**NAME:ROHAN NYATI**

**SAP ID:500075940**

**ROLL NO. : R177219148**

**BATCH-5(AI&ML)**

**EXPERIMENT NO – 9**

**TITLE:** Collections

**1. Write a program for the following:**

**(i) Read all elements from ArrayList by using Iterator.**

**CODE**

**package** rohan;

**import** java.util.ArrayList;

**import** java.util.Iterator;

**public** **class** ArrayListIterator

{

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

{

ArrayList<String> arrl = **new** ArrayList<String>();

arrl.add("First");

arrl.add("Second");

arrl.add("Third");

arrl.add("Random");

Iterator<String> itr = arrl.iterator();

**while**(itr.hasNext())

{

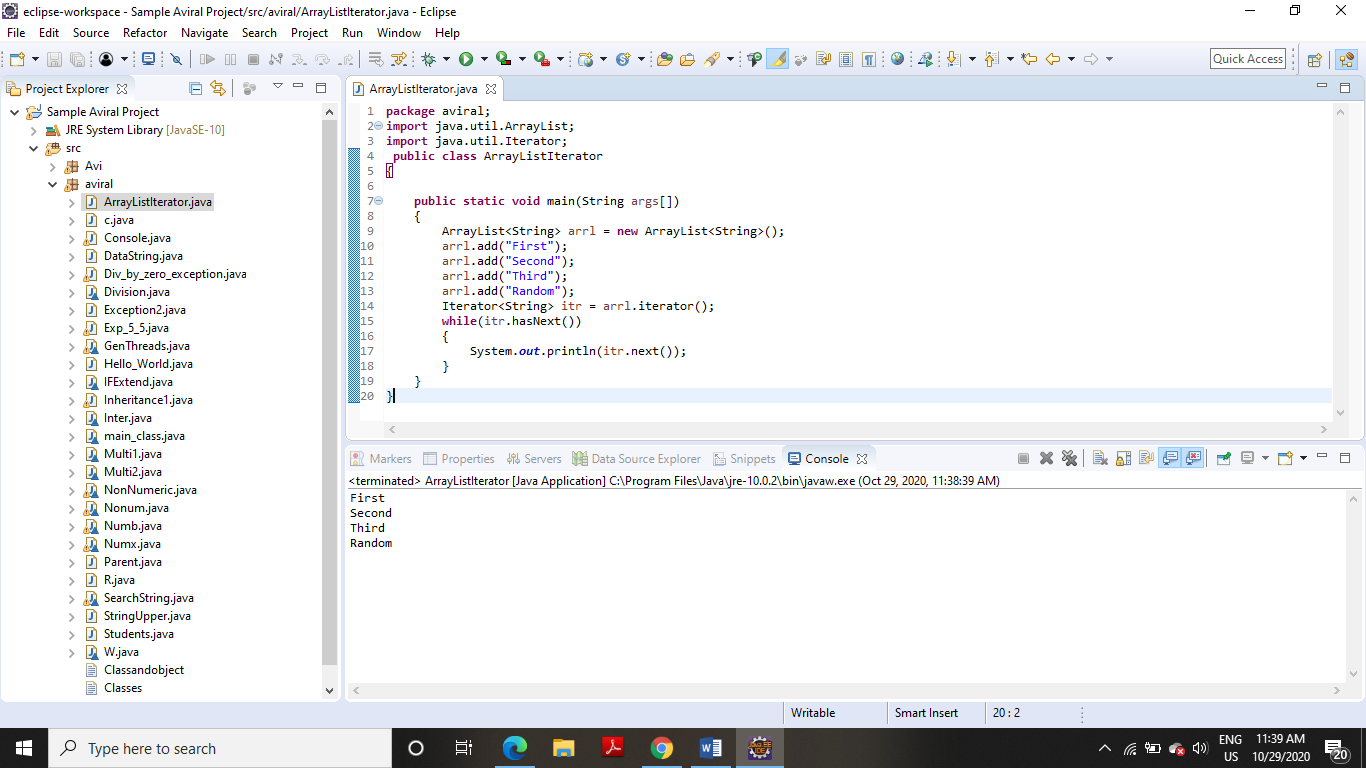
System.***out***.println(itr.next());

}

}

}

**OUTPUT**

****

**(ii) Create duplicate object of an ArrayList instance**.

**CODE**

**package** rohan;

**import** java.io.\*;

**import** java.util.ArrayList;

**public** **class** DuplicateObject

{

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

{

ArrayList<Integer>list= **new** ArrayList<Integer>();

list.add(10);

list.add(20);

list.add(30);

list.add(40);

list.add(50);

System.***out***.println("First ArrayList: "+ list);

ArrayList sec\_list = **new** ArrayList();

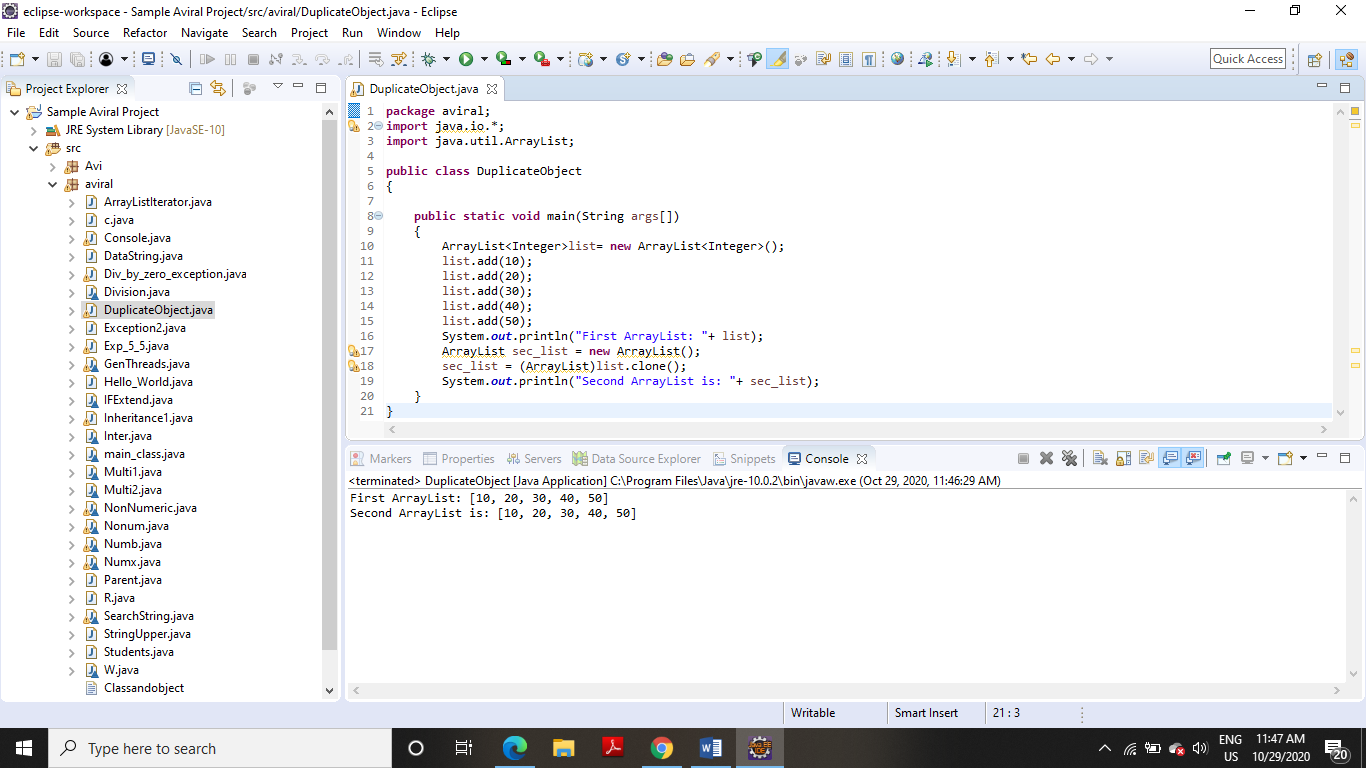
sec\_list = (ArrayList)list.clone();

System.***out***.println("Second ArrayList is: "+ sec\_list);

}

}

**OUTPUT**

****

**(iii) Reverse ArrayList content.**

**CODE**

**package** rohan;

**import** java.util.ArrayList;

**import** java.util.Collections;

**public** **class** ArrayListReverse

{

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

{

ArrayList<String> list = **new** ArrayList<String>();

list.add("Java");

list.add("Cric");

list.add("Play");

list.add("Watch");

list.add("Glass");

Collections.*reverse*(list);

System.***out***.println("Results after reverse operation:");

**for**(String str: list)

{

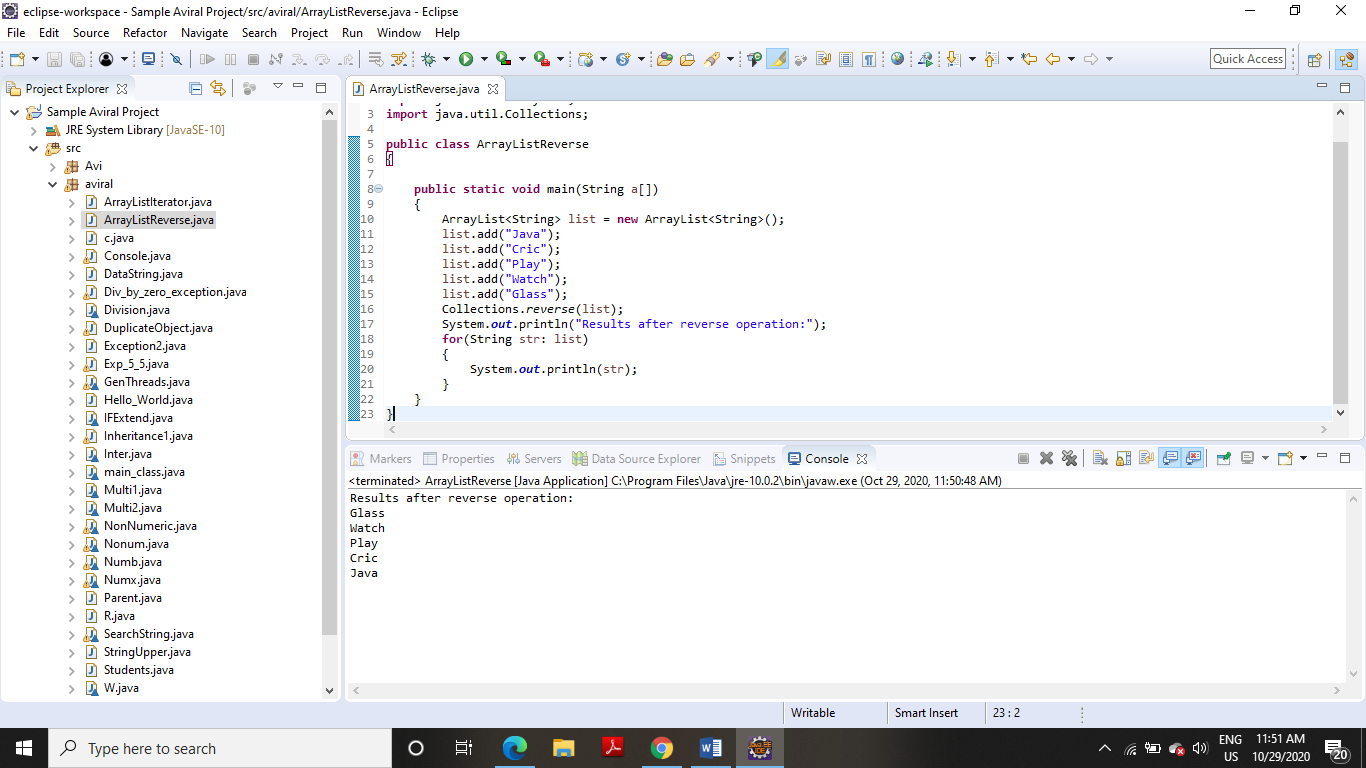
System.***out***.println(str);

}

}

}

**OUTPUT**

****

**2. Write a program for the following HashMap:**

**(i) find whether specified key exists or not**.

**CODE**

**package** rohan;

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

**public** **class** GFG

{

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

{

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

map.put(1, "Hi");

map.put(2, "Hello");

map.put(3, "How are you");

**int** keyToBeChecked = 2;

System.***out***.println("HashMap: "+ map);

Iterator<Map.Entry<Integer, String> >iterator = map.entrySet().iterator();

**boolean** isKeyPresent = **false**;

**while** (iterator.hasNext())

{

Map.Entry<Integer, String>entry=iterator.next();

**if** (keyToBeChecked == entry.getKey())

{

isKeyPresent = **true**;

}

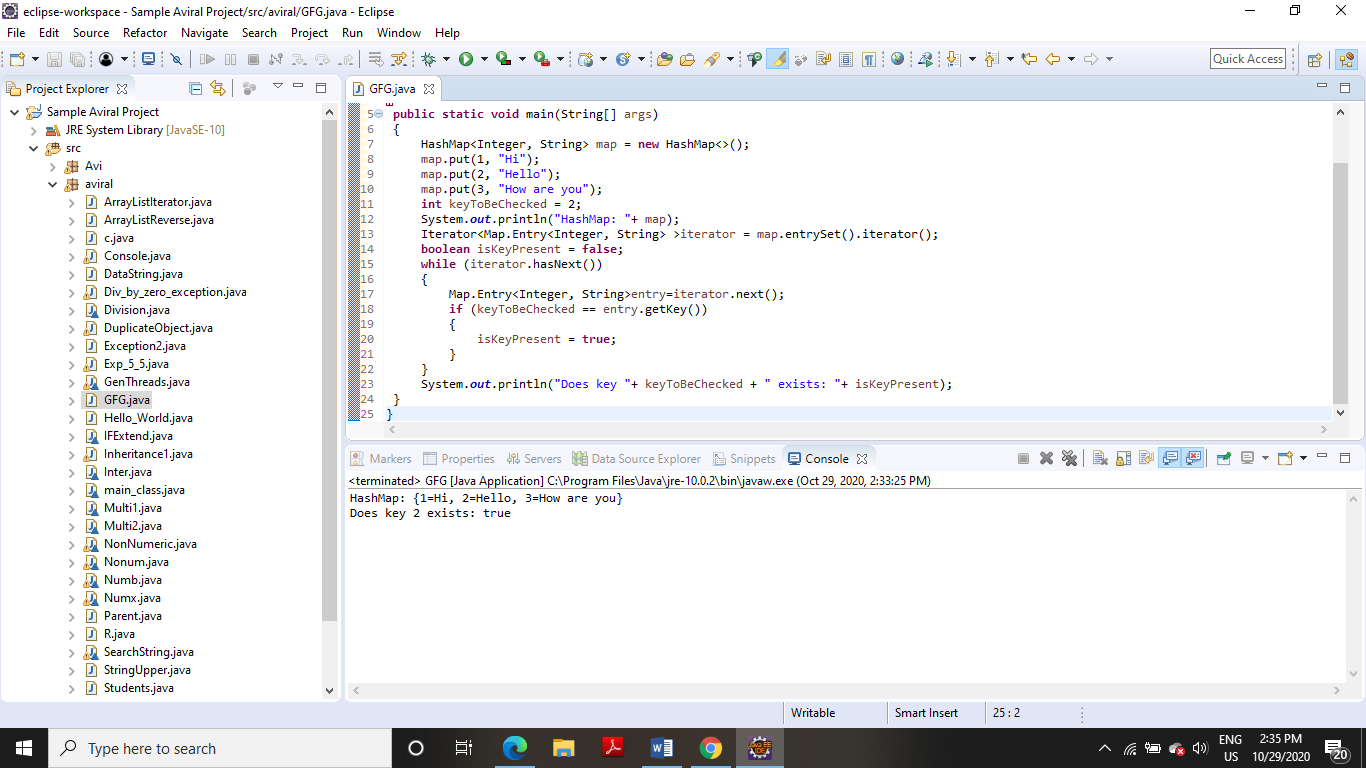
}

System.***out***.println("Does key "+ keyToBeChecked + " exists: "+ isKeyPresent);

}

}

**OUTPUT**

****

**(ii) find whether specified value exists or not**

**CODE**

**package** rohan;

**import** java.util.HashMap;

**public** **class** CheckValue

{

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

{

HashMap<Integer, String> hashmap = **new** HashMap<Integer, String>();

hashmap.put(11,"Aviral");

hashmap.put(22,"Mehra");

hashmap.put(33,"Singh");

hashmap.put(44,"Rajesh");

hashmap.put(55,"Kate");

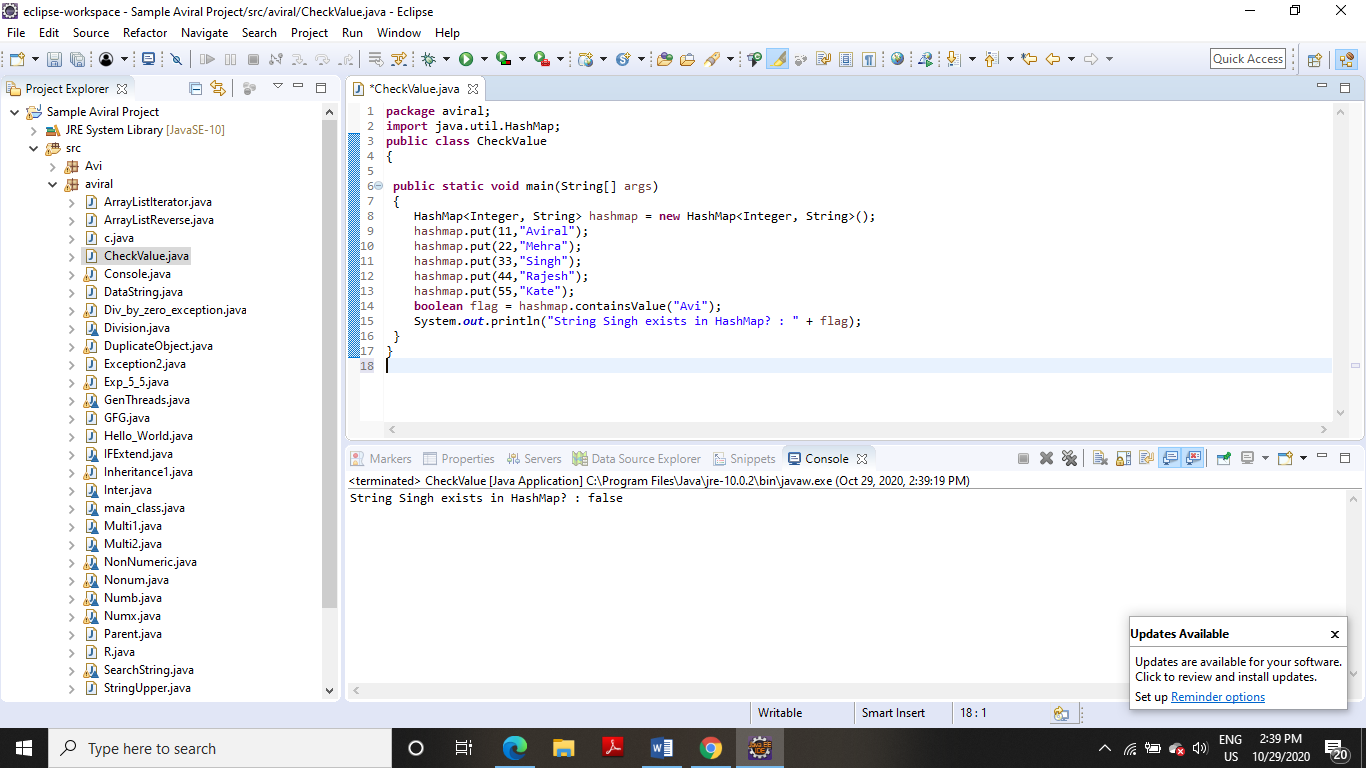
**boolean** flag = hashmap.containsValue("Avi");

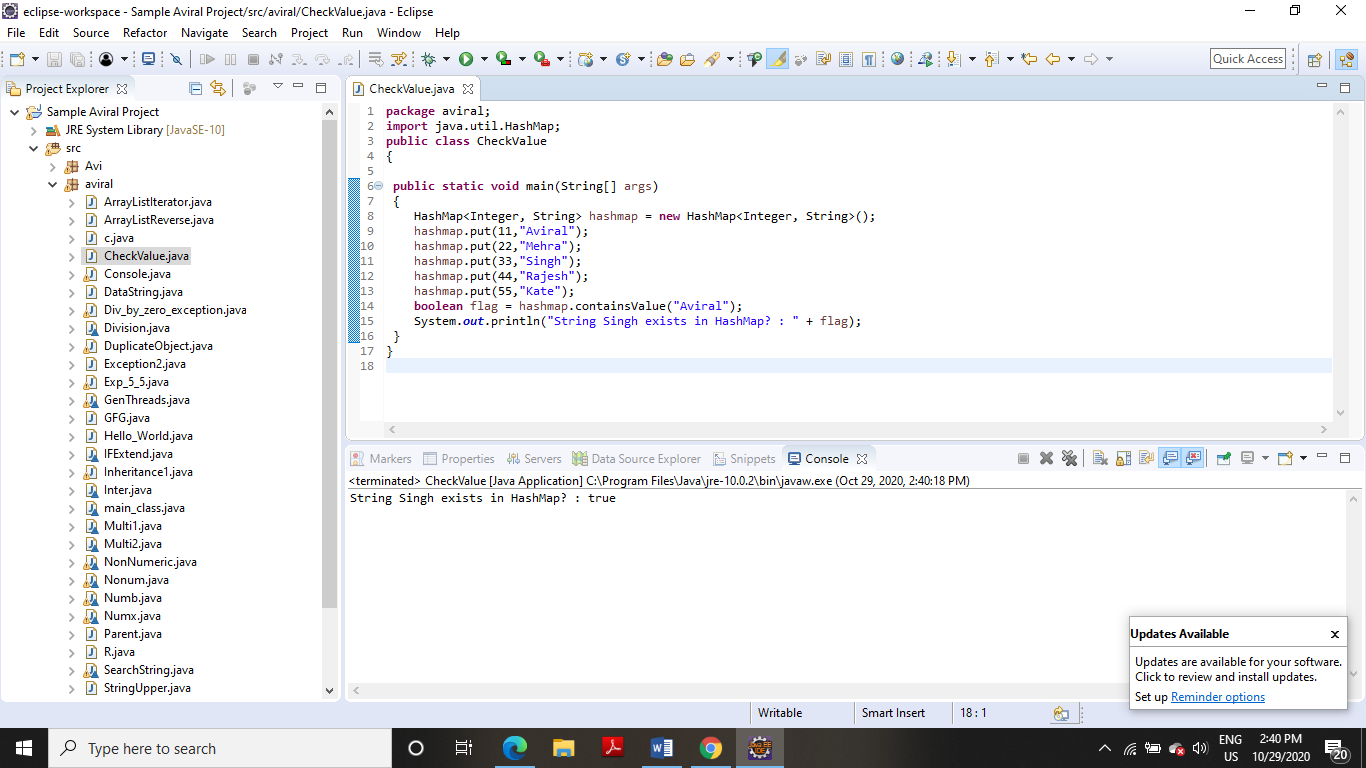
System.***out***.println("String Singh exists in HashMap? : " + flag);

}

}

**OUTPUT**

****

****

**(iii) get all keys from the given HashMap**

**package** rohan;

**import** java.util.HashMap;

**import** java.util.Set;

**public** **class** HashMapKeys

{

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

{

HashMap<String, String> hm = **new** HashMap<String, String>();

hm.put("first", "FIRST INSERTED");

hm.put("second", "SECOND INSERTED");

hm.put("third","THIRD INSERTED");

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

Set<String> keys = hm.keySet();

**for**(String key: keys)

{

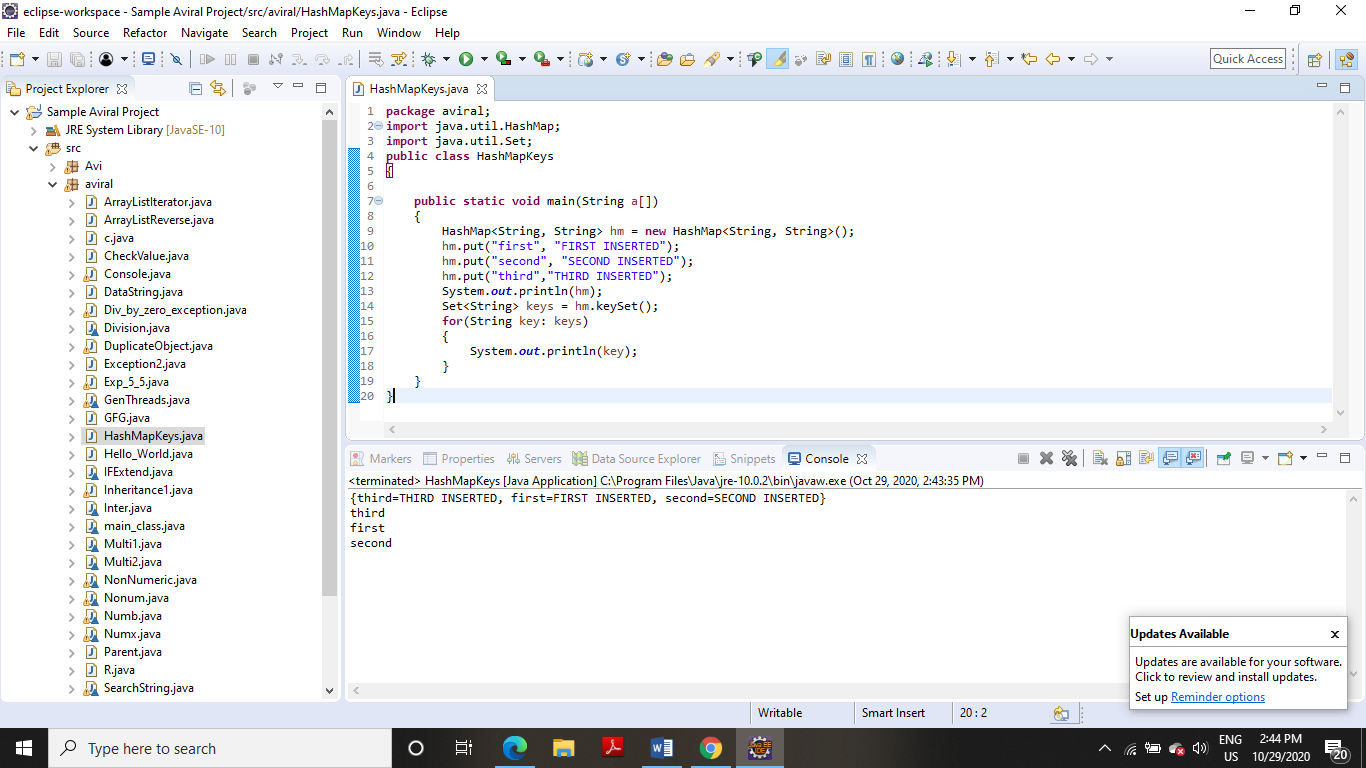
System.***out***.println(key);

}

}

}

**OUTPUT**

****

**(iv) get all key-value pair as Entry objects**

**CODE**

**package** rohan;

**import** java.util.HashMap;

**import** java.util.Map.Entry;

**import** java.util.Set;

**public** **class** HashEntrySet

{

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

{

HashMap<String, String> hm = **new** HashMap<String, String>();

hm.put("first", "FIRST INSERTED");

hm.put("second", "SECOND INSERTED");

hm.put("third","THIRD INSERTED");

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

Set<Entry<String, String>> entires = hm.entrySet();

**for**(Entry<String,String> ent:entires)

{

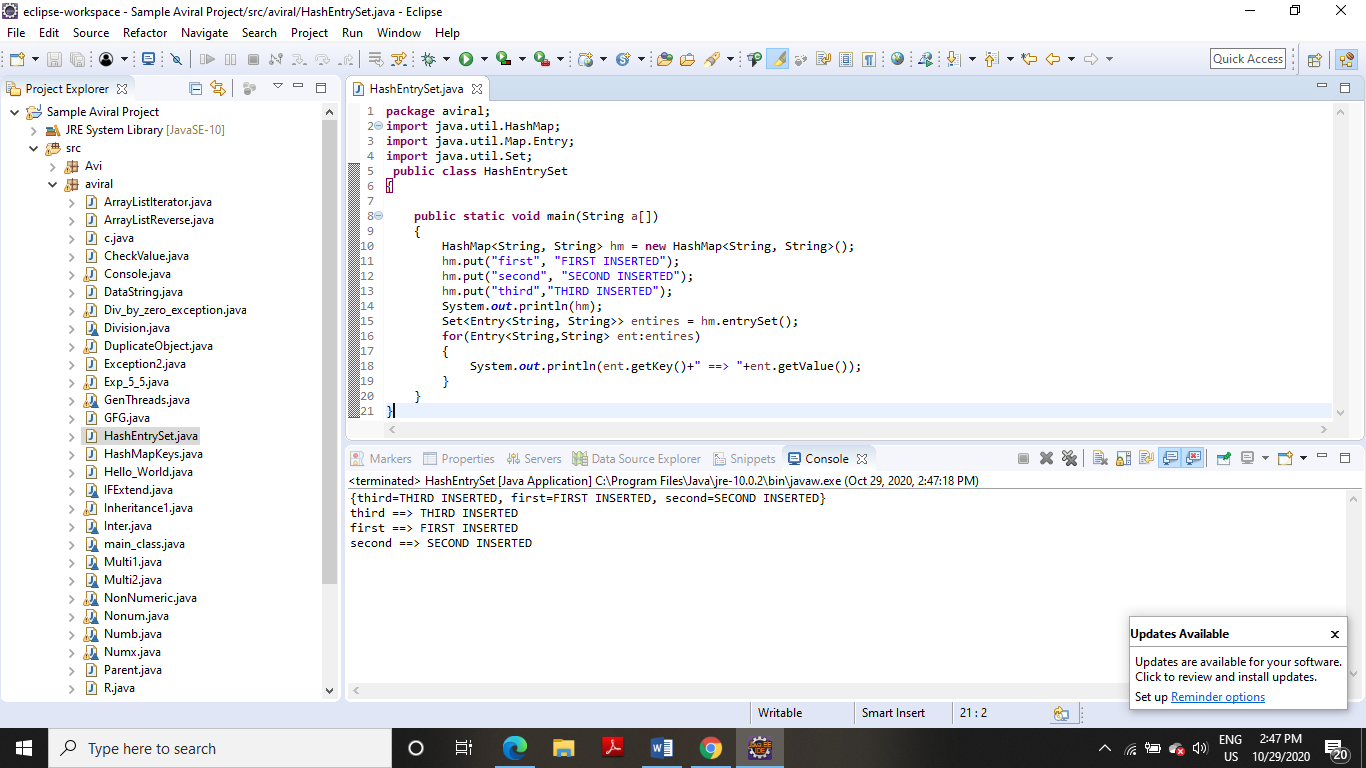
System.***out***.println(ent.getKey()+" ==> "+ent.getValue());

}

}

}

**OUTPUT**

****

**3. Write a program for the following:**

**(i) HashSet copy another collection object to HashSet object.**

**CODE**

**package** rohan;

**import** java.util.HashSet;

**public** **class** HashSetCopy

{

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

{

HashSet<String> hs = **new** HashSet<String>();

hs.add("first");

hs.add("second");

hs.add("third");

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

HashSet<String> subSet = **new** HashSet<String>();

subSet.add("s1");

subSet.add("s2");

hs.addAll(subSet);

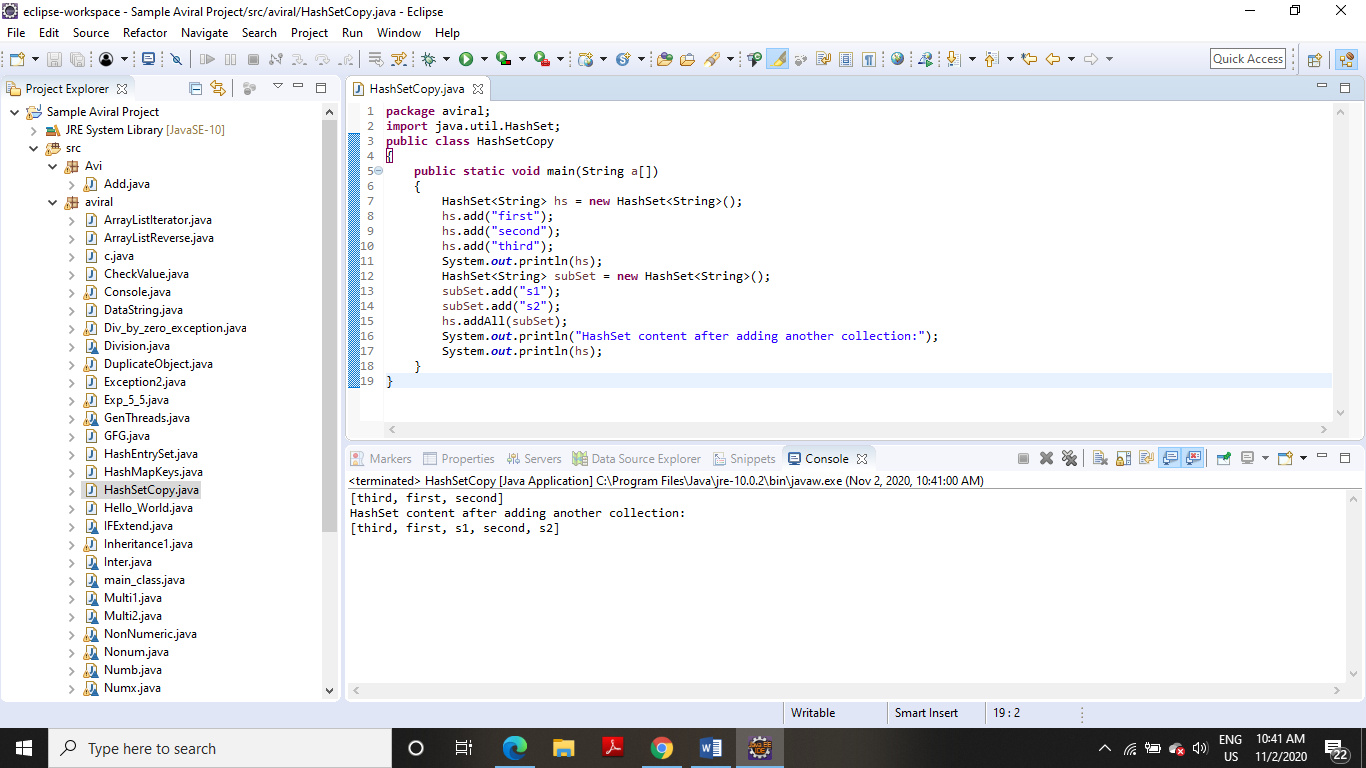
System.***out***.println("HashSet content after adding another collection:");

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

}

}

**OUTPUT**

****

**(ii) delete all entries at one call from HashSet**

**CODE**

**package** rohan;

**import** java.util.HashSet;

**public** **class** HashSetClear

{

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

{

HashSet<String> hs = **new** HashSet<String>();

hs.add("first");

hs.add("second");

hs.add("third");

System.***out***.println("My HashSet content:");

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

System.***out***.println("Clearing HashSet:");

hs.clear();

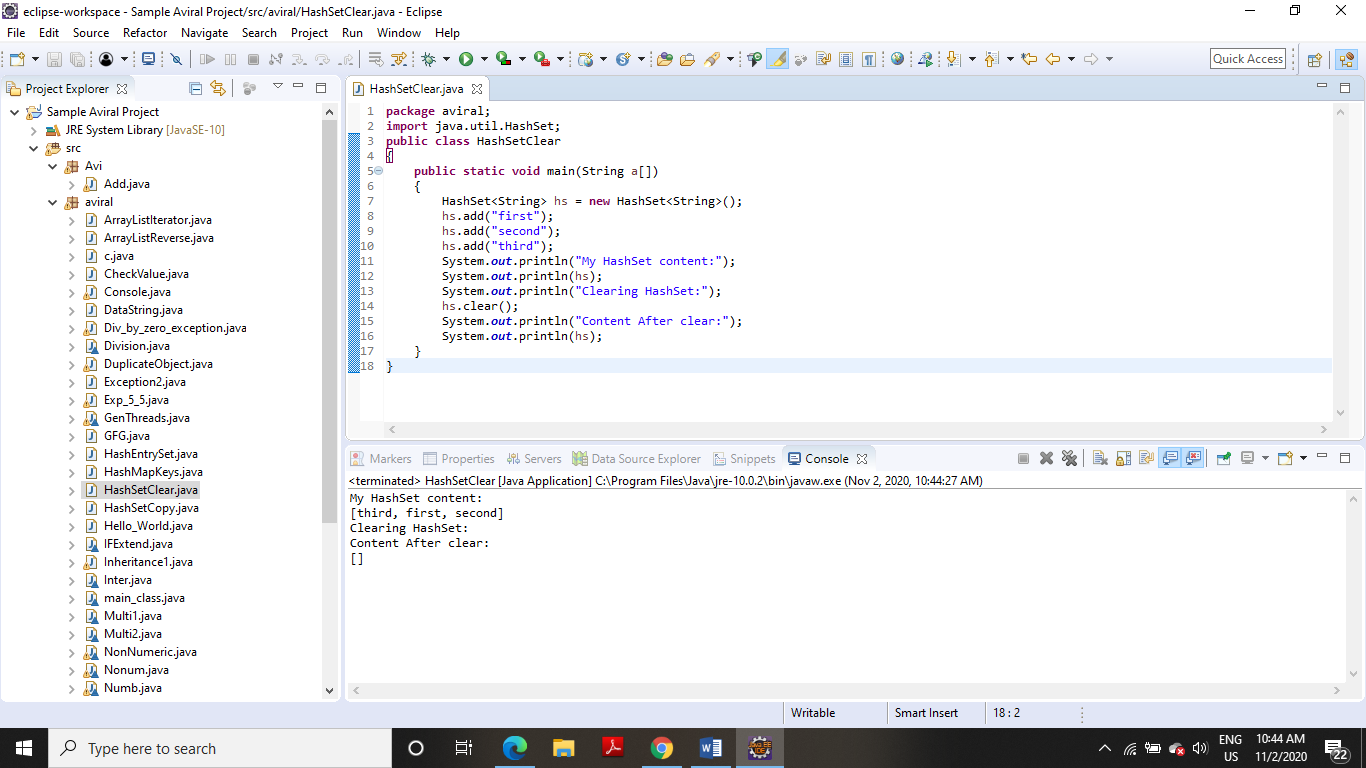
System.***out***.println("Content After clear:");

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

}

}

**OUTPUT**

****

**(iii) search user defined objects from HashSet**

**CODE**

**package** rohan;

**import** java.util.HashSet;

**public** **class** HashSetSearchObject

{

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

{

HashSet<Price> lhs = **new** HashSet<Price>();

lhs.add(**new** Price("Banana", 20));

lhs.add(**new** Price("Apple", 40));

lhs.add(**new** Price("Orange", 30));

**for**(Price pr:lhs)

{

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

}

Price key = **new** Price("Banana", 20);

System.***out***.println("Does set contains key? "+lhs.contains(key));

}

}

**class** Price

{

**private** String item;

**private** **int** price;

**public** Price(String itm, **int** pr)

{

**this**.item = itm;

**this**.price = pr;

}

**public** **int** hashCode()

{

System.***out***.println("In hashcode");

**int** hashcode = 0;

hashcode = price\*20;

hashcode += item.hashCode();

**return** hashcode;

}

**public** **boolean** equals(Object obj)

{

System.***out***.println("In equals");

**if** (obj **instanceof** Price)

{

Price pp = (Price) obj;

**return** (pp.item.equals(**this**.item) && pp.price == **this**.price);

} **else**

{

**return** **false**;

}

}

**public** String getItem()

{

**return** item;

}

**public** **void** setItem(String item)

{

**this**.item = item;

}

**public** **int** getPrice()

{

**return** price;

}

**public** **void** setPrice(**int** price)

{

**this**.price = price;

}

**public** String toString()

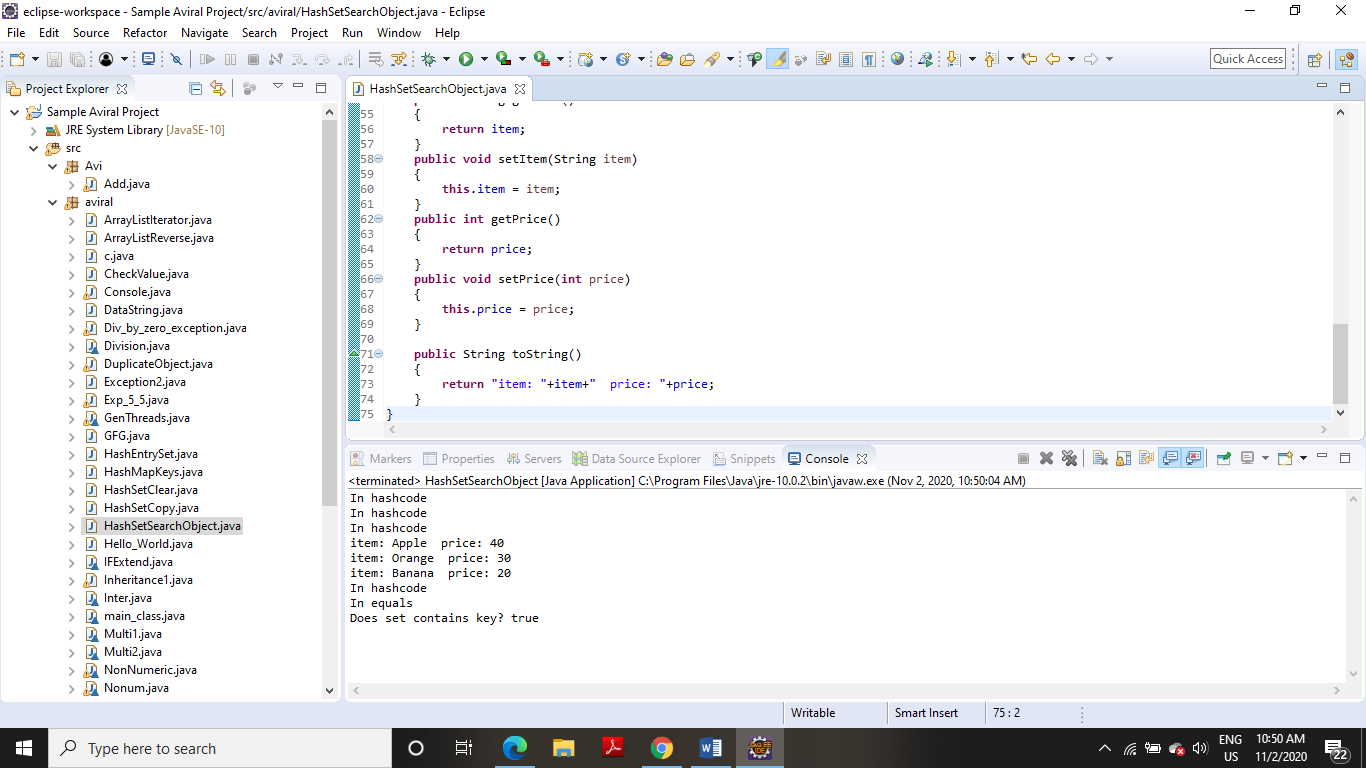
{

**return** "item: "+item+" price: "+price;

}

}

**OUTPUT**

****