**This & Super Key word**

**This Keyword**

1. This keyword is used **to refer current class instance variable**. It resolves the ambiguity between parameterized and initialized instance variable.

Program

public class Student {

int rollno;

String name ;

String City ;

public Student(int rollno,String name,String City)

{

this.rollno =rollno;

this.name = name;

this.City =City;

}

public void display(){

System.out.println("Roll no: "+rollno+", Name : "+name+ ", City: "+City);

}

public static void main(String[] args) {

Student s1 = new Student(12,"Vrushali","Panvel");

s1.display();

}

}

OUTPUT

Roll no: 12, Name : Vrushali, City: Panvel

NOTE : without this keyword , output will be

**Roll no: 0, Name : null, City: null**

1. **This: to invoke current class method.**

**You may invoke method of current class by using this. Keyword, if you don’t use this keyword, compiler automatically adds this keyword while invoking method.**

**public class Student {**

**void display(){**

**System.out.println("Display :=> ");**

**}**

**void exam(){**

**System.out.println("Exam :=> ");**

**this.display();**

**}**

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

**{**

**new Student().exam();**

**}**

**OUTPUT**

Exam :=>

Display :=>

1. **This () : this to invoke current class constructor.**

**This () constructor is used to invoke current class instructor. It is used to reuse constructor. In other word, it is used for constructor chaining.**

**import java.util.\*;**

**public class Vehicle {**

**int seats ;**

**double fuel ;**

**public Vehicle(){**

**System.out.println(" Default constructor");**

**}**

**public Vehicle(int seats , double fuel)**

**{**

**this();**

**this.seats = seats;**

**this.fuel = fuel;**

**}**

**void display()**

**{**

**System.out.println("Seat := "+seats+ "fuel : "+ fuel);**

**}**

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

**Vehicle v1 = new Vehicle(5, 1.5);**

**v1.display();**

**}**

**}**

Output:

Default constructor

Seat := 5fuel : 1.5

**Error case 1 : when we declares constructor for default constructor**

public class Vehicle {

int seats ;

double fuel ;

public Vehicle(){

this();

System.out.println(" Default constructor");

}

public Vehicle(int seats , double fuel)

{

this.seats = seats;

this.fuel = fuel;

}

void display()

{

System.out.println("Seat := "+seats+ "fuel : "+ fuel);

}

public static void main(String[] args) {

Vehicle v1 = new Vehicle(5, 1.5);

v1.display();

OUTPUT

Vehicle.java:8: error: recursive constructor invocation

public Vehicle(){

**Error case 2 : when this() constructor not written first**

public class Vehicle {

int seats ;

double fuel ;

public Vehicle(){

System.out.println(" Default constructor");

}

public Vehicle(int seats , double fuel)

{

this.seats = seats;

this.fuel = fuel;

this();

}

void display()

{

System.out.println("Seat := "+seats+ "fuel : "+ fuel);

}

public static void main(String[] args) {

Vehicle v1 = new Vehicle(5, 1.5);

v1.display();

}

}

OUTPUT

Vehicle.java:19: error: call to this must be first statement in constructor

this();