POLYMORPHISM :-

Many

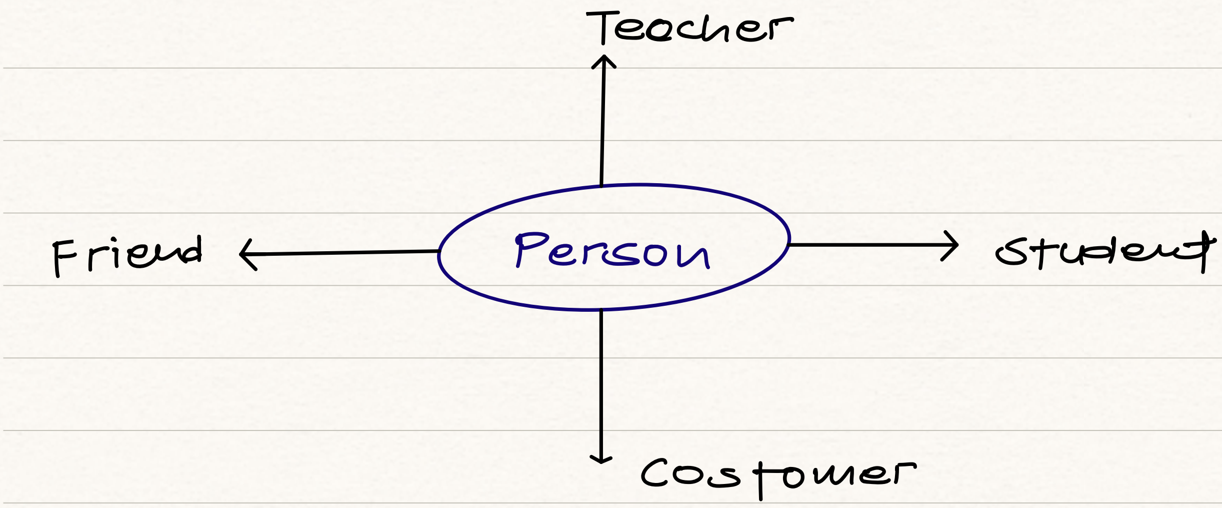
Ways to store
(FORM)

It simply means more than one Form. That is, the same entity can perform different operations in different scenarios.

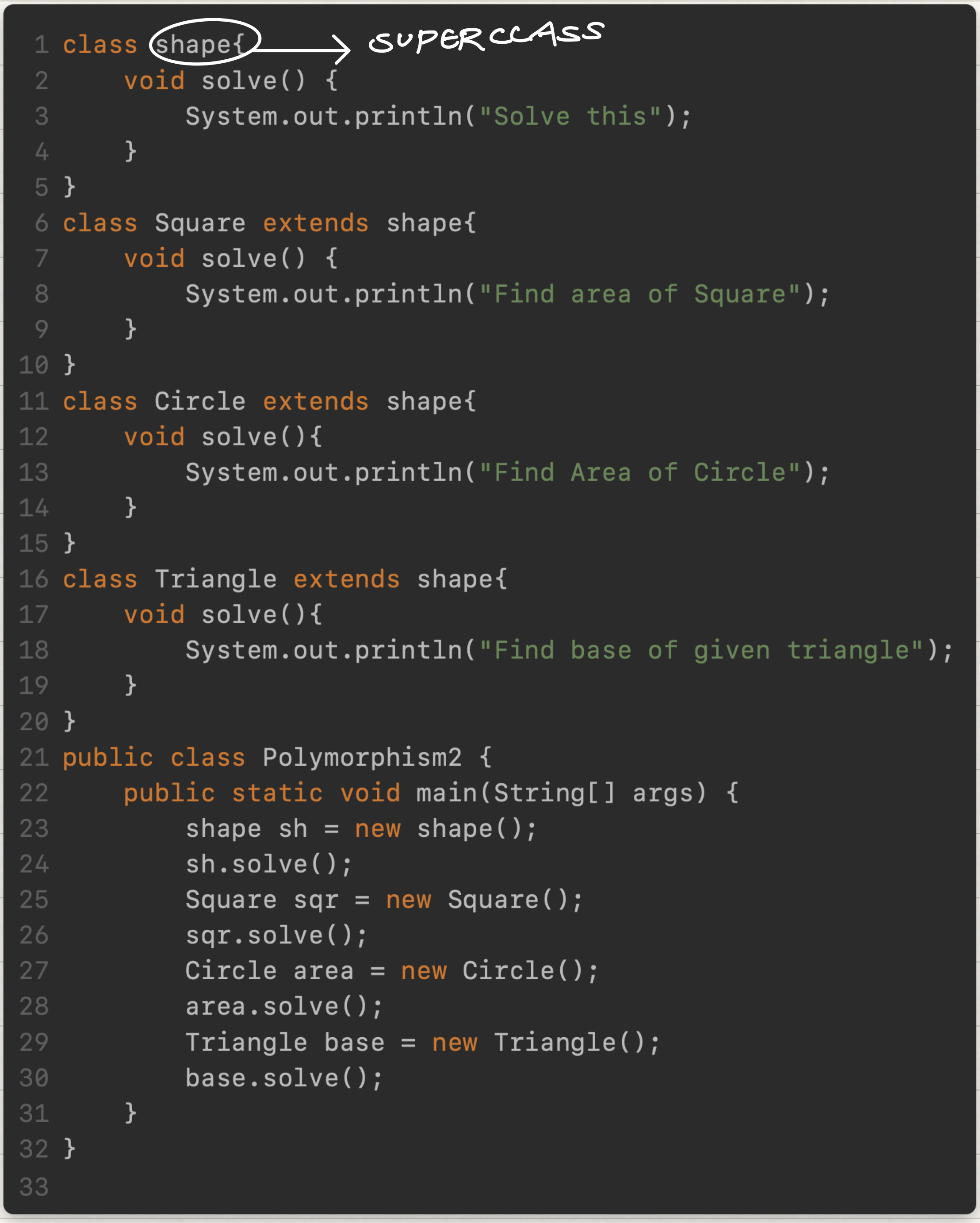
OR

It simply means same object having different behaviour.

FOR EXAMPLE :-



PROGRAM:-



```
class shape{
    void solve() {
        System.out.println("Solve this");
    }
}
class Square extends shape{
    void solve() {
        System.out.println("Find area of Square");
    }
}
class Circle extends shape{
    void solve(){
        System.out.println("Find Area of Circle");
    }
}
class Triangle extends shape{
    void solve(){
        System.out.println("Find base of given triangle");
    }
}
public class Polymorphism2 {
    public static void main(String[] args) {
        shape sh = new shape();
        sh.solve();
        Square sqr = new Square();
        sqr.solve();
        Circle area = new Circle();
        area.solve();
        Triangle base = new Triangle();
        base.solve();
    }
}
```

WHY POLYMORPHISM:-

→ allows us to create consistent code. In above code, For every shape, we need to create different methods. It will make our code inconsistent. To solve this, polymorphism in Java allows us to create a single method that will behave differently For different shapes.
OR
objects.

NOTE:- print() method is also an example of polymorphism. Because it is used to print values of different data types like; int, char, Float, double, string etc.

WE CAN ACHIEVE POLYMORPHISM IN JAVA using the Following ways:

1. Method overriding
2. Method overloading
3. Operator Overloading

METHOD-OVERRIDING:- If subclass (child class) has the same method as declared in parent class, it is known as method overriding.

USES:-

* Method overriding is used to provide the specific implementation of a method which is already provided by its superclass.

* Method overriding is used for runtime polymorphism.

RULES:-

1. The method must have the same name as in the parent class.

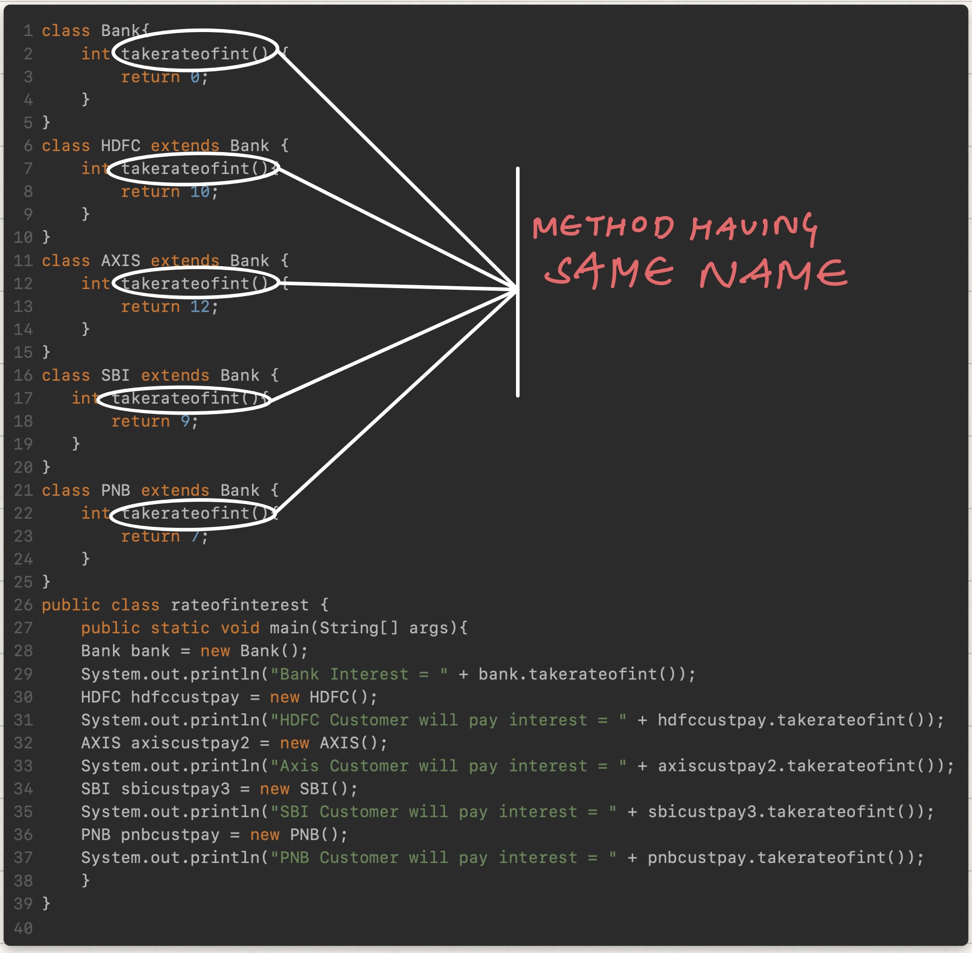
2. The method must have the same parameter as in the parent class.

3. There must be an IS-A relationship (inheritance).

NOTE:-

Static method cannot be overriden. It is because the static method is bound with class whereas instance method is bound with an object. Static belongs to the class area, and an instance belongs to the heap area.

PROGRAM:-



```
class Bank{
    int takerateofint(){
        return 0;
    }
}
class HDFC extends Bank {
    int takerateofint(){
        return 10;
    }
}
class AXIS extends Bank {
    int takerateofint(){
        return 12;
    }
}
class SBI extends Bank {
   int takerateofint(){
       return 9;
   }
}
class PNB extends Bank {
    int takerateofint(){
        return 7;
    }
}
public class rateofinterest {
    public static void main(String[] args){
    Bank bank = new Bank();
    System.out.println("Bank Interest = " + bank.takerateofint());
    HDFC hdfccustpay = new HDFC();
    System.out.println("HDFC Customer will pay interest = " + hdfccustpay.takerateofint());
    AXIS axiscustpay2 = new AXIS();
    System.out.println("Axis Customer will pay interest = " + axiscustpay2.takerateofint());
    SBI sbicustpay3 = new SBI();
    System.out.println("SBI Customer will pay interest = " + sbicustpay3.takerateofint());
    PNB pnbcustpay = new PNB();
    System.out.println("PNB Customer will pay interest = " + pnbcustpay.takerateofint());
    }
}
```

METHOD OVERLOADING:- If a class has multiple methods having same name but different in parameters, it is known as Method overloading.

*. If we have to perform only one operation, having same name of the methods increases the readability of the program.

ADVANTAGE:-

Method overloading increases the readability of the program.

DIFFERENT WAYS TO OVERLOAD THE METHOD:

1. By changing number of arguments.
2. By changing the data type.

Can we overload java main() method?

→ Yes, by method overloading. You can have any number of main methods in a class by method overloading. But JVM calls main() method which receives string array as argument only.