

5. Online Payment System

Program :

```
from abc import ABC, abstractmethod
```

```
# Abstraction
```

```
class Payment(ABC):
```

```
    def __init__(self, amount):
```

```
        self.amount = amount
```

```
    @abstractmethod
```

```
    def pay(self):
```

```
        pass
```

```
# Inheritance
```

```
class UPI(Payment):
```

```
    def __init__(self, amount):
```

```
        super().__init__(amount)
```

```
# Polymorphism
```

```
def pay(self):
```

```
    print(f"Paid ₹{self.amount} using UPI")
```

```
class Card(Payment):
```

```
def __init__(self, amount):
    super().__init__(amount)

# Polymorphism
def pay(self):
    print(f"Paid ₹{self.amount} using Card")

class Cash(Payment):
    def __init__(self, amount):
        super().__init__(amount)

    # Polymorphism
    def pay(self):
        print(f"Paid ₹{self.amount} using Cash")

# ---- Main Program ----

mode = input("Enter Payment Mode (UPI/Card/Cash): ")
amount = float(input("Enter Amount: "))

if mode.lower() == "upi":
    payment = UPI(amount)
elif mode.lower() == "card":
    payment = Card(amount)
elif mode.lower() == "cash":
    payment = Cash(amount)
```

```
else:  
    print("Invalid Payment Mode")  
    exit()  
  
payment.pay()
```

Output :

```
Enter Payment Mode (UPI/Card/Cash): UPI  
Enter Amount: 100  
Paid ₹100.0 using UPI
```