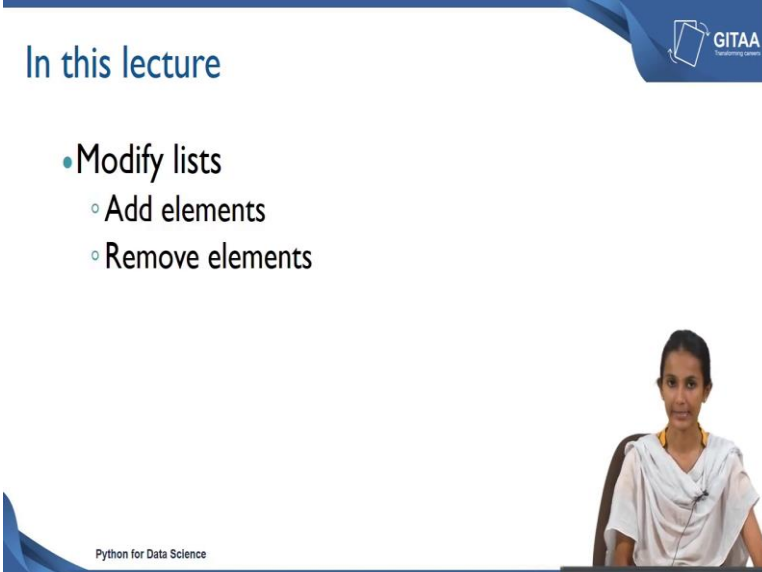


Python for Data Science
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture - 08
Lists Part - 2

Welcome to the lecture. In the previous lecture, we saw how to create a list and also how to access the elements in the list and also we saw indexing in Python.

(Refer Slide Time: 00:27)



In this lecture

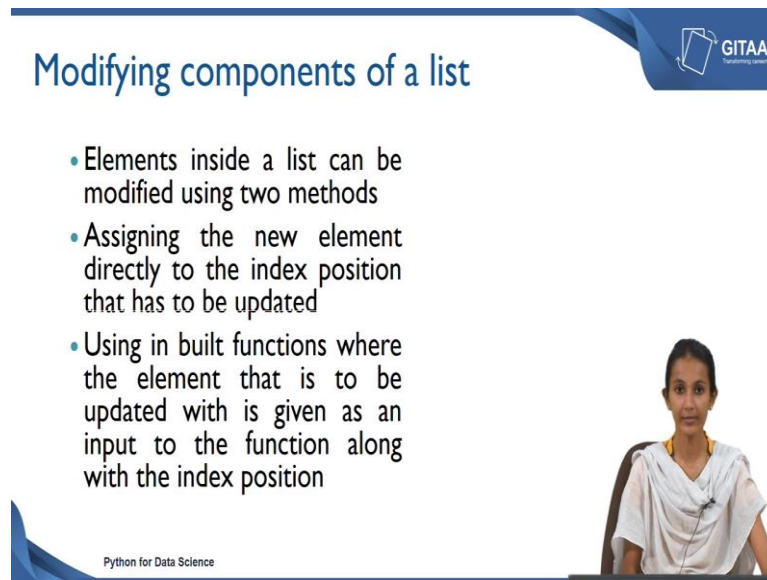
- **Modify lists**
 - Add elements
 - Remove elements

Python for Data Science

GITAA

So, in this lecture, we will see how to modify the lists; so, basically how to add the elements and also how to remove the elements from the list.

(Refer Slide Time: 00:37)



Modifying components of a list

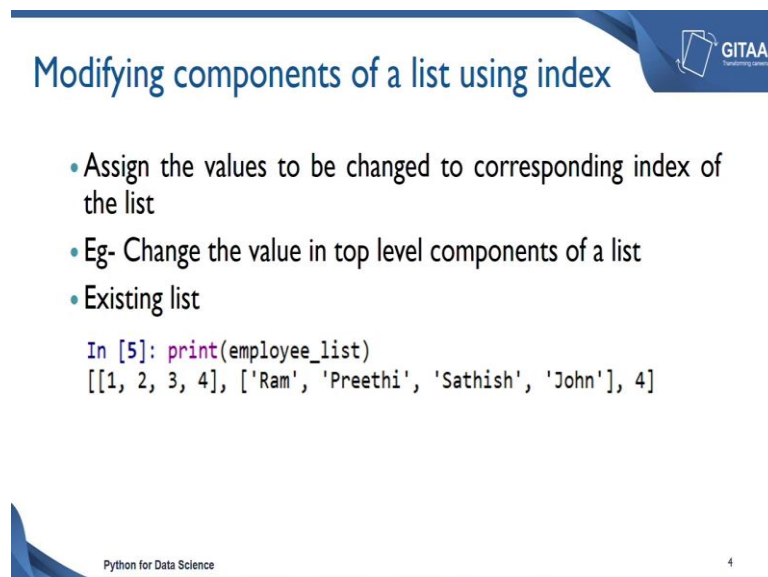
- Elements inside a list can be modified using two methods
- Assigning the new element directly to the index position that has to be updated
- Using in built functions where the element that is to be updated with is given as an input to the function along with the index position

Python for Data Science

GITAA

So, first one is modifying components of a list. So, elements inside the list can be modified using the two methods; the first method is we can assign the new element value based on the index position. So, second one is using the inbuilt functions. So, we can give as an input to the function along with the index value.

(Refer Slide Time: 00:59)



Modifying components of a list using index

- Assign the values to be changed to corresponding index of the list
- Eg- Change the value in top level components of a list
- Existing list

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

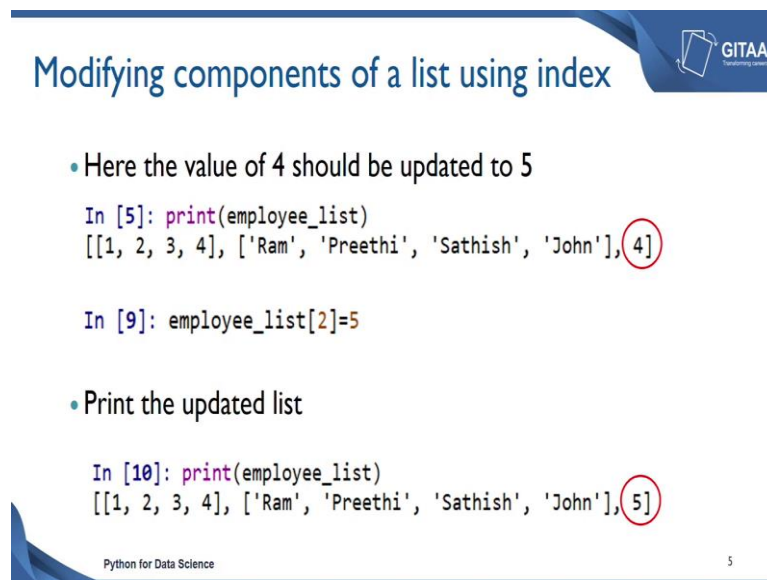
Python for Data Science

GITAA

Modifying components of a list using the index: so, first we will need to assign the values to be changed to the corresponding index of a list. So, there are two levels in a list. So, one is called as a top level component of a list and other one is called as a

sublevel component of a list. So, let us say if I wanted to change the value in top level component of a list. So, this is our existing list so, which we have already created employee_list which has a levels id which is 1, 2, 3 and 4; employee_names Ram, Preethi, Sathish and John and the number of employees which as 4.

(Refer Slide Time: 01:39)



The slide features a blue header with the title "Modifying components of a list using index" and the GITAA logo. The content includes two bullet points and corresponding Python code snippets. The first bullet point states that the value 4 should be updated to 5, followed by a code snippet showing the initial list state where the number 4 is circled in red. The second bullet point instructs to print the updated list, followed by a code snippet showing the list after the update, where the number 5 is circled in red. The footer contains the text "Python for Data Science" and the number "5".

Modifying components of a list using index

- Here the value of 4 should be updated to 5

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

- Print the updated list

```
In [10]: print(employee_list)
[[1, 2, 3, 4], ['Ram', 'Preethi', 'Sathish', 'John'], 5]
```

Python for Data Science 5

Let us say if I wanted to change the value of 4 to 5 so, which is the number of employees. So, we have total number of employees is 4 if I wanted to change to 5. So, then ids stands for the level 0 Ram, Preethi, Sathish, John so, which is employee_names which is at the level of 1 and number of employees at the level of 2.

Employee_list 2. So, I am assigning it to 5. So, if I print the updated list so, 4 will be replaced with 5. So, the encircled ones has been replaced.

(Refer Slide Time: 02:15)

Modifying components of a list using index

- Eg- Change value in sub level components of a list

```
In [10]: print(employee_list)
[[1, 2, 3, 4], ['Ram', 'Preethi', 'Sathish', 'John'], 5]
```

```
In [12]: employee_list[1][3]="Karan"
```

```
In [13]: print(employee_list)
[[1, 2, 3, 4], ['Ram', 'Preethi', 'Sathish', 'Karan'], 5]
```

↑
John has been replaced with Karan

Python for Data Science 6


Let us say if I wanted to change the value in sub level component of a list. So, this is our list. So, if I wanted to change John to Karan, how will you do that? So, 1 2 3 4 which is basically id; again it is a level 0, employee_names is level 1 and number of employees is level 2.

So, if I give employee_list 1 which is our top level component so, our sub level is 3. So, John is placed in the fourth position which has a index number 3 so, I wanted to replace with Karan. So, I am giving inside the double quotes. So, if I print the updated list so, John will be replaced with Karan.

(Refer Slide Time: 03:01)

Modifying components using append()

- **append()** - adds an object at the end of the list
- Syntax: **list_name[index].append(object)**
- In the above syntax if the 'index' is not specified, then the object gets added as a new level in the existing list
- There are two ways to add an object to a list:-
 - Adding an element to a list
 - Adding a list to a list



Python for Data Science

GITAA
Geospatial Information Technology and Analytics

Next we look at the inbuilt functions available in python. So, let us say if I wanted to modify components using the append method. So, what is append does is it is an adds an object at the end of the list. So, we let us look at the syntax of the append; list name within the square brackets, you have to specify the index number dot append and inside the parenthesis you have to specify the object.

So, if the index is not specified the elements, the object gets added at the new level in the existing list. There are two ways to add an object to a list. First is you can add an element to a list, second one you can add a list to a list so, which is called as the concatenation of a list. First we will see how to add an element to a list using the append method.

(Refer Slide Time: 03:53)

Modifying components using append()

- Adding an element to a list
- Adding number '5' to the level **id** in **employee_list**
In [14]: `employee_list[0].append(5)`
- Adding name 'nirmal' to the level **employee_name** in **employee_list**
In [15]: `employee_list[1].append('nirmal')`
- Print the updated list
In [16]: `print(employee_list)`
[[1, 2, 3, 4, 5], ['Ram', 'Preethi', 'Sathish', 'Karan', 'nirmal'], 5]

Python for Data Science 8

Let us say if I wanted to add an element to a list. So, today one person has joined. So, employee id will be increased by 1 and also employee_name will be increased by 1 right.

Let us say if I wanted to add the id 5 in the employee_list. So, employee_list is a list name inside the square brackets, we have to mention the index number which is 0 dot append of 5. Similarly I wanted to add nirmal to the level employee_name in the list. employee_list again, it will be instead of 0, it will be 1, dot append nirmal.

If we print the updated lists; now it will have 5 ids 1, 2, 3 and 4 and 5 and we have 5 employee_names, Ram, Preethi, Sathish, Karan and Nirmal and also the number of employees is 5.

(Refer Slide Time: 04:55)

Modifying components using append()

- Adding a list to a list
- Adding a new list **age** to the existing **employee_list**
age=[23,25,36,43,52]
In [17]: employee_list.append([23,25,36,43,52])
- The new list gets added as a new level at the end
- Print the updated list
In [18]: print(employee_list)
[[1, 2, 3, 4, 5], ['Ram', 'Preethi', 'Sathish', 'Karan', 'nirmal'], 5,
[23, 25, 36, 43, 52]]

Python for Data Science 9

Let us say if I wanted to add a list to a list. Adding a new list age to the existing employee_list. So, now what I will do is I will create a age as a list and I will appended to the existing employee_list. So, we already have 5 employees. So, the respective ages will also be created as a list.

So, I am creating a age as a list which is equal to 23, 25, 36, 43 and 52. So, if I wanted to append to the existing list so, I will call the employee_lists.append on the sets of values. So, this is a one way or else you can do employee_list.append(age) as well.

So, this list its gets added as a new level at the end of the existing list. So, if you print the updated list so, we already had id, employee_name, number of employees. Now since we have added age to the existing employee_list so, it will be added at the last.

So, next we will use a inbuilt function called insert. So, in append, we saw how to add a list at the last. So, if you wanted to add an element at the specified position, then insert command works well.

(Refer Slide Time: 06:19)

Modifying components using insert()

- `insert()` - adds an object at the given position in a list
- Syntax: `list_name[index].insert(position,object)`
- Existing list

```
In [18]: print(employee_list)
[[1, 2, 3, 4, 5], ['Ram', 'Preethi', 'Sathish', 'Karan', 'nirmal'], 5,
 [23, 25, 36, 43, 52]]
```

- Adding number **'6'** at the **1st position** to the level **id** from **employee_list**

```
In [22]: employee_list[0].insert(0,6)
```

Python for Data Science 10

So, what does insert does is, it adds an object at the given position in the list. So, let us look at the syntax; `list_name[index].insert(position,object)`. So, this is our existing list. So, 1, 2, 3 ids and the 5 employee_names number of employees and the respective ages. Let us say if I wanted to add id number 6 to the id at the first position.

So, it will be `employee_list` name and we have to specify the index. So, the id as a level 0 dot insert position is at which position you wanted to insert. So, I am inserting at the first position. So, corresponding index will be 0 in python indexing starts from 0 to n-1. So, I wanted to add the value of 5. So, in the object I will specify 6.

(Refer Slide Time: 07:27)

Modifying components using insert()

```
In [22]: employee_list[0].insert(0,6)
```

- Print the updated list

```
In [23]: print(employee_list)
[[6, 1, 2, 3, 4, 5], ['Ram', 'Preethi', 'Sathish', 'Karan', 'nirmal'], 5, [23, 25, 36, 43, 52]]
```

Python for Data Science 11

So, if you print the updated list 6 id will be placed at the first position. So, the indexing will be this 0 remaining will be same.

Next we will see how to remove elements from the list. So, there are various methods we can use del, remove and pop. So, let us look at one by one.

(Refer Slide Time: 07:49)

Modifying components using del

- **del**- removes the object at the specified index number
- Syntax: **del list_name[index1][index2]**
- In the above syntax,
 - **index1**- index number of the top level of components to be dropped
 - **index2** corresponds to the sub level of components to be dropped

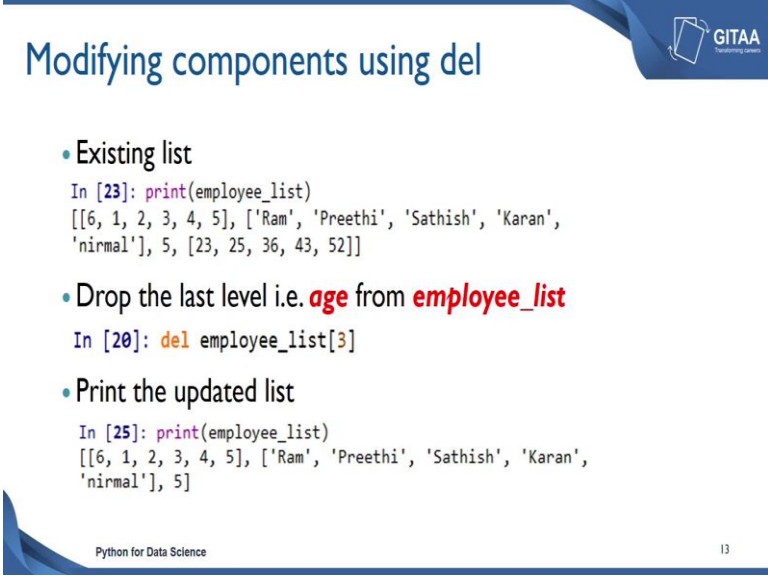
Python for Data Science 12

So, del it removes the object at the specified index number, let us look at the syntax. So, if you give del is a key word list name and inside the square brackets you have to

mention the index 1 and the index 2. Index 1, it corresponds to the index number of the top level components of the list which has to be dropped.

Similarly, index two is a index number of the sub level components to be dropped.

(Refer Slide Time: 08:21)



The slide is titled "Modifying components using del" and features the GITAA logo in the top right corner. It contains three bullet points and corresponding code snippets:

- Existing list

```
In [23]: print(employee_list)
[[6, 1, 2, 3, 4, 5], ['Ram', 'Preethi', 'Sathish', 'Karan', 'nirmal'], 5, [23, 25, 36, 43, 52]]
```
- Drop the last level i.e. **age** from **employee_list**

```
In [20]: del employee_list[3]
```
- Print the updated list

```
In [25]: print(employee_list)
[[6, 1, 2, 3, 4, 5], ['Ram', 'Preethi', 'Sathish', 'Karan', 'nirmal'], 5]
```

At the bottom of the slide, it says "Python for Data Science" on the left and "13" on the right.


So, this is our existing list. So, if I wanted to drop the last level that is age; id, it is a level 0 and Ram, Preethi, Sathish is the employee_names. So, if the level is 1 and number of employees which is 5 is level 2 and age is at the level 3. So, if I wanted to remove the age from the employee_list so, it will be del employee_list[3] which is basically the index number. So, if you print the updated list, now our age list will be removed from the employee_list next look at how to use the remove option.

(Refer Slide Time: 09:03)

Modifying components using remove()

- **remove()** - removes the first matching object from a list
- Syntax: **list_name[index].remove(object)**
- Existing list

```
In [25]: print(employee_list)
[[6, 1, 2, 3, 4, 5], ['Ram', 'Preethi', 'Sathish', 'Karan', 'nirmal'], 5]
```



Python for Data Science

GITAA

So, remove what it does is it removes a first matching object from a list. So, we will look at the syntax list name inside the square brackets, we have to mention the index number dot remove and inside the parenthesis, we have to specify the object. So, this is our existing lists. So, after we deleted the age.

(Refer Slide Time: 09:27)

Modifying components using remove()

- Remove **'Ram'** from the level **employee_name** from **employee_list**

```
In [22]: employee_list[1].remove("Ram")
```

- Print updated list

```
In [27]: print(employee_list)
[[6, 1, 2, 3, 4, 5], ['Preethi', 'Sathish', 'Karan', 'nirmal'], 5]
```

- Here **'Ram'** occurs only once

Python for Data Science

GITAA

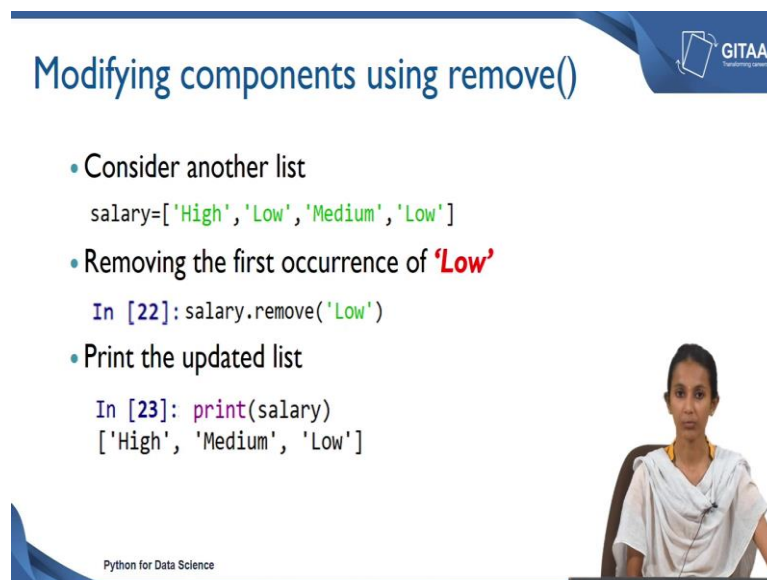
15

So, let us say if I wanted to remove Ram from the employee_name in the employee_list. So, employee_list so, the index number is one dot remove inside the double quotes Ram.

So, what it does is it searches for the first occurrence of the Ram and then it will be removed from the list.

So, we have only one Ram. So, we have one Preethi, one Sathish, one Karan and one Nirmal. So, if you give dot remove Ram so, it will remove the Ram from the employee_list.

(Refer Slide Time: 10:02)



The slide features a blue header with the title "Modifying components using remove()" and the GITAA logo. The main content includes a list of bullet points and code snippets. A woman in a white sari is visible in the bottom right corner of the slide area.

- Consider another list

```
salary=['High', 'Low', 'Medium', 'Low']
```
- Removing the first occurrence of **'Low'**

```
In [22]: salary.remove('Low')
```
- Print the updated list

```
In [23]: print(salary)  
['High', 'Medium', 'Low']
```

Python for Data Science

So, when you print the updated list, Ram will be removed from the employee_list. So now you will have 6 ids 6, 1, 2, 3, 4 and 5 and you will have 4 employees which is Preethi, Sathish, Karan, Nirmal and you will have the number of employees is equal to 5 since has ram has occurred once it has been removed.

So, let us consider another list create a list which has salary High, Low, Medium and Low. So, if I wanted to remove the first occurrence of low which is so, High stands for the index number 0, Low stands for the index number 1, Medium stands for a index number 2; again Low stands for the index number 3.

So, if I wanted to remove the first occurrence of low which since in the index position of 1, then if I give salary dot remove low so, it will be removed from the salary list. So, if you print the updated list. So, it returns the value of high, medium and low.

So, the corresponding index position now it will be changed to 0, 1 and 2. Next we will the modify the components using the pop function.

(Refer Slide Time: 11:15)

Modifying components using pop()

- **pop()** - displays the object that is being removed from the list at the specified index number
- Syntax: **list_name[index1].pop(index2)**
- In the above syntax,
 - **index1** - index number of the top level of components to be dropped
 - **index2** corresponds to the sub level of components to be dropped

Python for Data Science GITAA
Geospatial Intelligence Training Academy 17

So, pop what it does is it displays the objects that is being removed from the list at the specified index number. So, let us look at the syntax list name index1 dot pop and index2. So, the index1 corresponds to the index numbers of the top level of components to be dropped, index2 it corresponds to the sub level of components to be dropped.

(Refer Slide Time: 11:43)

Modifying components using pop()

- Existing list

```
In [27]: print(employee_list)
[[6, 1, 2, 3, 4, 5], ['Preethi', 'Sathish', 'Karan', 'nirmal'], 5]
```
- Removing number '4' from the 5th position of level **id** from **employee_list**

```
In [29]: employee_list[0].pop(4)
Out[29]: 4
```
- Print the updated list

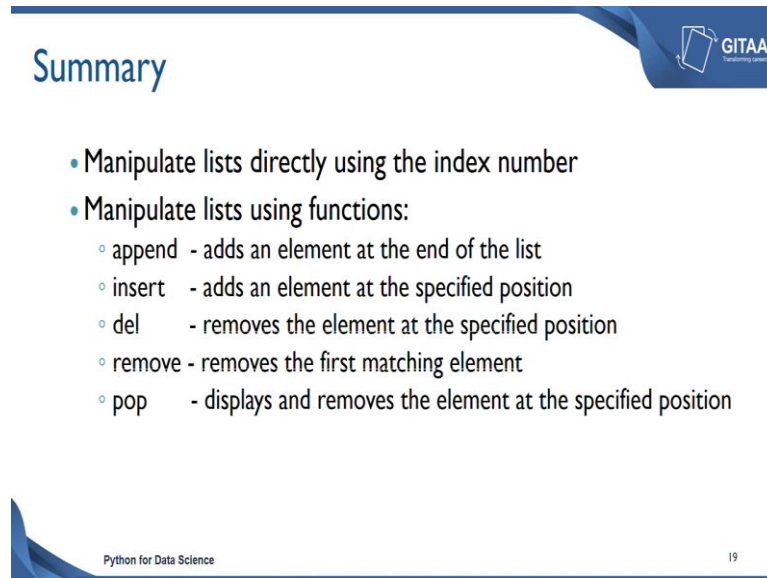
```
In [30]: print(employee_list)
[[6, 1, 2, 3, 5], ['Preethi', 'Sathish', 'Karan', 'nirmal'], 5]
```

Python for Data Science GITAA
Geospatial Intelligence Training Academy 18

So, this is our existing list. So, I have ids and 4 employees and the number of employees is 5. Let us say if I wanted to remove the fourth id from the fifth position of the employee_list so, in the id.

So, if I give employee_list 0 dot pop 4. So, it pops up the element in the list is going to be removed so, which is four is going to be removed from the id level. If we print the updated list 6, 1, 2, 3; 4 will be removed from the level id and you will have 4 employee_names Preethi, Sathish, Karan and Nirmal and you will have number of employees is equal to 5.

(Refer Slide Time: 12:33)



Summary

- Manipulate lists directly using the index number
- Manipulate lists using functions:
 - append - adds an element at the end of the list
 - insert - adds an element at the specified position
 - del - removes the element at the specified position
 - remove - removes the first matching element
 - pop - displays and removes the element at the specified position

Python for Data Science 19

So, let us summarize. So, first we saw how to manipulate the list using the index number and we also saw some of them built in functions. So, first we saw append. So, append it adds an element at the end of the list and also it creates a new list; it is also called as a concatenation of a list. So, next we saw insert. So, if I wanted to add an element at the specified position insert works very well.

Next we saw how to remove the elements. So, there are three ways one is del; first one is del so, it removes a element at the specified position. So, if we give remove, it removes the first matching element from the list. If I give pop it displays and removes element at the specified position of the lists.

Thank you.