**DAY-8 ASSESSMENT**

# **INHERITANCE QUESTIONS**

1. class Person:  
 def display\_name(self):  
 print("name:Vidya")  
class Student(Person):  
 pass  
obj=Student()  
obj.display\_name()

2. class Animal:  
   def eat(self):  
       print("Animal eats")  
  
class Mammal(Animal):  
   def walk(self):  
       print("Mammal walks")  
  
class Dog(Mammal):  
   def bark(self):  
       print("Dog barks")  
  
d = Dog()  
d.eat()  
d.walk()  
d.bark()

3. class Flyable:  
   def fly(self):  
       print("Can fly")  
  
class Swimmable:  
   def swim(self):  
       print("Can swim")  
  
class Duck(Flyable, Swimmable):  
   pass  
  
d = Duck()  
d.fly()  
d.swim()

4. class Vehicle:  
   def start(self):  
       print("Vehicle started")  
  
class Car(Vehicle):  
   def drive(self):  
       print("Driving car")  
  
class Bike(Vehicle):  
   def ride(self):  
       print("Riding bike")  
  
c = Car()  
c.start()  
c.drive()  
  
b = Bike()  
b.start()  
b.ride()

5. class Parent:  
   def greet(self):  
       print("Hello from Parent")  
  
class Child(Parent):  
   def greet(self):  
       super().greet()  
       print("Hello from Child")  
  
c = Child()  
c.greet()

6. class A:  
   def show(self):  
       print("A")  
  
class B(A):  
   def show(self):  
       print("B")  
  
class C(A):  
   def show(self):  
       print("C")  
  
class D(B, C):  # Diamond inheritance  
   pass  
  
d = D()  
d.show()  
print(D.\_\_mro\_\_)

 # MRO: D → B → C → A → object

7. class Person:  
   def \_\_init\_\_(self, name):  
       self.name = name  
  
class Student(Person):  
   def \_\_init\_\_(self, name, roll\_no):  
       super().\_\_init\_\_(name)  
       self.roll\_no = roll\_no  
  
s = Student("Vidya", "S01")  
print(s.name, s.roll\_no)

8. class Shape:  
   def area(self):  
       return 0  
  
class Circle(Shape):  
   def \_\_init\_\_(self, radius):  
       self.radius = radius  
  
   def area(self):  
       return 3.14 \* self.radius \*\* 2  
  
c = Circle(5)  
print("Area:", c.area())

9. class Animal:  
   def speak(self):  
       print("Animal sound")  
  
class Dog(Animal):  
   def speak(self):  
       print("Bark")  
  
class Cat(Animal):  
   def speak(self):  
       print("Meow")  
  
d = Dog()  
c = Cat()  
d.speak()  
c.speak()

10. animals = [Dog(), Cat(), Animal()]  
  
for a in animals:  
   a.speak()

11. class Demo:  
   def show(self, \*args):  
       for arg in args:  
           print(arg)  
  
d = Demo()  
d.show(10)  
d.show(10, 20)

12. class Calculator:  
   def add(self, a, b, c=0):  
       return a + b + c  
  
c = Calculator()  
print(c.add(2, 3))       # 5  
print(c.add(2, 3, 4))    # 9

 13. class Book:  
   def \_\_init\_\_(self, title):  
       self.title = title  
  
   def \_\_str\_\_(self):  
       return f"Book Title: {self.title}"  
  
b = Book("Python Mastery")  
print(b)  # calls \_\_str\_\_()

14. class Car:  
   def start(self):  
       print("Car started")  
  
class Bike:  
   def start(self):  
       print("Bike started")  
  
def start\_engine(vehicle):  
   vehicle.start()  
  
start\_engine(Car())  
start\_engine(Bike())

15. def calculate\_area(shape):  
   return shape.area()  
  
class Rectangle:  
   def area(self):  
       return 10 \* 5  
class Circle:  
   def area(self):  
       return 3.14 \* 7 \*\* 2  
shapes = [Rectangle(), Circle()]  
for s in shapes:  
   print(calculate\_area(s))