this, this();

* Used to refer non-static member of the current instance.
* This keyword cann’t be used inside static context.

**package** com.te.datatypes;

**public** **class** ThisIsFun {

**int** a;

**int** b;

**public** **void** setData(**int** a , **int** b) {

**this**.a = a;

**this**.b = b;

// System.out.println(a); //3

// System.out.println(b); //4

}

**public** **void** display() {

System.***out***.println(a); //0

System.***out***.println(b); //0

}

}

**package** com.te.datatypes;

**public** **class** Main {

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

ThisIsFun main = **new** ThisIsFun();

main.setData(3, 4);

main.display();

}

}

**package** com.te.datatypes;

**public** **class** Main {

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

ThisIsFun main = **new** ThisIsFun();

main.setData(3, 4);

main.display();

}

}

Constructor overloading/this();

* Creating multiple constructors in a class with different args.

**package** com.te.variables.common;

**public** **class** Book {

String title;

String author;

**double** price;

**int** pages;

**public** Book(String title, **double** price, **int** pages) {

**this**.title = title;

**this**.price = price;

**this**.pages = pages;

}

**public** Book(String title, **double** price, **int** pages, String author) {

**this**(title, price, pages);

**this**.author = author;

}

**public** **void** display() {

System.***out***.println("Title : " + title);

System.***out***.println("Author : " + author);

System.***out***.println("Price : " + price);

System.***out***.println("Pages : " + pages);

}

}

**package** com.te.variables.common;

**public** **class** MainClass {

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

Book book = **new** Book("c", 500.34 , 2000 ,"Dennis");

book.display();

}

}