**Example 1 :**

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

{

*test*(10);

}

**private** **static**<T> **void** test(T i) {

System.***out***.println(i);

}

**Example 2:**

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

{

*test*(10,"Smith");

}

**private** **static**<T,V> **void** test(T i,V j) {

System.***out***.println(i);

System.***out***.println(j);

}

**Example 3:**

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

{

*test*(10,"Smith");

}

**private** **static**<T,V> **V** test(T i,V j) {

System.***out***.println(i);

return j

}

**public** **class** IntegerPrinter {

**int** toPrint;

**public** IntegerPrinter(**int** toPrint) {

**this**.toPrint = toPrint;

}

**public** **void** print()

{

System.***out***.println(toPrint);

}

}

**public** **class** StringPrinter {

String toPrint;

**public** StringPrinter(String toPrint) {

**super**();

**this**.toPrint = toPrint;

}

**public** **void** print()

{

System.***out***.println(toPrint);

}

}

**public** **class** DriverApp {

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

IntegerPrinter ip=**new** IntegerPrinter(9);

ip.print();

StringPrinter sp=**new** StringPrinter("Alice");

sp.print();

}

}

-----

Example 4:

**public** **class** GenericPrinter<T> {

T toPrint;

**public** GenericPrinter(T toPrint) {

**this**.toPrint = toPrint;

}

**public** **void** print()

{

System.***out***.println(toPrint);

}

}

**public** **class** DriverApp {

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

GenericPrinter<Integer> gp=**new** GenericPrinter("Ben");

gp.print();

}

}

---

Example 5:

**public** **class** GenericPrinter<T,V> {

T toPrint;

V anotherPrint;

**public** GenericPrinter(T toPrint, V anotherPrint) {

**this**.toPrint = toPrint;

**this**.anotherPrint = toanotherPrint

}

**public** **void** print()

{

System.***out***.println(toPrint);

System.***out***.println(anotherPrint);

}

}

**public** **class** DriverApp {

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

GenericPrinter<Integer> gp=**new** GenericPrinter(18,"Ben");

gp.print();

}

}

Example 6 : Bounded Generic

**public** **interface** Animal {

String eat();

}

**public** **class** Cat **implements** Animal {

@Override

**public** String eat() {

**return** "Cat Eat";

}

}

**public** **class** Dog **implements** Animal {

@Override

**public** String eat() {

**return** "Dog Eat";

}

}

**public** **class** GenericPrinter<T **extends** Animal> {

T toPrint;

**public** GenericPrinter(T toPrint) {

**this**.toPrint = toPrint;

}

**public** **void** print()

{

System.***out***.println(toPrint.eat());

}

}

**public** **class** DriverApp {

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

GenericPrinter<Animal> gp=**new** GenericPrinter(**new** Dog());

gp.print();

}

}