**What is Polymorphism**

The dictionary definition of polymorphism refers to a principle in biology in which an organism or species can have many different forms or stages. This principle can also be applied to object-oriented programming and languages like the Java language.

Remember any java object that can pass more than one IS-A test can be considered polymorphic.

Other than object of type object, all java objects are polymorphic in that they pass the IS-A test for their own type and for class object.

Polymorphism is an Oops concept which advice use of common interface instead of concrete implementation while writing code

When we program for interface our code is capable of handling any new requirement or enhancement arise in near future due to new implementation of our common interface. If we don't use common interface and rely on concrete implementation, we always need to change and duplicate most of our code to support new implementation.

Exmple :

**public** **class** IT {

**private** String name;

**private** String post;

**protected** **double** salary;

**public** IT(String name, String post, **double** salary) {

**this**.name = name;

**this**.post = post;

**this**.salary = salary;

}

**public** **double** salaryadd(){

**return** salary;

}

}

**public** **class** TeamLaeder **extends** IT {

**private** **double** extra;

**public** TeamLaeder(String name, String post, **double** salary) {

**super**(name, post, salary);

}

**public** TeamLaeder(String name, String post, **double** salary,**double** extra) {

**super**(name, post, salary);

**this**.extra=extra;

}

@Override

**public** **double** salaryadd() {

salary=salary+extra;

**return** salary ;

}

@Override

**public** String toString() {

**return** "TeamLaeder [extra=" + extra + ", salary=" + salary

+ "]";

}

}

**public** **class** Developer **extends** IT {

**private** **double** rise;

**public** Developer(String name, String post, **double** salary) {

**super**(name, post, salary);

}

**public** Developer(String name, String post, **double** salary,**double** rise) {

**super**(name, post, salary);

**this**.rise=rise;

}

**public** **double** salaryadd() {

salary=salary+rise;

**return** salary;

}

@Override

**public** String toString() {

**return** "Developer [rise=" + rise + ", salary=" + salary+"]";

}

}

**public** **class** ITTest {

**public** **static** **void** main(String[] args) {

TeamLaeder teamLaeder=**new** TeamLaeder("parvez", "senior", 30000, 3000);

System.*out*.println(teamLaeder);

System.*out*.println("New Salary = "+teamLaeder.salaryadd());

Developer developer=**new** Developer("sudip", "jounir", 10000, 5000);

System.*out*.println(developer);

System.*out*.println("New Salary = "+developer.salaryadd());

}

}