the indexing value of 2, John will have the indexing value of 3.

So, if you run from the leftmost to right most; so that is a positive indexing. So, it always starts from 0 till n -1.

(Refer Slide Time: 03:09)



Indexing

- Negative indexing
  - Starts from the right most element
  - -1 is the first index
- Consider the same list

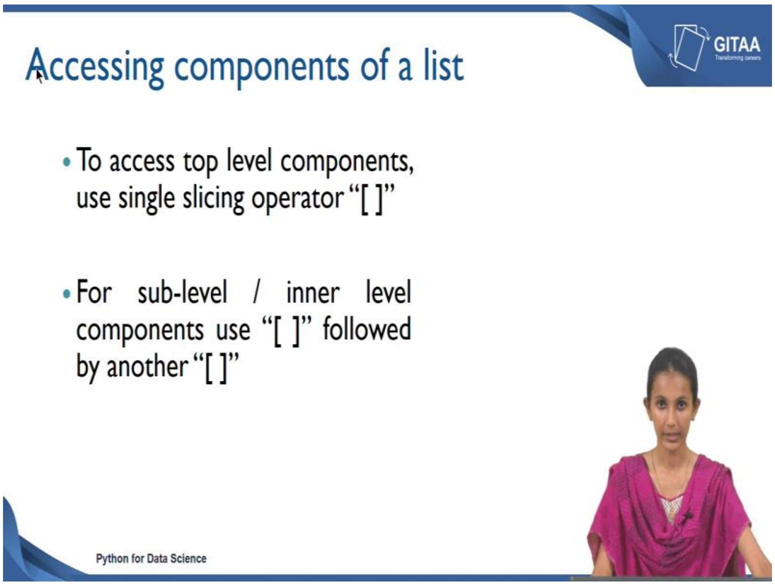
```
In [2]: employee_name = ["Ram","Preethi","Sathish","John"]
```

Ram	Preethi	Sathish	John
-4	-3	-2	-1

Next we look at the negative indexing. So, negative indexing it starts from the rightmost element and -1 will be the first index. So, we will again consider the same list which is a employee_name. So, again it has Ram, Preethi, Sathish and John. Now for the negative indexing, so it starts from the rightmost element John. So, John will have the index value of -1, Sathish will have the index value of -2, Preethi will have the index value of -3, Ram will have the index value of -4.

So, this is the negative indexing; so it starts from the rightmost element -1 and it goes on.

(Refer Slide Time: 03:49)



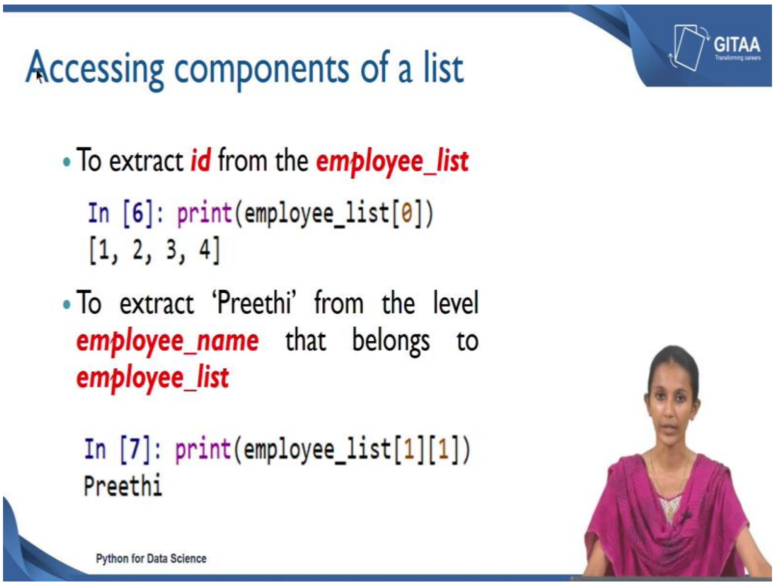
Accessing components of a list

- To access top level components, use single slicing operator "[ ]"
- For sub-level / inner level components use "[ ]" followed by another "[ ]"

Now, let us see how to access a components of the list. So, there are two components inside the list; so, one is top level component and other one is sub level components. So, if you want it acts as a top level components in the list. So, leave use single slicing operator, so which is will have a square brackets. So, if you wanted to access the sub-level components or the inner level components, use square brackets followed by the another square brackets.

So, if you wanted to access id from the employee_list. So, our id is a top level component and employee_name is a top level component and also the num_emp is our top level components.

(Refer Slide Time: 04:27)



Accessing components of a list

- To extract *id* from the *employee_list*

```
In [6]: print(employee_list[0])
[1, 2, 3, 4]
```

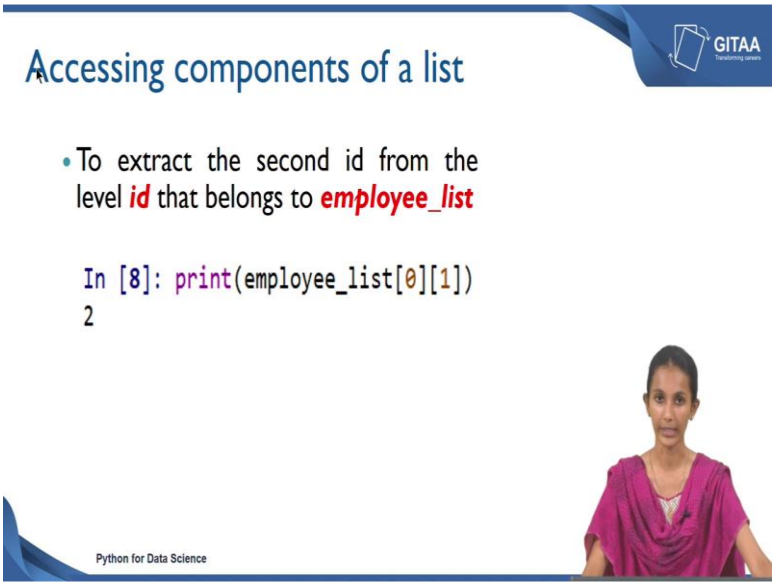
- To extract 'Preethi' from the level *employee_name* that belongs to *employee_list*

```
In [7]: print(employee_list[1][1])
Preethi
```

So, if I wanted to extract id from the employee_list; so, the command is employee_lists followed by the index numbers. Since, id is our top level component; so the it has a index value of 0. So, if you give employee_lists 0, so it will be able to displays elements in the id.

So, if I wanted to extract Preethi from the employee name; so which belongs to the employee_list; so then print employee_list followed by the index value of the top level component and followed by the index value of the sublevel component. So, employee_names are top level component which has the index value of 1. The sub level component Preethi is in the sub level which is index value of 1; so it is employee_list 1 of 1.

(Refer Slide Time: 05:27)



Accessing components of a list

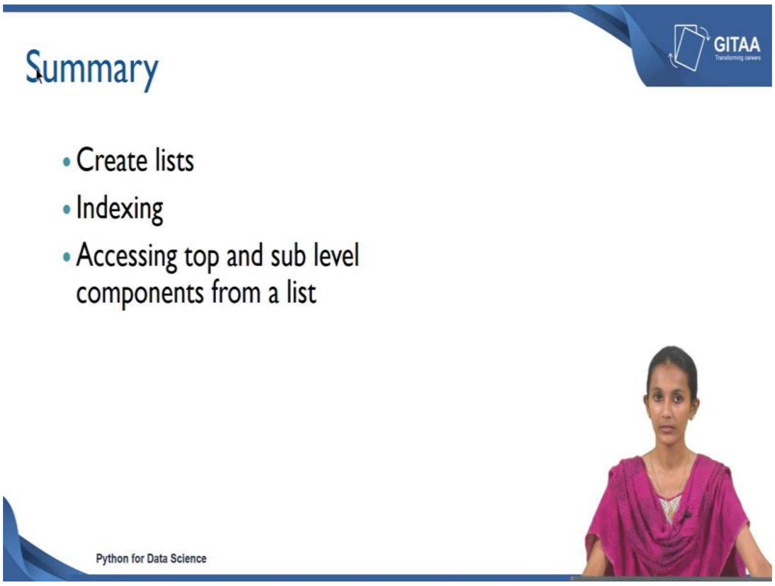
- To extract the second id from the level *id* that belongs to ***employee_list***

```
In [8]: print(employee_list[0][1])
2
```

Python for Data Science

Next, we will see how to extract the particular id from the id which belongs to the employee list. So, if you wanted to extract the second id then print employee_list. So, 0 is our top level component followed by the 1, which is our sub level component you will be able to extract the second id from the employee_list.

(Refer Slide Time: 05:53)



Summary

- Create lists
- Indexing
- Accessing top and sub level components from a list

Python for Data Science

Now, let us summaries what we have covered in the structure. First, we saw how to create a list and we also saw indexing in python. So, there are two types of indexing one is positive indexing and another one is negative indexing. Positive indexing, it always starts from the left most elements; so it has start with an index of 0. So, the negative indexing, it starts from the rightmost element with the index value of -1. We also saw how to access the top level component and also the sub level components in a list.

Thank you.