Polymorphism in python lets you use the same function name for different types of objects

* This means one function or method can act in many forms.
* It makes your code flexible and easy to manage.
* It helps you write code that can work with various objects.
* You do not need to know their exact type.
* This saves time.
* 3 Ways to use polymorphism in python
* It also makes your programs more adaptable.
* You can use polymorphism in python in a few Ways:
* With Built –in Function :Python’s own functions often use polymorphism.
* With Class Methods:Different classes can have methods that share the same name.
* With inheritance (Method Overriding); Child classes can change methods they get from their parent classes.

## Polymorphism with Built-in Functions

Many built-in python functions work in different ways. Their action changes based on the type of data you give them . The len() function is a good example.

When you use len() on a string.it counts the characters.

When you use len() on a list,it counts the items.

When you use len() on a dicitionary,it coulds the keys.

# Use len()with a string

My string=”hello”

Print(len(my string))

# use len() with a list

My list =[1,2,3,4]

Print(len(my list))

#Use len() with a dicitionary

My dict=[‘a”:1,”b”:2}

Print(len(my dict))