1. Using HASHSET store data through a class

**class** Employ

{

**int** id;

**float** salary;

String Department;

//constructor of employe

**public** Employ(**int** id,**float** salary,String Department)

{

// System.out.println("Employe\_id-"+id +" Employee salary- "+ salary+ " Employe Department - "+ Department);

**this**.id =id;

**this**.salary=salary;

**this**.Department=Department;

}}

**public** **class** Emplpoyee {

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

HashSet<Employ> h1= **new** HashSet<Employ>();

Employ emp = **new** Employ(101, 454000, "Sumit joshi");

Employ emp1 = **new** Employ(102, 223000, "Jack");

Employ emp2 = **new** Employ(103, 12000, "Giant ");

Employ emp3 = **new** Employ(104, 34000, "SLayer");

h1.add(emp);

h1.add(emp1);

h1.add(emp2);

h1.add(emp3);

// emp=emp.DisplayDeatils(12, 123, "asd");

// h1.add(emp).DisplayDeatils(101, 1000000, "IT"));

System.***out***.println("ID | salary | Department");

**for**(Employ i:h1){

System.***out***.println(i.id+" | "+i.salary+" | "+i.Department);

} }}

1. Storing random number in key and value pair in HASHMAP

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

Random rand = **new** Random();

Map<Integer, Double> map = **new** HashMap<>();

**for**(**int** i =0;i<11;i++)

{

**int** int\_random = rand.nextInt(20);

**double** double\_random=rand.nextDouble();

map.put(int\_random,double\_random );

}

**for**(Map.Entry i:map.entrySet())

{

System.***out***.println(i.getKey()+ " "+ i.getValue());

}

}

1. Swap array with the help of generic method

**class** gen

{

**public** **static** **void** swap(Object[] a, **int** i, **int** j){

Object temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

**public** **class** Emplpoyee {

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

gen g= **new** gen();

String[] a={"7","10"};

System.***out***.println("before: "+Arrays.*toString*(a));

g.*swap*(a,0,1);

System.***out***.println("after: "+Arrays.*toString*(a));

}a

}

1. A> Using generic method made a method for taking key and value…

**class** pair<T1,T2>

{

T1 i;

T2 j;

**void** addkey(T1 i1)

{

i=i1;

}

T1 getkey()

{

**return** i;

}

**void** addvalue(T2 i2)

{

j=i2;

}

T2 getvalue()

{

**return** j;

}}

**public** **class** GenricQ4 {

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

// Map< Integer, String > map = new HashMap<>();

pair<String,String> p1= **new** pair<>();

p1.addkey("name");

p1.addvalue("sss");

System.***out***.println(p1.getkey()+ " "+ p1.getvalue());

}

B> Store Date in value pair …

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

**class** pair<T1,T2>

{

T1 i;

T2 j;

**void** addkey(T1 i1)

{

i=i1;

}

T1 getkey()

{

**return** i;

}

**void** addvalue(T2 i2)

{

j=i2;

}

T2 getvalue()

{

**return** j;

}

}

**public** **class** GenricQ4 {

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

// Map< Integer, String > map = new HashMap<>();

pair<String,Date> p1= **new** pair<>();

p1.addkey("name");

p1.addvalue(**new** java.sql.Date(Calendar.*getInstance*().getTime().getTime()));

System.***out***.println(p1.getkey()+ " "+ p1.getvalue());

}

}