

Single-Level-Inheritance :--

1. Zepto – Grocery Delivery

```
class Order:
    def __init__(self, order_id, items, amount):
        self.order_id = order_id
        self.items = items
        self.amount = amount
    def show_order(self):
        tax = 50 # local variable
        print(f"Order ID: {self.order_id}")
        print(f"Items: {self.items}")
        print(f"Amount: {self.amount}")
        print(f"Tax: {tax}")
class Delivery(Order):
    def show_delivery(self):
        self.show_order()
        print("Delivery Status: Out for delivery")
d1 = Delivery("ZP123", ["Milk", "Bread"], 250)
d1.show_delivery()
```

output:

Order ID: ZP123

Items: ['Milk', 'Bread']

Amount: 250

Tax: 50

Delivery Status: Out for delivery

2. Amazon – E-commerce

```
class Product:
```

```
    def __init__(self, name, price, category):
```

```
        self.name = name
```

```
        self.price = price
```

```
        self.category = category
```

```
    def show_product(self):
```

```
        platform = "Amazon"
```

```
        print(f"Product Name: {self.name}")
```

```
        print(f"Price: {self.price}")
```

```
        print(f"Category: {self.category}")
```

```
        print(f"Platform: {platform}")
```

```
class DiscountedProduct(Product):
```

```
    def show_discount(self, discount_percentage):
```

```
        self.show_product()
```

```
        final_price = self.price - (self.price * discount_percentage / 100)
```

```
        print(f"Discount: {discount_percentage}%")
```

```
        print(f"Final Price after Discount: {final_price}")
```

```
dp1 = DiscountedProduct("Smartphone", 30000, "Electronics")
```

```
dp1.show_discount(10)
```

output:

Product Name: Smartphone

Price: 30000

Category: Electronics

Platform: Amazon

Discount: 10%

Final Price after Discount: 27000.0

3. Uber – Ride Booking

class Ride:

```
def __init__(self, ride_id, pickup, drop):
```

```
    self.ride_id = ride_id
```

```
    self.pickup = pickup
```

```
    self.drop = drop
```

```
def show_ride(self):
```

```
    distance = 12
```

```
    print(f"Ride ID: {self.ride_id}")
```

```
    print(f"Pickup Location: {self.pickup}")
```

```
    print(f"Drop Location: {self.drop}")
```

```
    print(f"Distance: {distance} km")
```

class Driver(Ride):

```
def show_driver(self):
```

```
    self.show_ride()
```

```
    print("Driver Status: Arriving soon")
```

```
driver1 = Driver("UBR567", "Airport", "Hotel")
```

```
driver1.show_driver()
```

output:

Ride ID: UBR567

Pickup Location: Airport

Drop Location: Hotel

Distance: 12 km

Driver Status: Arriving soon