Python

Data Types

Chapter 13

Suresh techs

Suresh techs – Technology and Entertainment

Subscribe

Subscribe: You don’t miss our future videos

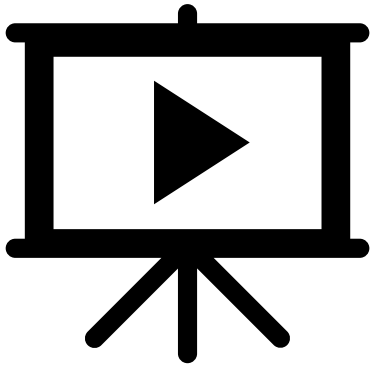
Doubts

Follow In Instagram
#sureshtechs

- Link in the description
- Clear your doubts chatting with me

Data types

- Data + Type
- మన మనుషుల్లో రకాలు



Built In **data types** In Python

- Numeric (int, float, complex)
- Sequence Types (strings, list, tuple)
- Boolean(True, False)
- Set
- Dictionary

Few terms before we start

- Mutable
- Immutable
- Hashable

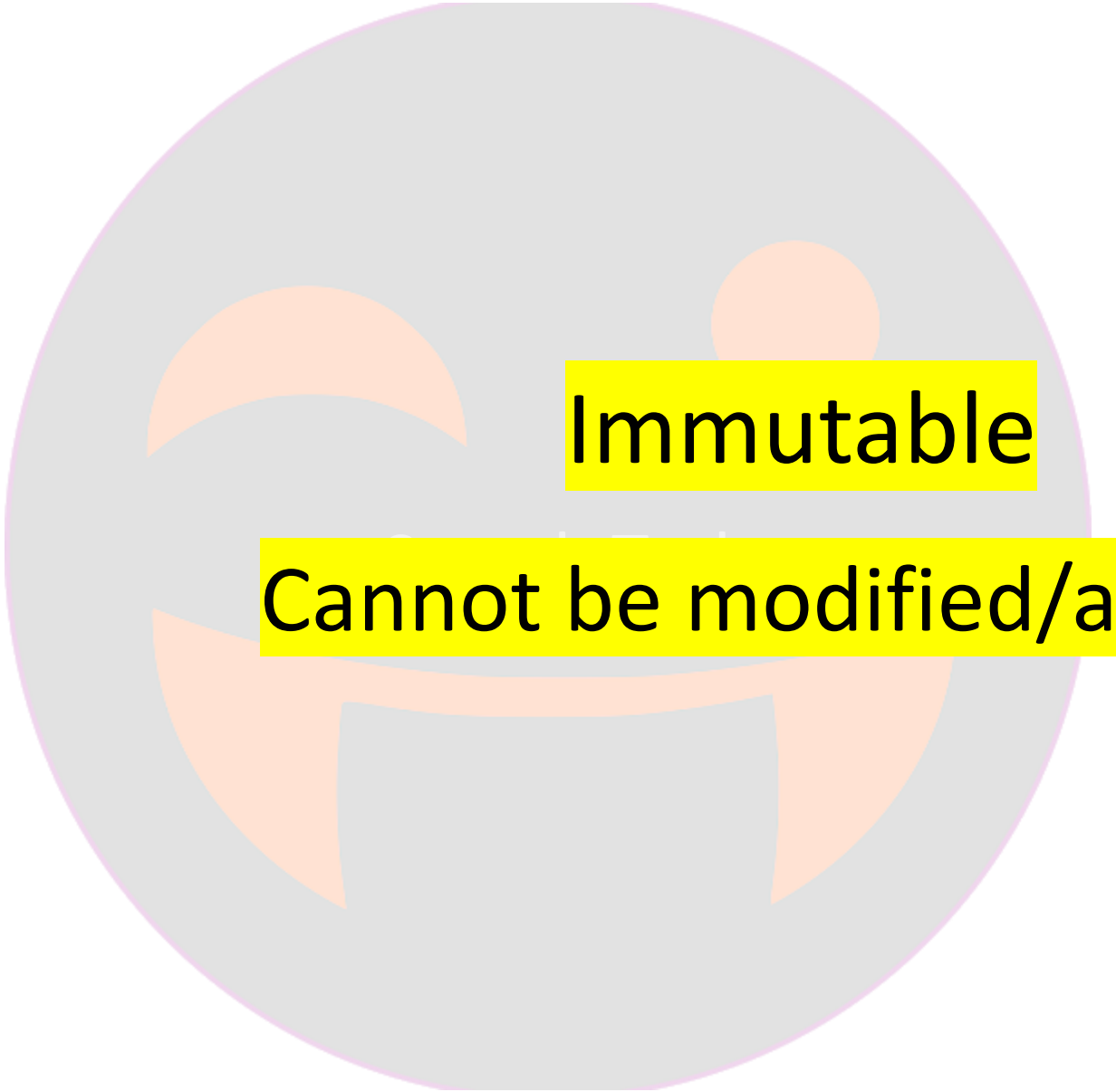
echs

Mutable

Can be altered

Titles

- Power star
- Mega star
- Rebel star
- Burning star

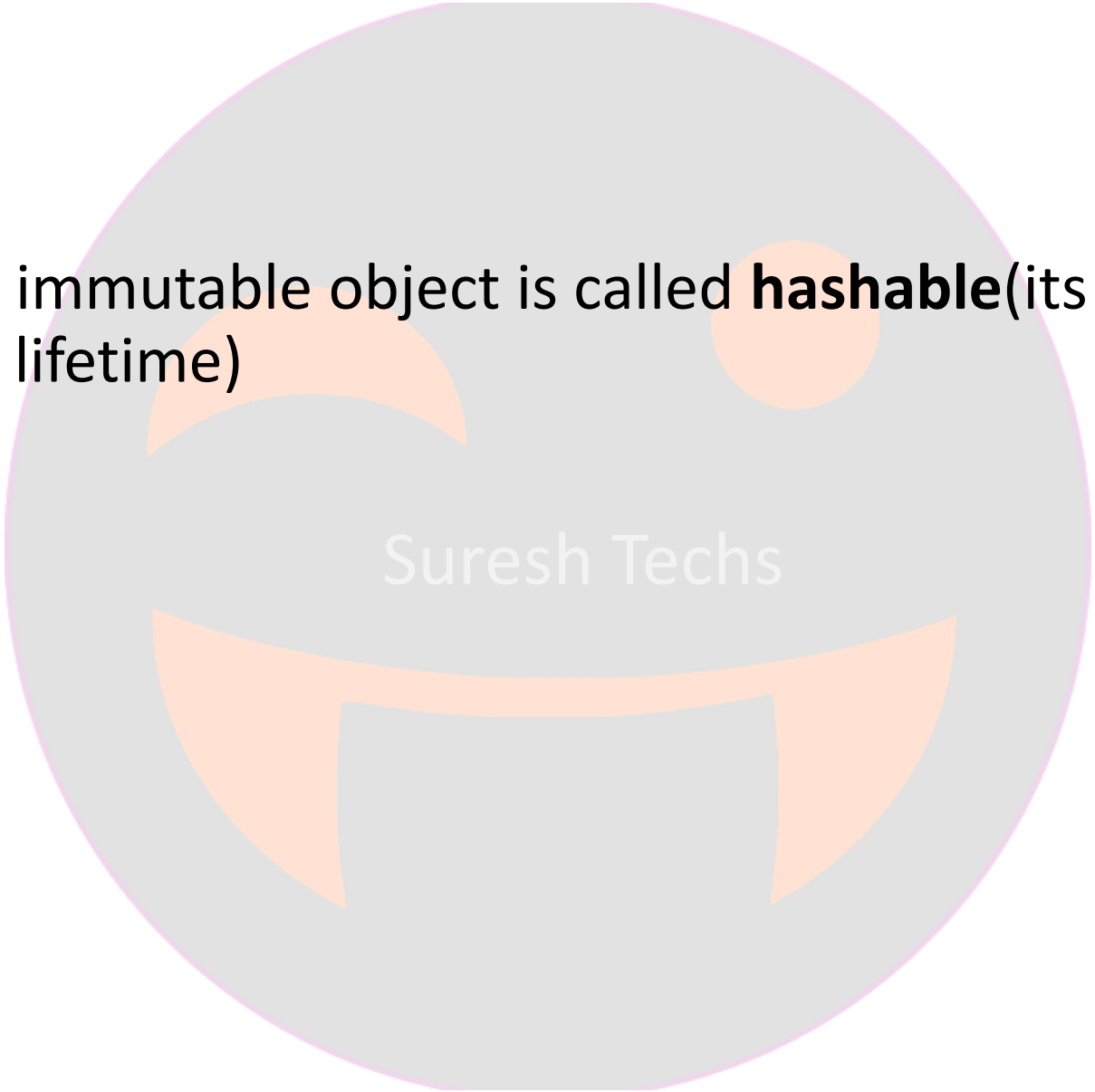


Immutable

Cannot be modified/altered

Hashable

- Hashable: Any immutable object is called **hashable**(its value does not change during lifetime)



Another **Important note** before we start

- If you don’t have knowledge in java, it may confuse a bit but don’t worry.
- I will try my best to explain

Programming - What is class, object?

- Please watch “Java in 10 minutes”
- Sorry for bringing Java in python course but it is required.

```
Home home1 = new Home();
home1.clean();
home1.getNumberOfBathRooms();
```

Home is class and home1 is instance (object) of the class Home

```
public class Home{
    //properties
    int length = 50;
    int breadth = 7;
    int numberOfRooms = 4;
    int capacity = 6;

    //methods or functions
    void clean(){

    }
    void paint(){

    }
    void lock(){

    }

    void setNumberBathRooms(){
    }

    int getNumberOfBathRooms(){
        return numberOfRooms;
    }
}
```

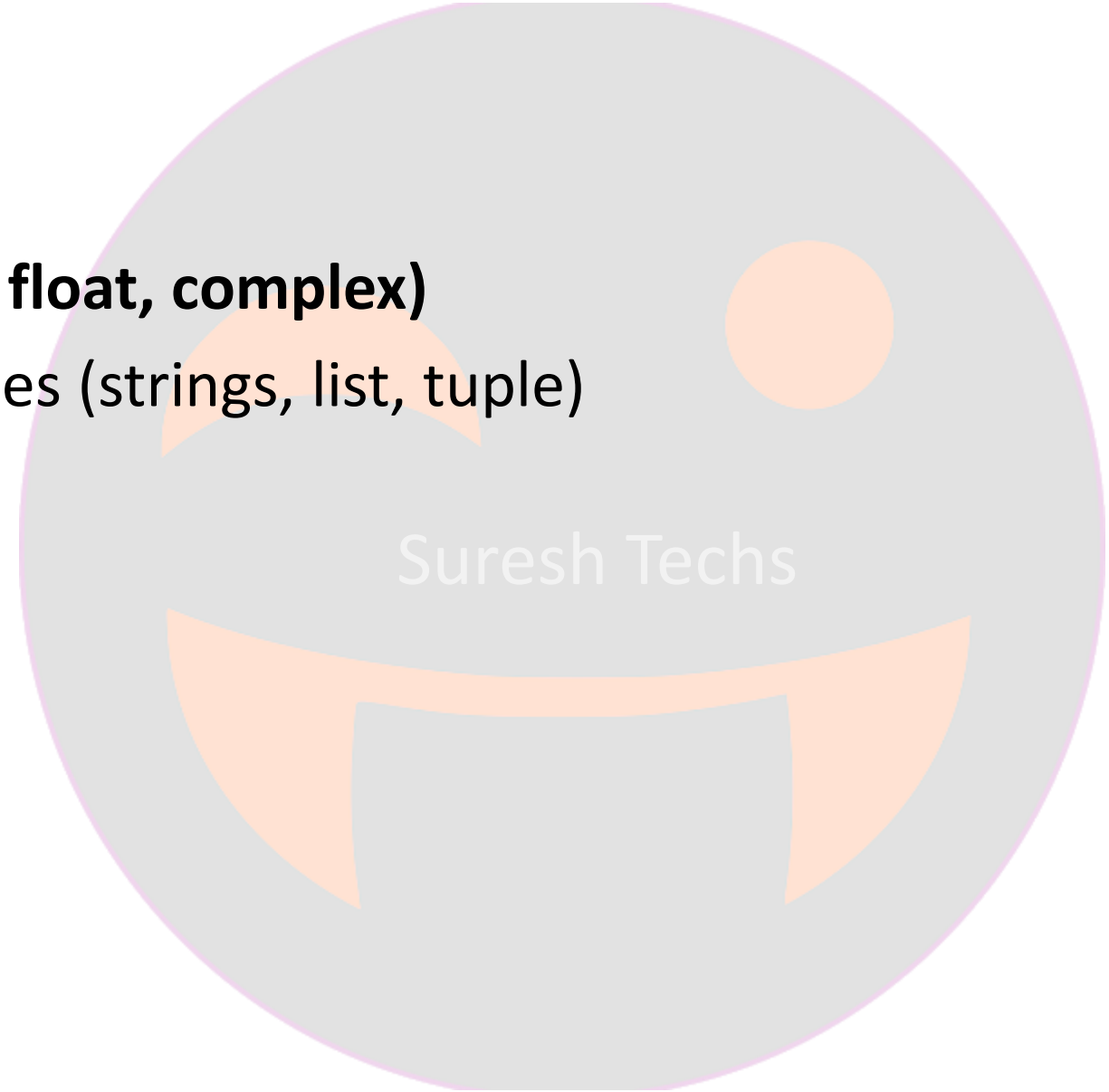
Big note:

- In python **Data types are classes** and variables are instances(objects) of these classes
- **Variables are objects – please keep this in mind**

Ex: sum = 10

Let’s start

- **Numeric (int, float, complex)**
- Sequence Types (strings, list, tuple)
- Boolean
- Set
- Dictionary



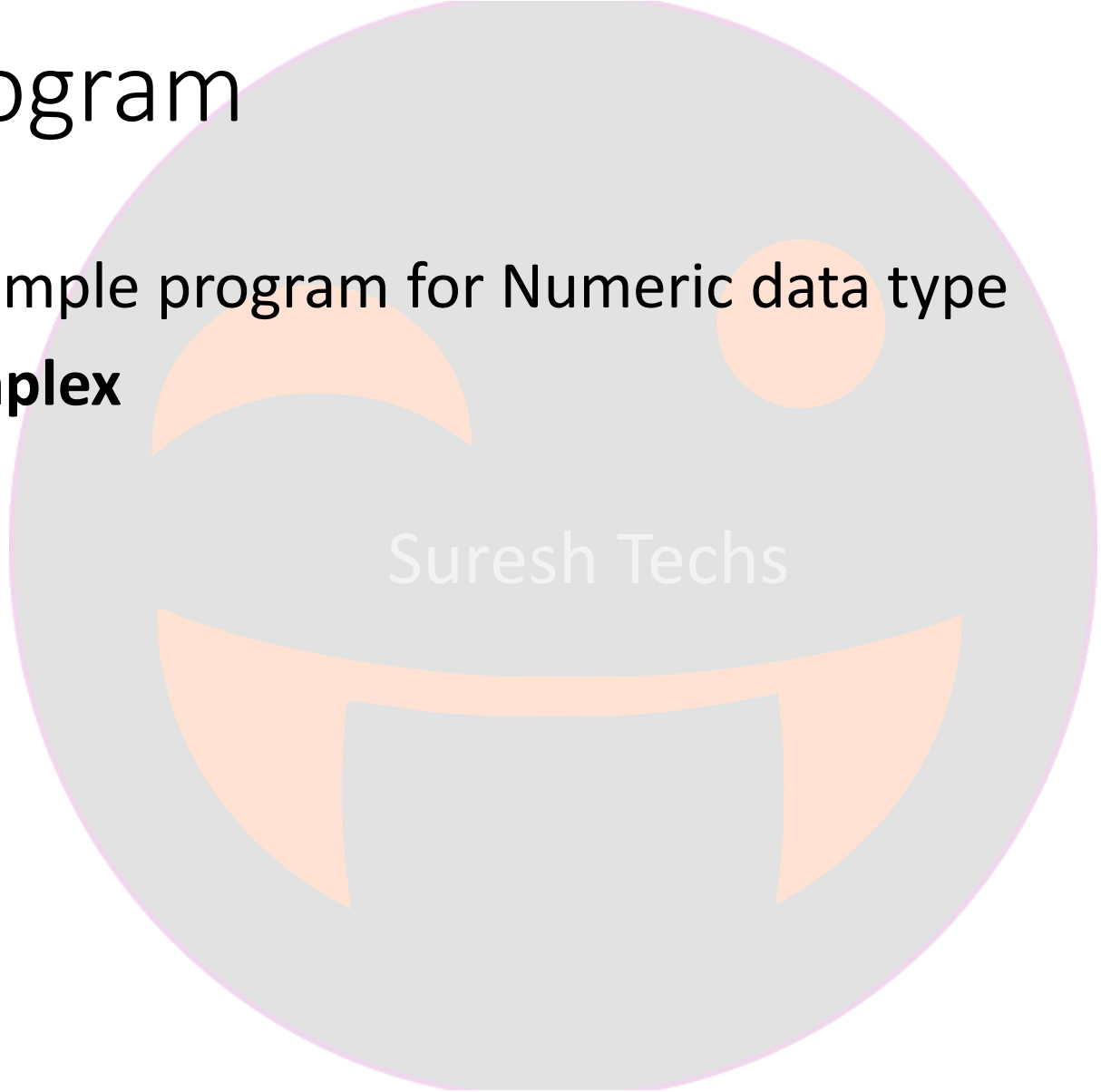
int, float, complex

Numeric – Integers, Float, Complex

- Integers
  - Contains positive or negative whole numbers(10, 20, -100, -90). There is no limit to how long an integer can be
  - Represented by **int** class
- Float
  - Real number with floating point representation(1.02,-29.87,22.09)
  - Represented by **float** class
- Complex
  - Specified as (real part)+(imaginary part)
  - Ex: 10+3j
  - Represented by **complex** class

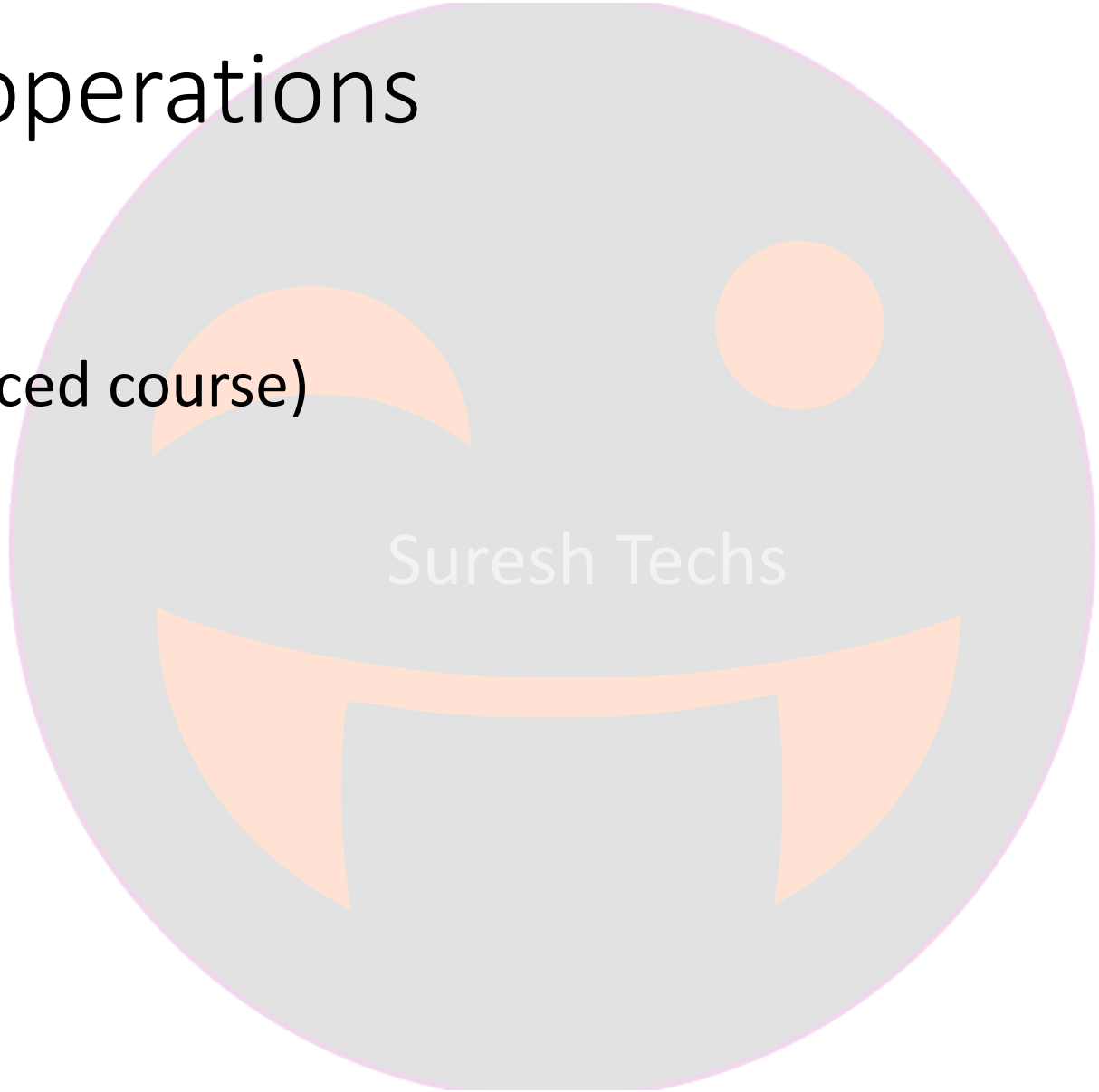
Simple program

- Let’s write a simple program for Numeric data type
- **int, float, complex**



Complex operations

- complex
- cmath (Advanced course)



Built In data types In Python

- Numeric (int, float, complex)
- **Sequence Types (strings, list, tuple)**
- Boolean
- Set
- Dictionary

Sequence Type

- Ordered collection of similar or different data types
- Used to store multiple values in some order
  - **String**
  - List
  - Tuple

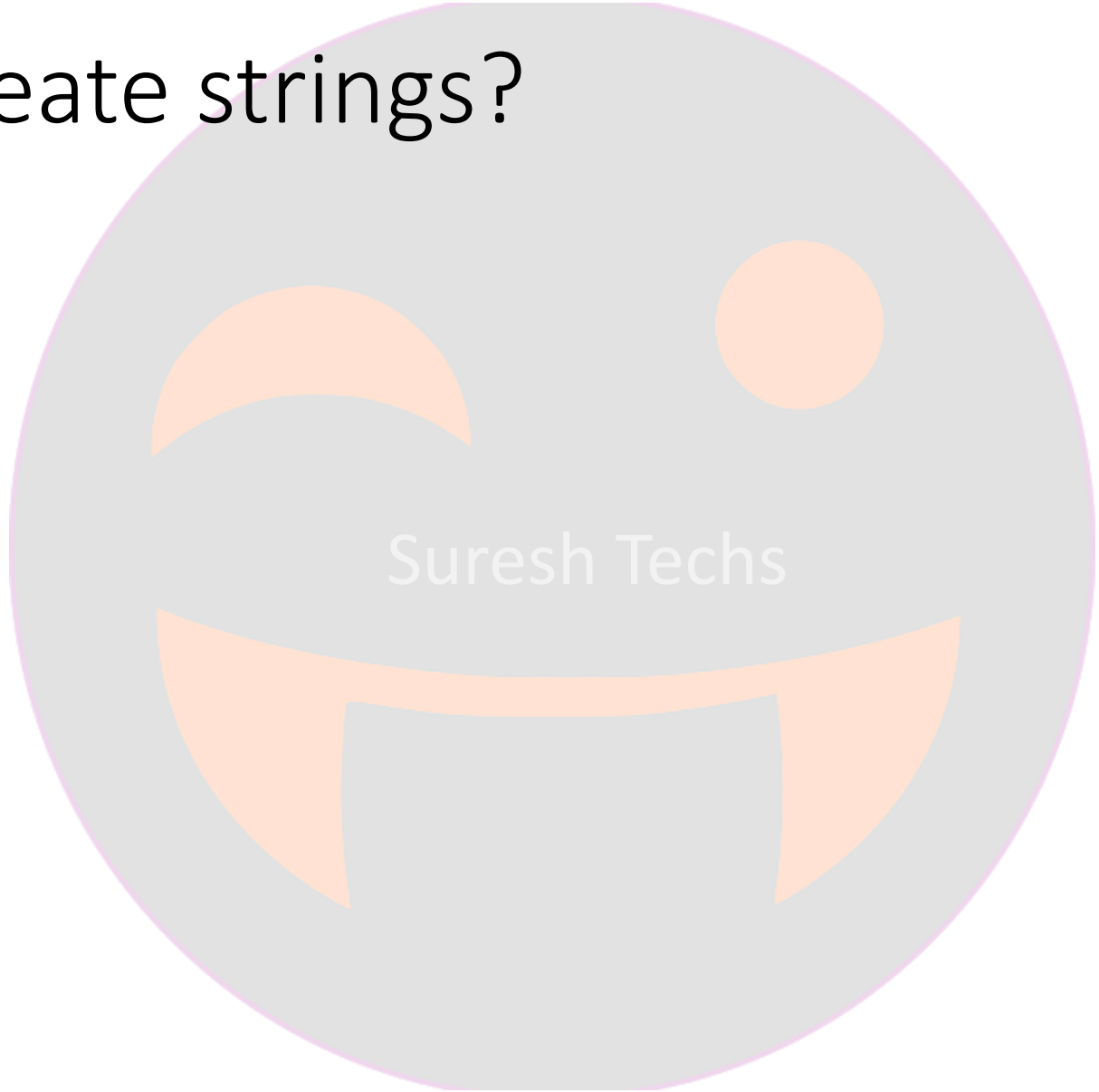
String

String

- Sequence of characters
- “Suresh Techs”
- There is not character data type in python
- Single character is also considered as a string in python

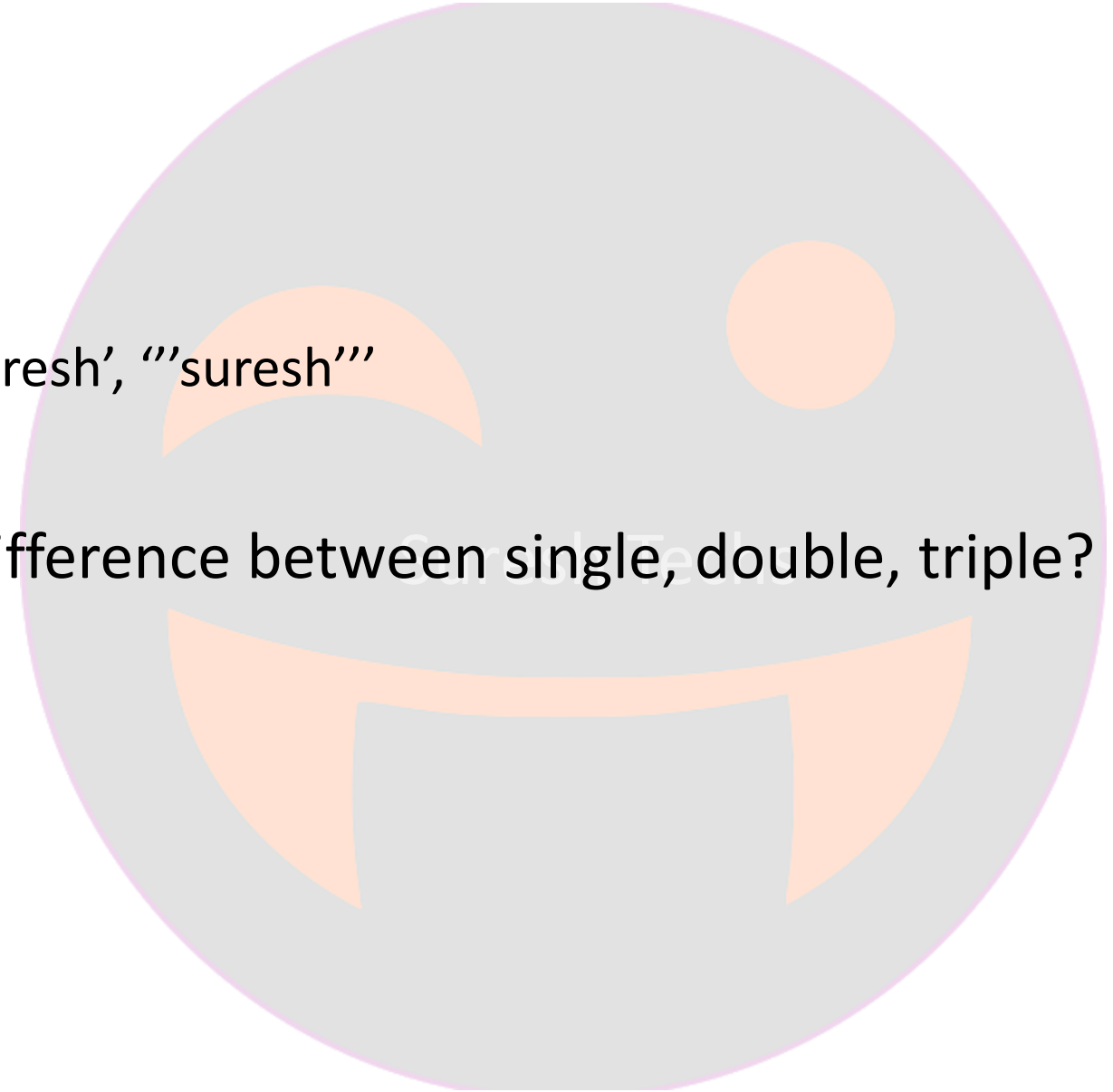
How to create strings?

- Single Quote
- Double Quote
- Triple Quote



Strings

- Ex: suresh
  - “suresh”, ‘suresh’, “’’suresh’’’
- What is the difference between single, double, triple?



Differences

- No difference between single and double quotes, **we can use as per the requirement**
- Use triple quotes for multi line strings

Program

- Print below string:

suresh techs is a youtube channel started in 2020, let’s please subscribe and motivate suresh for more videos.

Suma akka “anchoring” is awesome

Length of string

- len(string)
- Note: len function calculates the length of any object that has a size(lists, dictionaries etc)

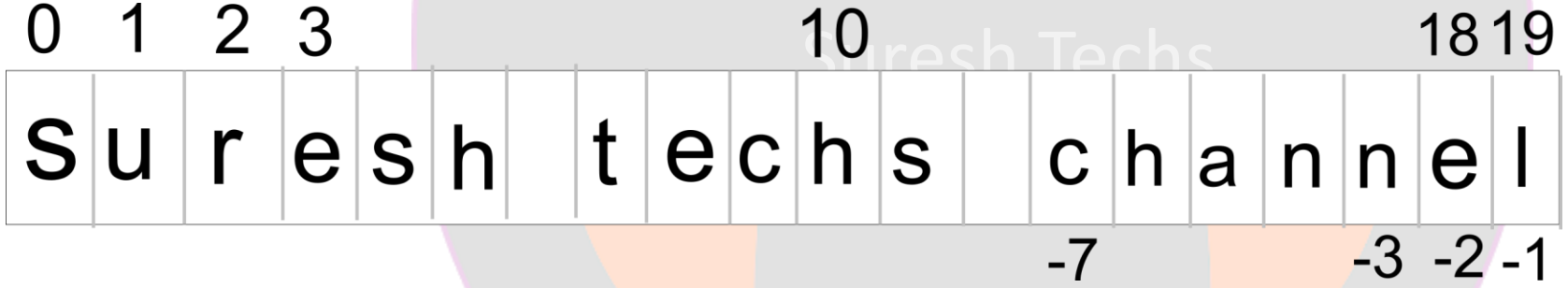
Accessing characters in a string?

- Can access individual characters using **indexing** and range of characters using **slicing**


1. Biyyam
2. Kandipappu
3. Tomato
4. Chintapandu
5. Salt

Indexing & Slicing

- “Suresh techs channel”

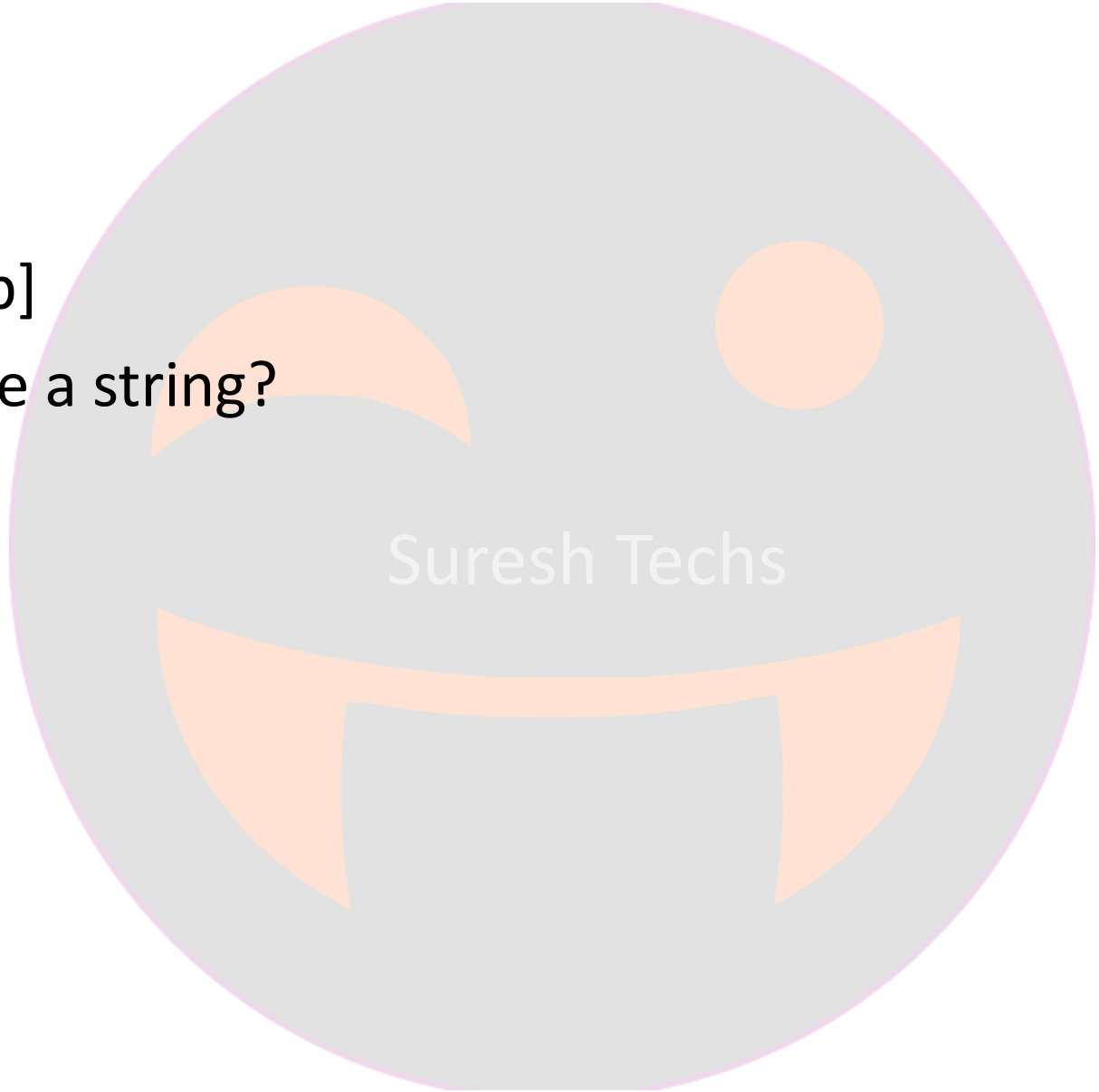


Program(Indexing and slicing)

```
#slicing
info = "suresh techs channel"
print('suresh techs channel',len(info))
print(info[0])
print(info[2:15])
print(info[-1])
print(info[12:])
print(info[-8:])
print(info[-9:])
print(info[:-8])
```

Step

- [start:end:step]
- How to reverse a string?



Deleting/updating a string

- You can’t update/delete a characters from a string as they are **immutable**(elements cannot be changed once assigned)
- Only new strings can be assigned to the same name
  - name=“suresh”
  - Name[2]=“w” – not possible
  - name=“hareesh” - possible

Deleting entire string

- We cannot delete or remove characters from strings
- We can delete them all together
- name = “suresh”
- del name[0]
- del name

String operations

- There are many but will discuss few here
- Remaining in Advanced course

Concatenation

- Joining of two or more strings into a single string is called concatenation
- + operator is used to concatenate
- * operator is used to repeat the string

Iterating through string

- We can use loops(for, while) to iterate over strings
- Will work on this when we talk about control flows

String membership test

- Test if substring exists in a string or not
- in
- not in

- This is enough for now, will discuss more about string in advanced course.

Sequence Type

- Ordered collection of similar or different data types
- Used to store multiple values in some order
  - String
  - **List**
  - Tuple

List

Lists

- Holds heterogenous data(different types of data)
- Frequently used data type
- **Ordered collection of items**
- **Mutable**
- **Allows duplicate items**

- Syntax: [data1,data2,data3, ….]

Creating lists

- Place items inside square brackets [ ], separated by commas.
- It can be of any number of items with any type

Creating lists

- Create a list to hold your 10th class marks
- Create a list with 5 of your friends whom you met first in school/college
- Empty list?: list without any items is called empty list

Can we keep one list inside another list?

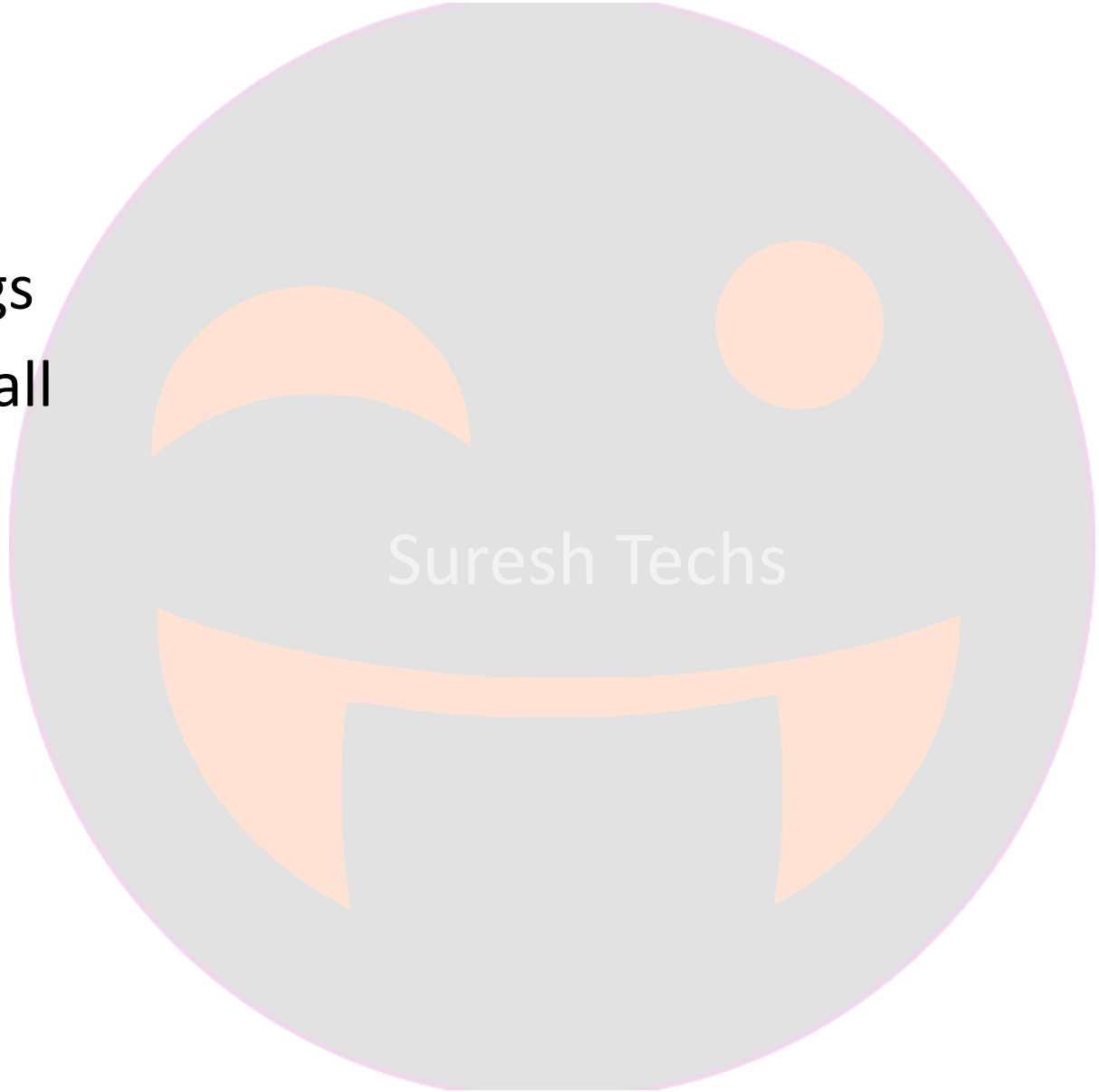
- **YES**
- Also known as **nested lists or multidimensional lists…**

Accessing elements from list

- Indexing
- Index Error, Type Error
- **Print your second friend**
- Can we use negative indexing?
- **Print your second friend from last**
- **Print your last friend**
- **How to access elements from nested lists?**

Slicing

- Same as strings
- No change at all

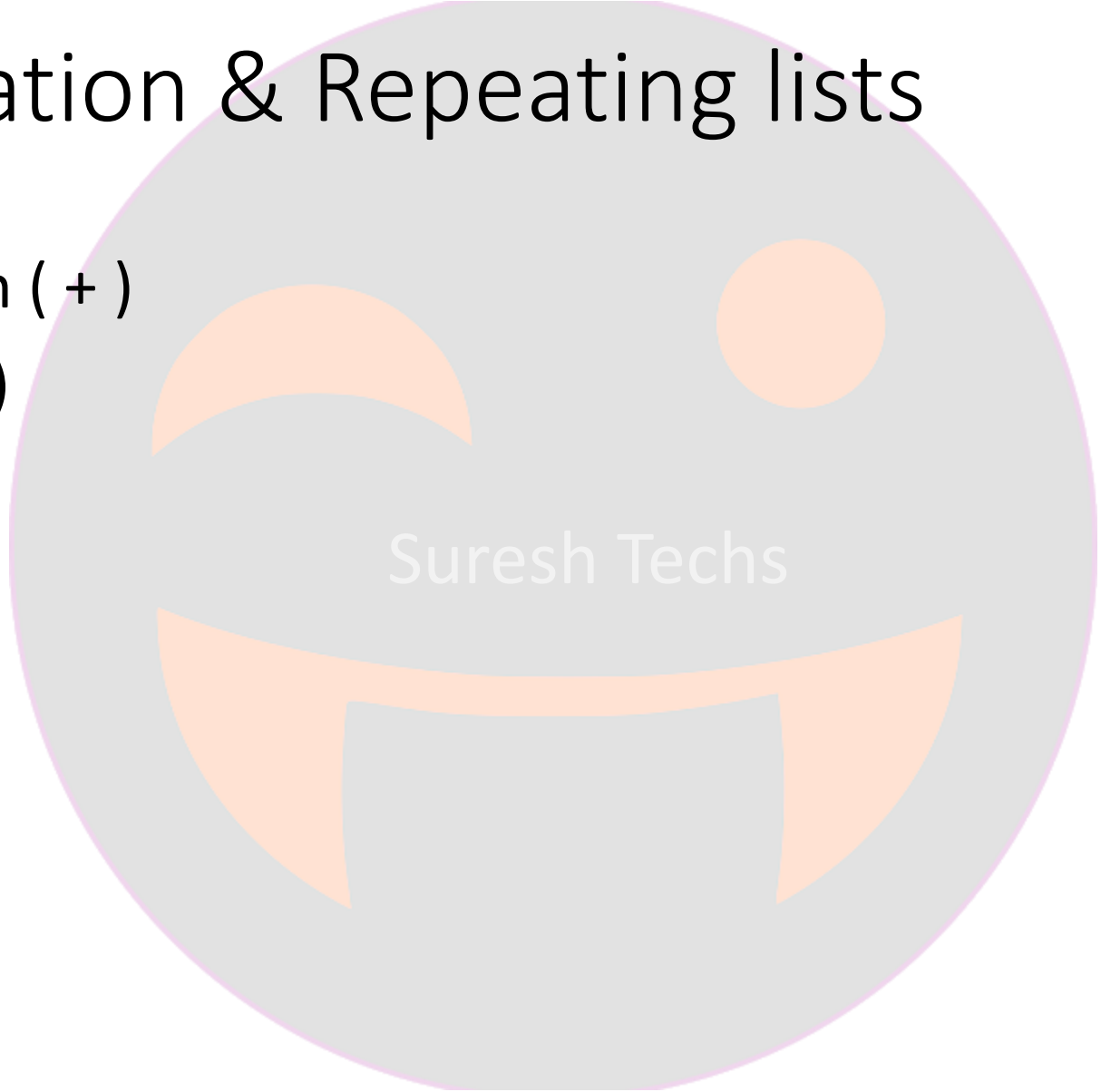


Change list items?

- Mutable
- So, we can change them
- = (What is this operator?)
- We can use assignment operator to change an item from list
- **Update math's marks after supplementary**

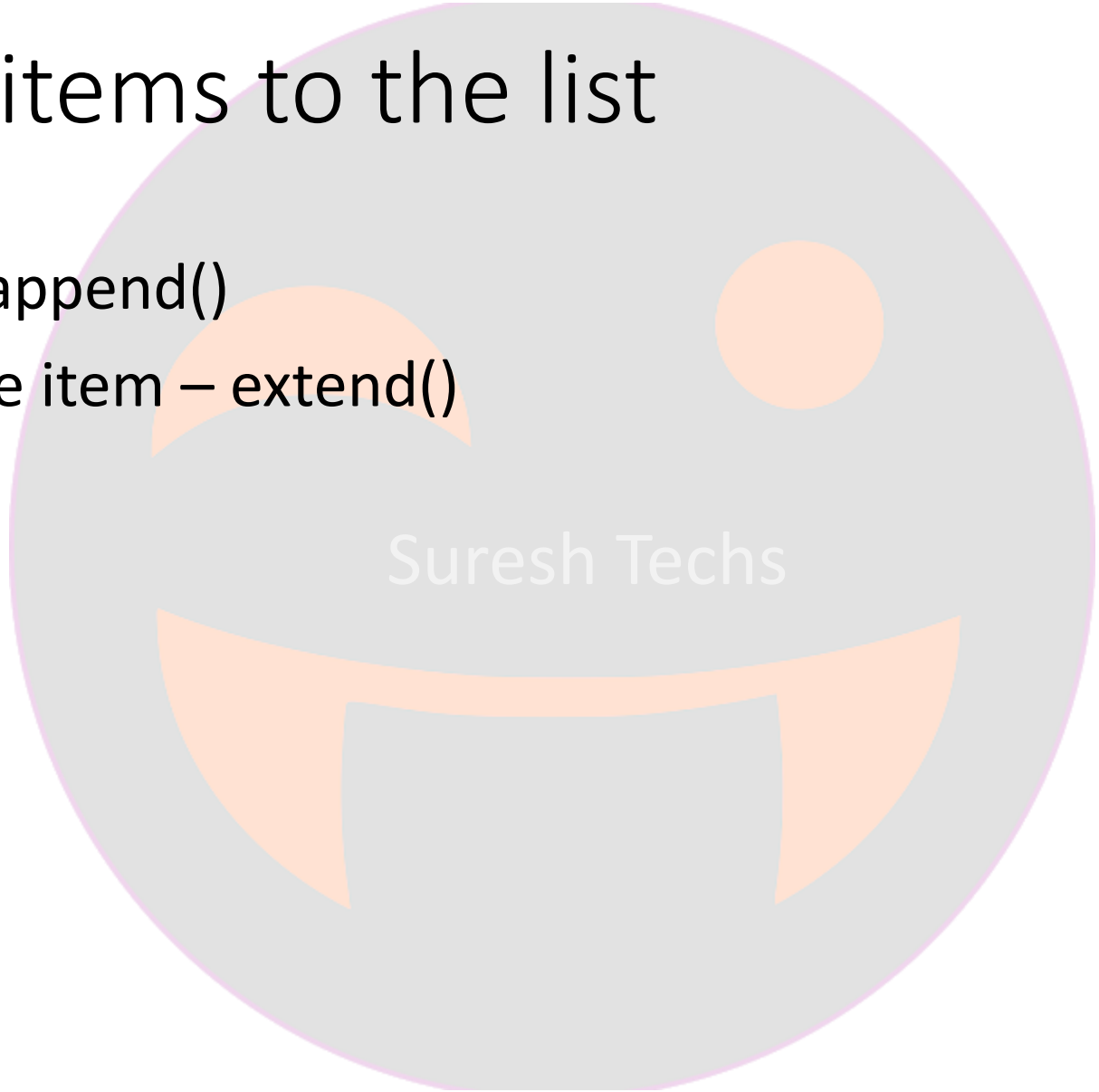
Concatenation & Repeating lists

- Concatenation (+)
- Repeating (*)



Add item/items to the list

- Single item – append()
- More than one item – extend()

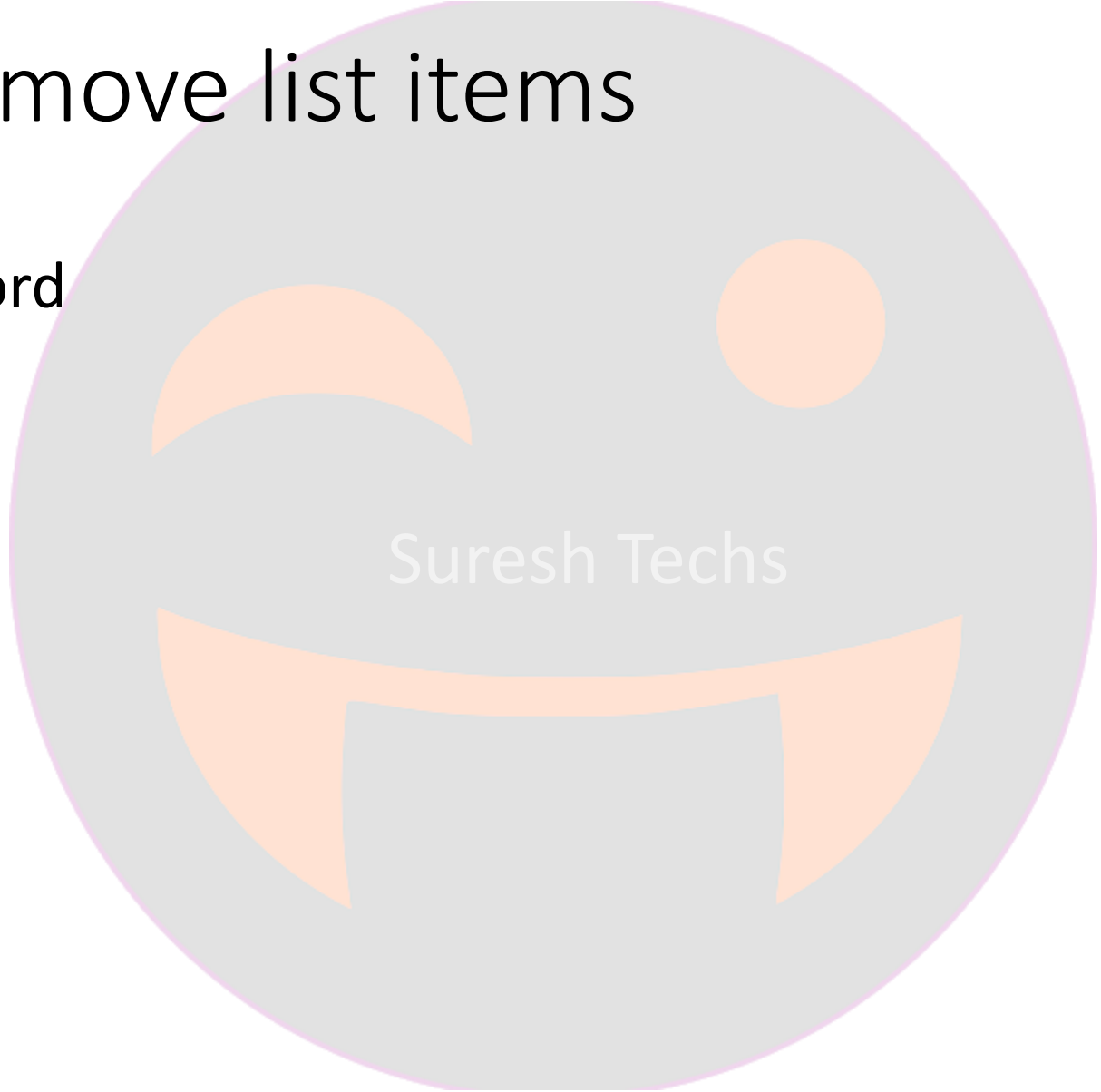


How to add at specific position?

- insert(position, value)
- Insert multiple elements at specific position using slicing

Delete/Remove list items

- Use del keyword
- del list[0]
- del list
- del list[1:3]



Delete/Remove list items

- remove(item): used to remove an item from the list
- pop(index): remove and returns item at specified index
- pop() – remove and returns last item from the list

How to clear a list?

- clear()
- That’s all on List

Filter, map, reduce

- Very important concepts, will be discussed later when we discuss anonymous functions(**lamba**)

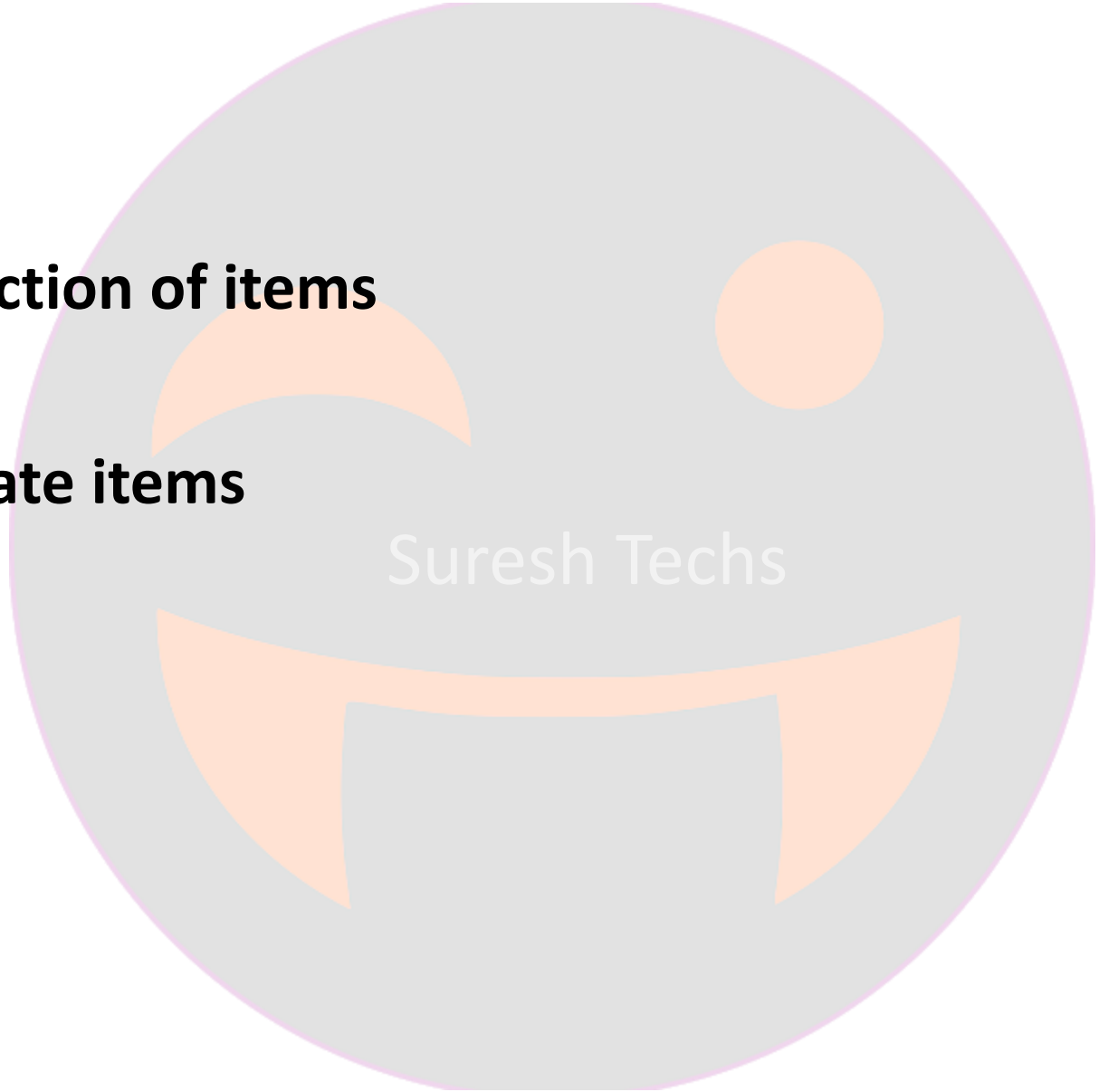
Sequence Type

- Ordered collection of similar or different data types
- Used to store multiple values in some order
  - String
  - List
  - **Tuple**

TUPLE

Tuple

- **Ordered collection of items**
- **Immutable**
- **Allows duplicate items**



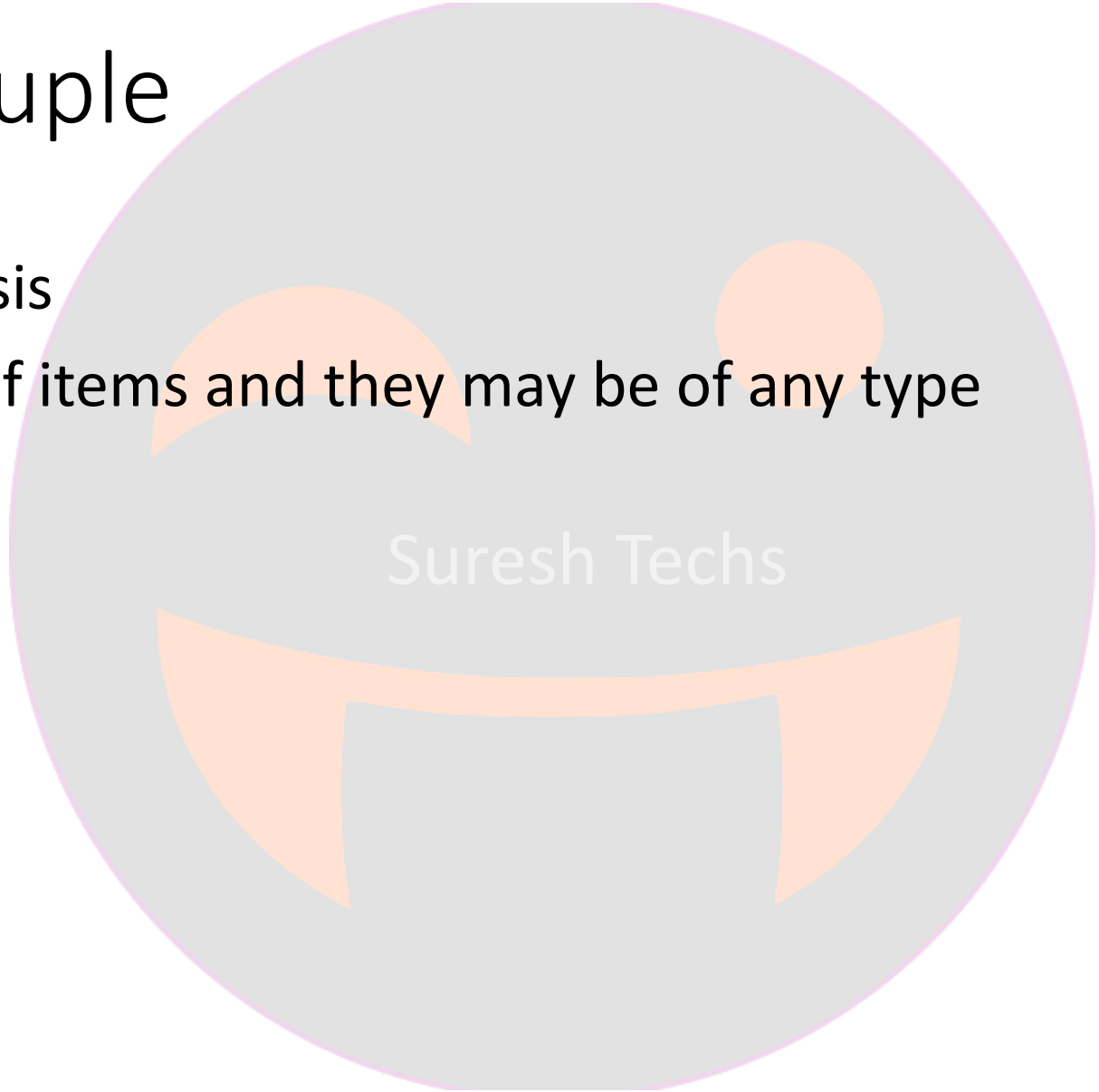
Tuple

- Similar as lists
- Use parenthesis ( ) to group elements
- Note: it is not mandatory to enclose in parenthesis, but it is a good practice to use

- **Main Difference:** We cannot change the elements of the tuple once it is assigned

Creating tuple

- ( ) – parenthesis
- Any number of items and they may be of any type



Tuple

- Create a tuple to hold your 10^{th} class marks
- Create a list with 5 of your friends whom you met first in school/college

Can we keep one tuple inside another tuple?

- **YES**
- Also known as nested tuples or multidimensional tuples…
- Print first element from both the tuples…

Tuples can also be created without parenthesis

- Then what if we have only one item in the tuple?
- We need to put comma to make it as a tuple

Accessing tuple elements

- Indexing
- Index Error, Type Error
- Nested tuples are accessed using nested indexing
- Negative indexing is also same
- Slicing to get range of values is also same

Changing tuple

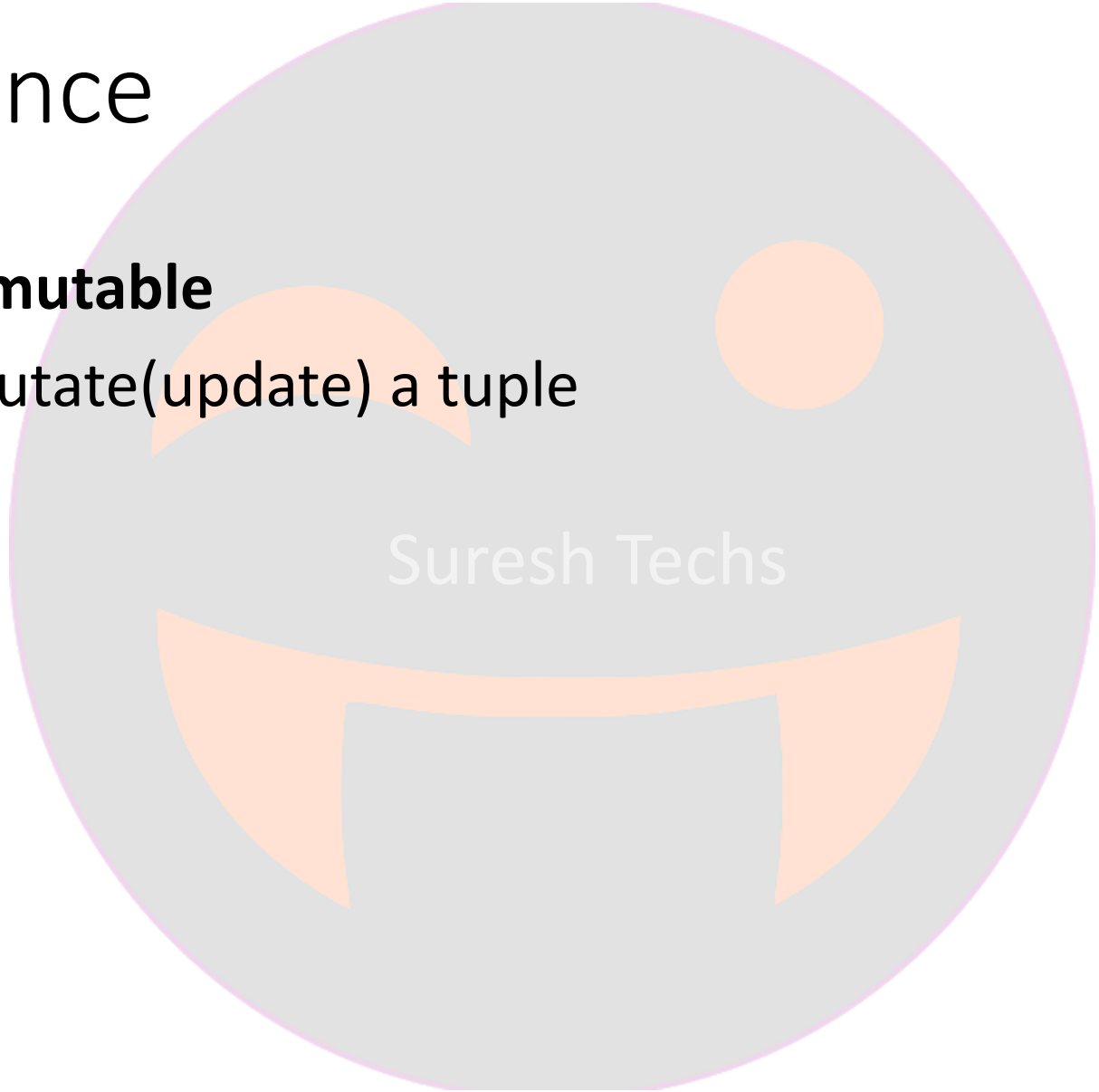
- Tuples are immutable
- Elements of a tuple cannot be changed once they are assigned but if the element itself is a mutable data type like list, then it can be changed

Deleting a tuple

- Since tuples are immutable, we can’t delete elements from it
- Instead, we can delete entire tuple using **del** keyword

Big difference

- Tuples are **immutable**
- We can not mutate(update) a tuple



Converting a list to tuple

- List can be converted to tuple
- tuple(list)

Concatenation

- Process of joining two or more tuples

- Note: we can only concatenate similar data types
- Important interview question

Built-in methods

- index(value) – Returns index of the given value
- count(value) – Returns the frequency of occurrence of a specified value

- Note: Few more are there. Will discuss about them as in when time comes.

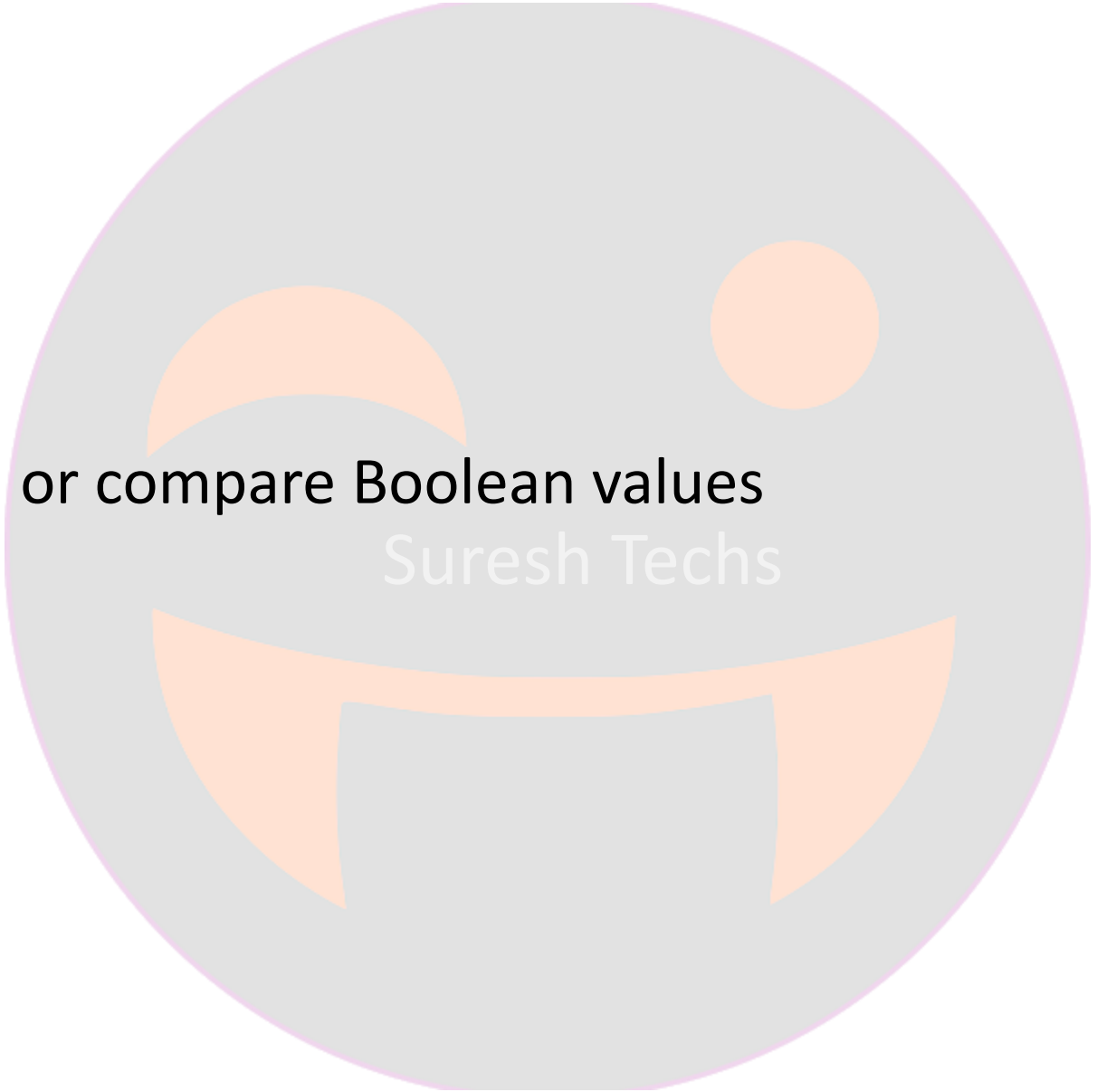
Built In data types In Python

- Numeric (int, float, complex)
- Sequence Types (strings, list, tuple)
- **Boolean**
- Set
- Dictionary

BOOLEAN
TRUE, FALSE

Boolean

- True
- False
- Used to assign or compare Boolean values



Numeric values as Booleans – bool()

- Any integer, floating point number or complex number **having zero** as a value is considered **False**
- Any integer, floating point number or complex number **having positive or negative number** as a value is considered **True**

Built In data types In Python

- Numeric (int, float, complex)
- Sequence Types (strings, list, tuple)
- Boolean
- **Set**
- Dictionary

SET

Set

- **Unordered collection of items**
- **Mutable**
- **Does not allow duplicate items(every element is unique)**

- Best at performance to find whether a specified element is available in the list?

Creating a set

- { } – curly braces
- built in set() function

- **NOTE:** Can have any number of items and they may be of different types(integer, float, tuple, string) but cannot have mutable elements like lists, sets, dictionaries as it’s elements

How to create an empty set?

- Empty list – []
- Empty set {} ?
- Use set() to create an empty set

Accessing element from set

- Indexing?
- Since they are unordered, we cannot use indexing or slicing
- **We can iterate over set using loops**

Add/update

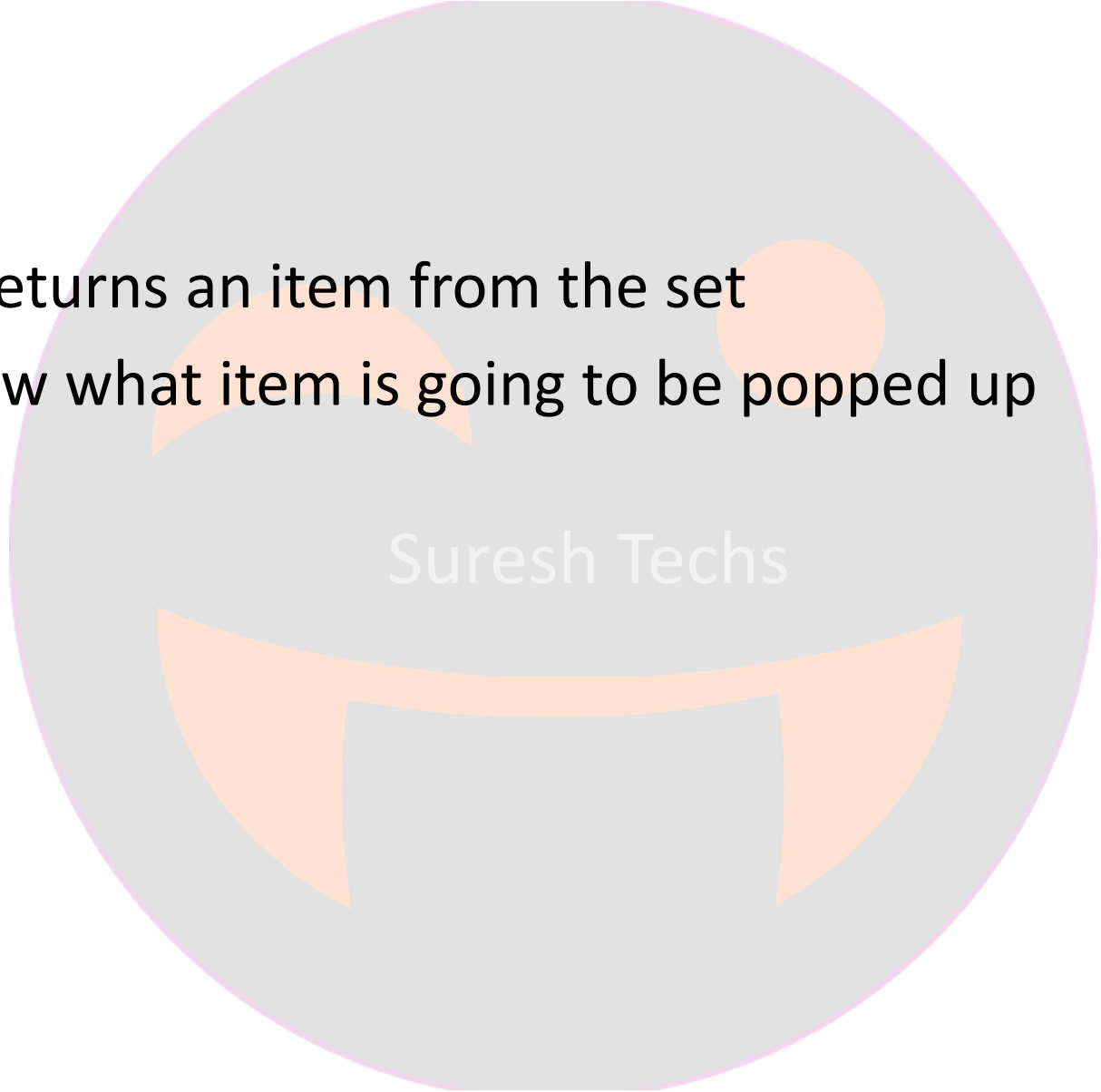
- add() method is used to add single element to the set
- update() method is used to add multiple elements. It can take tuples, lists, strings or other sets as arguments

Removing

- discard(item) – specified item will be removed from the set
- remove(item) – specified item will be removed from the set
- Then what is the main difference?

pop()

- Remove and returns an item from the set
- You never know what item is going to be popped up



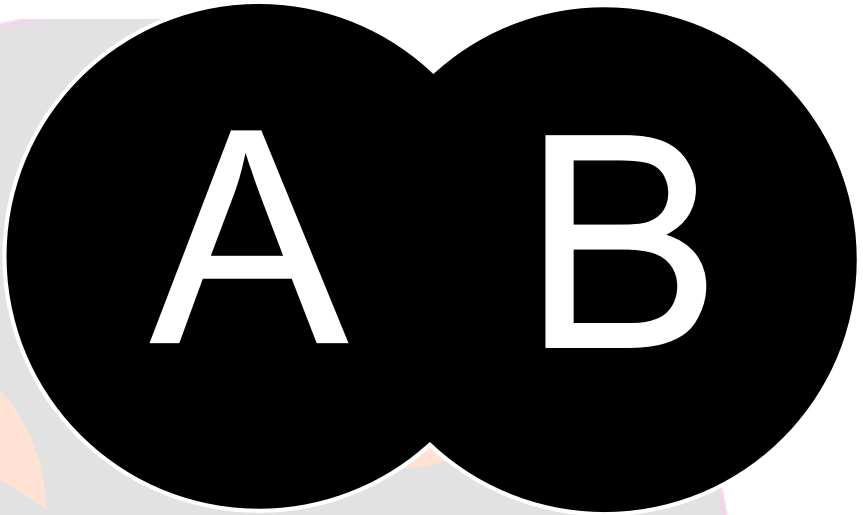
Clear

- To remove all the items from the set

Set operations

- Mathematical operations like union, intersection, difference, symmetric difference
- We can either use operators or methods

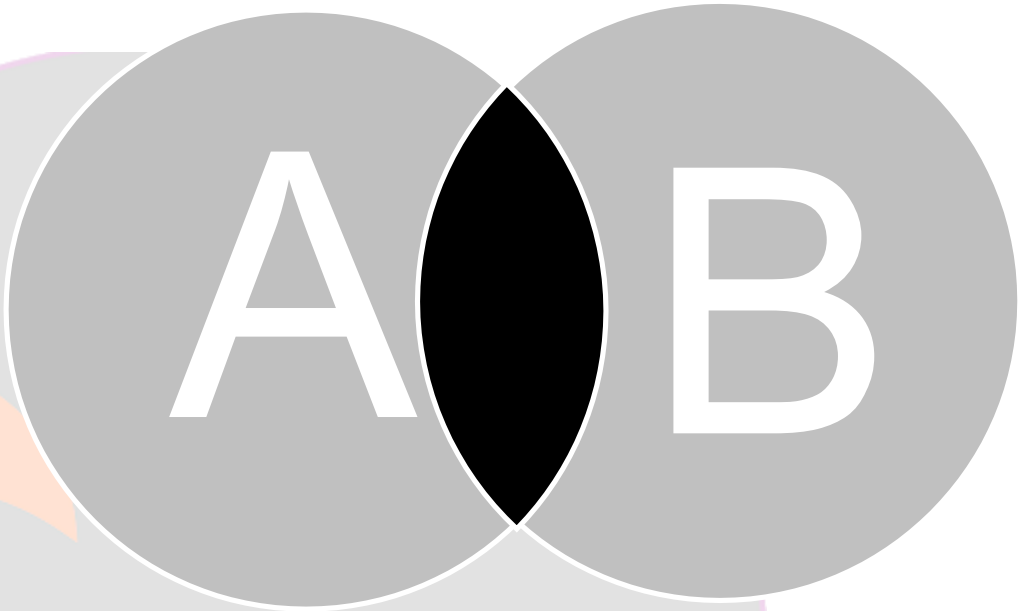
Union



- a = {10,20,30,40}
- b = {20,80,70,100}

- Set of all the elements from both the sets

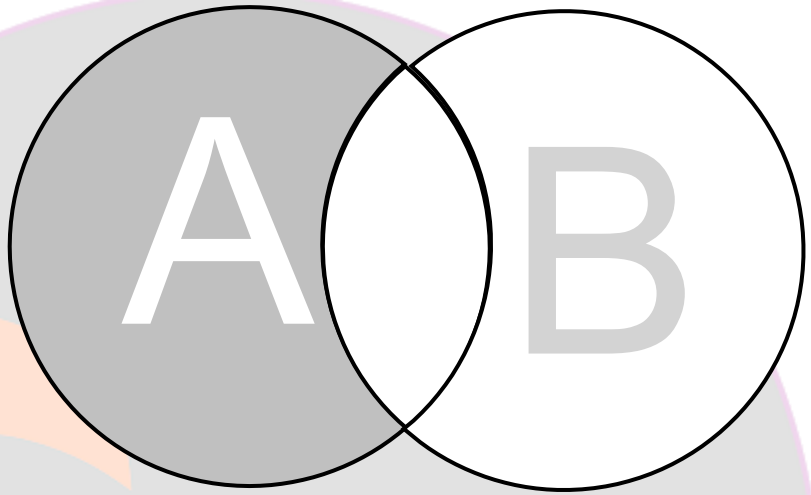
Intersection



- a = {10,20,30,40}
- b = {20,80,70,100}

- Set of elements that are common in both the sets

Difference

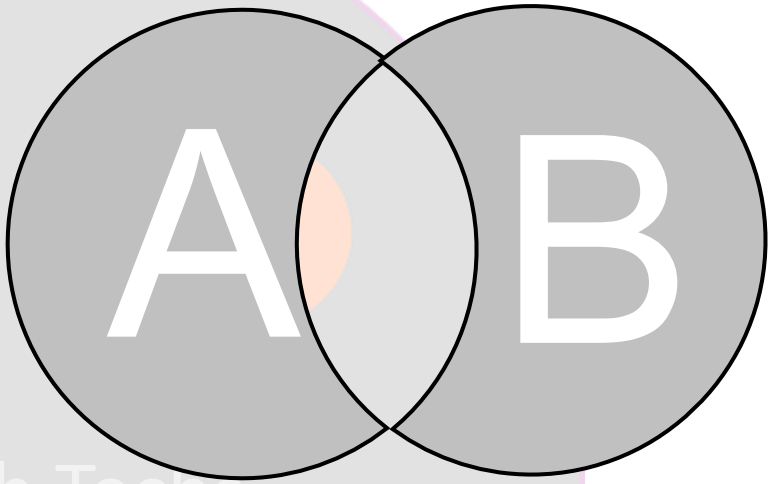


- a = {10,20,30,40}
- b = {20,80,70,100}

- **a – b** (set of elements that are only in **a** but not in **b**)
- **b – a** (set of elements that are only in **b** but not in **a**)

Symmetric difference

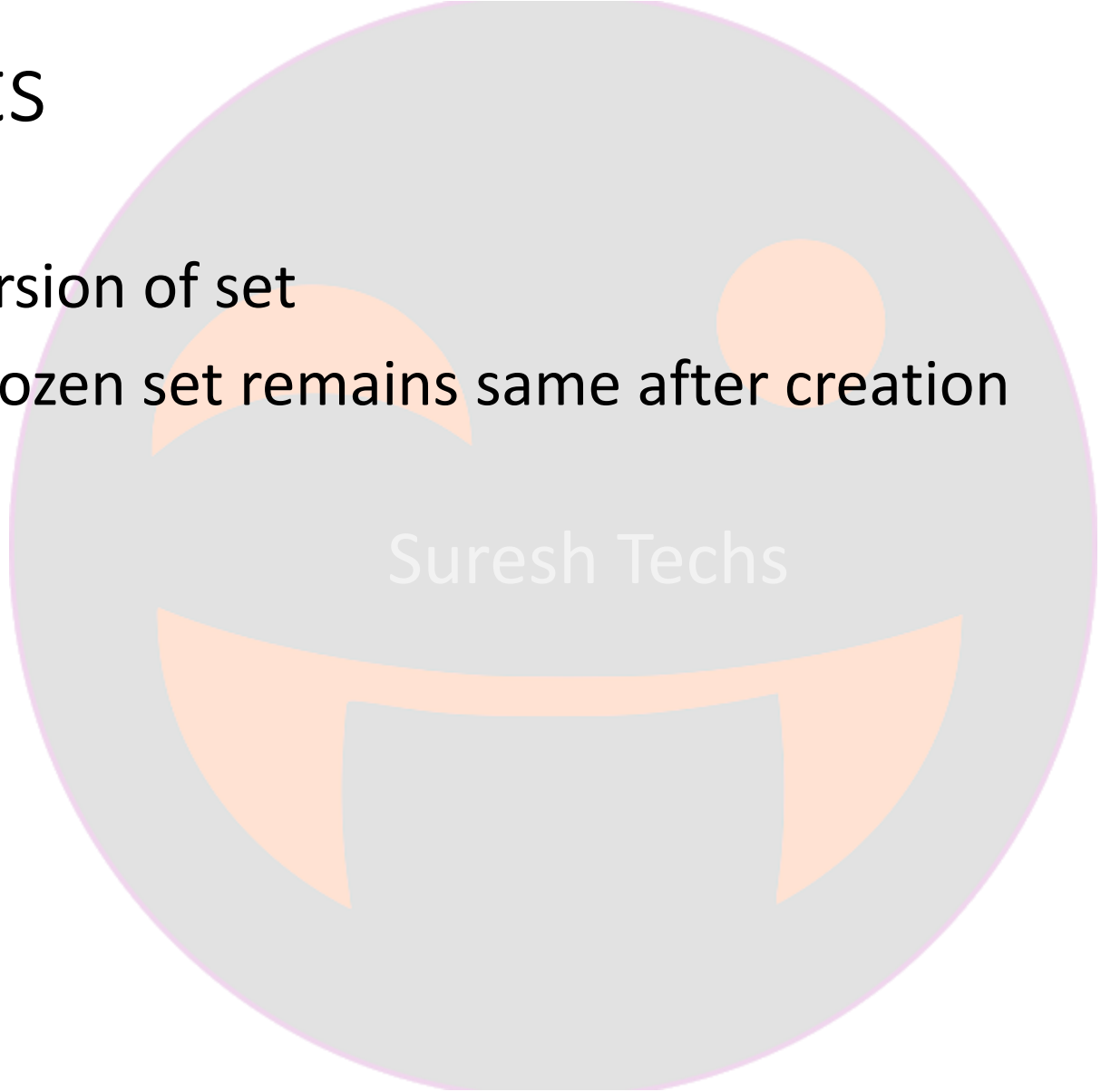
- a = {10,20,30,40}
- b = {20,80,70,100}



- Symmetric difference of **a** and **b** is a set of elements in **a and b but not in both**
- ^
- symmetric_difference()

Frozen sets

- Immutable version of set
- Elements of frozen set remains same after creation



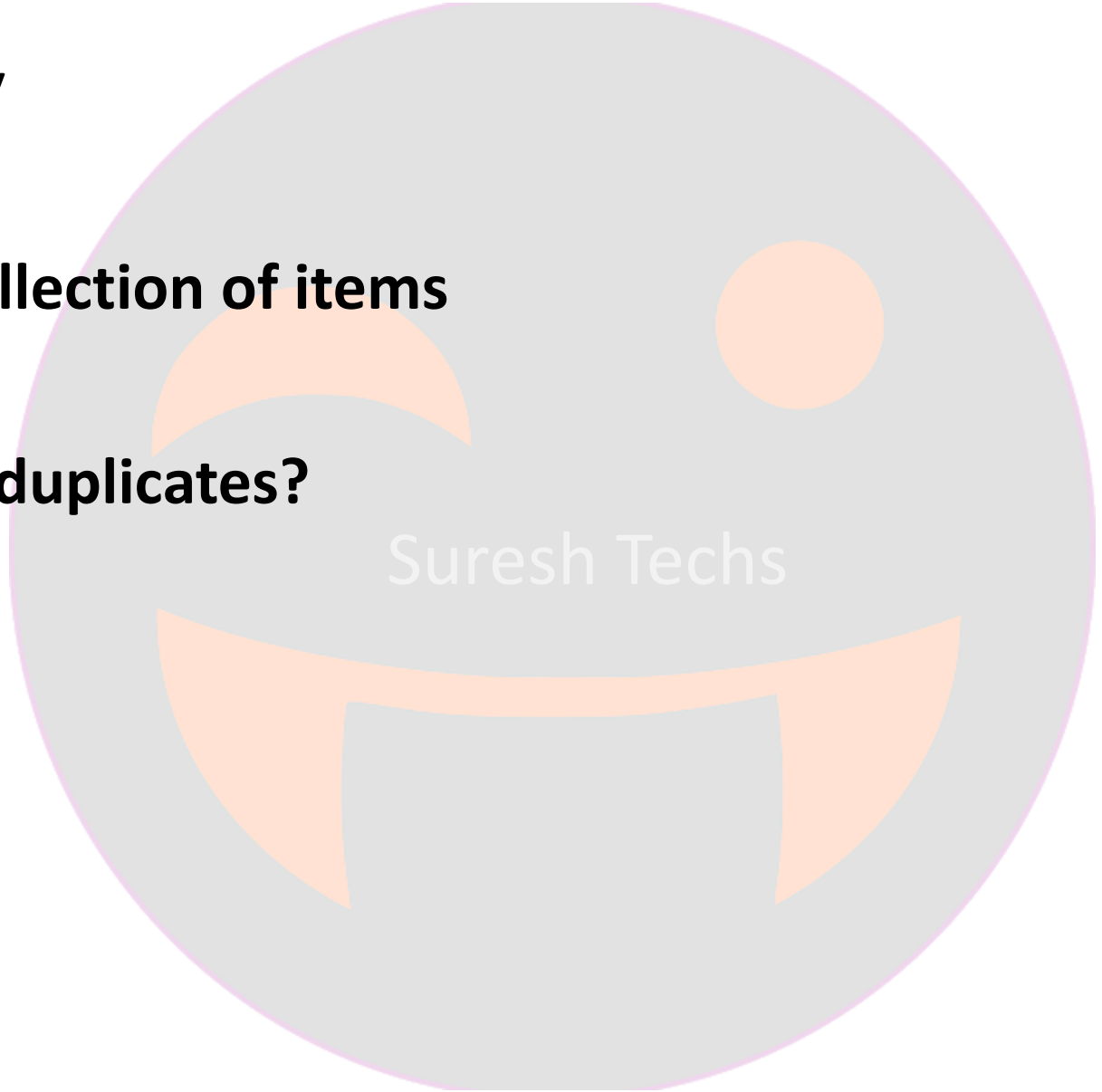
Built In data types In Python

- Numeric (int, float, complex)
- Sequence Types (strings, list, tuple)
- Boolean
- Set
- **Dictionary**

Dictionary

Dictionary

- **Unordered collection of items**
- **Mutable**
- **Does it allow duplicates?**



When do we use dictionary?

- If we want to know the meaning of some unknown word
- We search meaning of word based on character(this character will act as key here)

- Dictionary holds **key : value** pair instead of a single element like other data types

Creating dictionary

- Dictionaries can be created by placing elements inside {} curly braces
- Syntax: friends**={key:value, key:value,key:value}**

- **Values can be** of any type and can be **duplicated**, where as **key's can't be repeated/duplicated**
- Keys must be unique and must be immutable like (string, number, or a tuple with immutable elements)
- We can also use dict() to create a dictionary

Nested Dictionary

- Creating dictionary inside a dictionary is called nested dictionary
- nestedDic = {1:'one',2:{20:'twenty',40:'fourty'}}

Accessing elements from dictionary

- We will use keys to get the values
- Keys can be used either with square brackets [ ] or get() method

- [ ] – key error will raise if the key is not found
- get() – returns None if the key is not found

Nested dictionaries question from one of our family member

Mounika Kondapalli

to me

Hi,

Can you help me with nested dictionaries?

```
D1={1: {"Rollno":12, "class":"B.com FY", "precentage":78.50 },
    2: {"Rollno":14, "class":"B.com FY", "precentage":78.70},
    3: {"Rollno":15, "class":"B.com FY", "precentage":88.50}
   }
```

I need to get the output to enter a roll number
and if percentage > 45 it needs to print "you passed" else failed

Adding/Updating dictionary elements

- **Mutable**
- If the key is present, then the value get's updated
- If the key is not present, then new **key : value** pair will be added
- We can use assignment operator (=) to add/update elements

Removing elements

- del dictionary[key]
- Deleting an element without an existing key will throw error

- pop(key) removes and returns the value
  - poped=nestedDic.pop(2)
  - print(poped)

- popitem() – will remove and returns an arbitrary item from the dictionary
- clear() – removes all the items at once

Built In data types In Python

- Numeric (int, float, complex)
- Sequence Types (strings, list, tuple)
- Boolean
- Set
- Dictionary

We can even create our own data types

Share, Like, Comment, Subscribe

- PLEASE Family!
- Control Flows
- **Interview questions on Data Types**

PYTHON

చిన్న బ్రేక్

చిటికలో వచ్చేస్తా

#sureshtechs