FEATURES OF OOP:

1] Encapsulation- binding of data and procedure as a single unit

2] Data Hiding – who can access the data

3] Abstraction – the way you view an object

4] Polymorphism – many forms

5] Inheritance – acquiring the property of parent object

Generalization → Specialization : Reusability, info will not be cluttered

Collaboration → Containership / Composition : Object has another obj

LIST COMPREHENSION: (square brackets are used)

A concise way of defining and creating a list within a single line of code.

It always creates a new list by evaluating the expression in the context of for and if clauses which follows it.

THIS IS FASTER THAN A LIST USING FOR LOOP

NESTED LIST COMPREHENSION:

It's a way to combine not only one, but multiple for loops , if statements and functions into a single line of code. (list of lists)

INHERITANCE:

Obtaining and acquiring the properties of another class is known as Inheritance.

The inheriting class is called Subclass (a.k.a derived class / child class)

The class inherited from is called Superclass(a.k.a base class/parent class)

ADVANTAGES OF INHERITANCE:

1. Represents Real world relationships between types
2. Provides reusability of a code
3. Simple, understandable model structure
4. Transitive in nature
5. Less development and maintenance expenses

RELATIONSHIP BETWEEN CLASSES:

1) "Is a" Relationship –
   One class is the parent of other class or classes

   a. Action on child imply an action on the parent.
   b. Action on the child overrides the action on the parent.
   c. Action on the child alters the action on the parent.

   Types = Single Level Inheritance

   - One class inherits the features of another class

   Multi-Level Inheritance

   - At any one point, direct parent is only 1 type

   Multiple Inheritance

   - A class can have more than one super class and inherits features from all the parent classes

   Hierarchical Inheritance

   - One class serves as super class for more than one subclasses

   Hybrid Inheritance

   - A mix of two or more of the above types of inheritance . A.k.a Diamond shaped Inheritance

USAGE OF super() – It provides a way to access methods and properties from a parent or superclass within a subclass

__bases__ - An attribute on each class that can be used to obtain a list of classes the given class inherits

Type() – returns the type of the object

"Has a" Relationship:

One class contains the object of other class as its attribute

[When there is no change in output – Delegation/Forwarding]

EXCEPTION:

When an exception is raised, exception-propagation mechanism takes control.

If an exception is thrown or raised inside a function, it can be handled in two ways:
o Inside the function
o Outside the function

The function call is a part of try block, any exception raised during the execution of the function is handled in the corresponding except block.

FUNCTIONS:
Filter() – returns a filter obj where items are filtered thru a func to test if item is accepted or not

Characteristics:

- Input : An iterable of some number of elements(say n)
- Output: A lazy iterator of 0 to n elements(between 0 and n)
- Elements in the Output: apply callback on each element of the iterable

Reduce() - Available in “functools” module. This func applies a given func to the elements of an iterable, reducing them to a single value.

Characteristics:

- Input: An iterable of some number of elements(say n)
- Output: A single element
- Processing: Requires a callback which takes 2 elements

POLYMORPHISM:
It refers to an object’s ability to take on multiple forms with the same name.

Polymorphism with operators – Operator Overloading means giving extended meaning beyond their predefined operational meaning.
The same built-in operator shows different behavior for objects of different classes, this is called Operator Overloading.

Method overriding in Polymorphism – This is an ability of lang that allows a subclass to provide specific implementation of method that is already provided by its parent class.

PARADIGM - It is a methodology to solve some problems or to do some tasks. A programming paradigm is an approach to solve the problem using some programming lang or a method to solve a problem using tools and techniques that are available to us.

Advantages of Functional Programming

- Simple to understand
- Making debugging and testing easier
- Enhances the comprehension and readability of the code

Map() - Returns a map object(which is an iterator) of to each item of a given iterable.

Lambda – is used to create a func without any name for it (aka throwaway func or anonymous func)

Class – it specifies a set of instance variables and methods that are "bundled together" for defining a user defined type.
Object - An object is an instance of a type/class.

CONSTRUCTOR : It is used to define the attributes of an instance and assign values to them.

TYPES OF CONSTRUCTORS:
1] Non parametrized constructor – only self is the parameter
2] Parametrized constructor – Arguments are sent in the func that have same name as the class name

DESTRUCTORS : A reference to objects is also deleted when the object goes out of
reference or when the program ends.

GETTER AND SETTERS : The main purpose of using getters and setters in oop is data encapsulation.
Getter: A function which returns the value of an instance variable
Setter: A function which sets the value to an instance variable

➢ A static method is a method which is bound to the class and not the object of the class. It is present in a class because it makes sense for the method to be present in class.
static method does not receive an implicit first argument.

Features of static method:

- Allocated memory once when the object for the class
- Created outside of methods but inside a class
- It can be accessed through a class but not directly with an instance if no decorator called

- Behavior doesn’t change for every object

ADVANTAGES OF STATIC METHOD:
1] Memory efficiency
2] Shared state
3] Easy to access
4] Initialization
5] Readability

TYPES OF ERRORS:
1] Syntax error / Parsing error – Due to incorrect format of python statement
2] Runtime error / Exceptions – Occurs when a syntactically correct code raises an error. (or) It is an event that occurs during the execution of a program which disrupts of the normal flow of execution of the program

There are 2 types of Exceptions:

a) Built-in exception (Python3 defines 63 built in exceptions)
b) User defined Exception

The finally block:
This block of code is always executed.