**Lab#9**

**Abstract Data Types**

**OBJECTIVE:** Understanding and implementing abstract data types.

**LAB TASK:** Considering a bank interface, it will have different concrete classes for receiving bill payments, opening of new accounts and contacting those loan takers whose limits of extension have been expired. Now, develop this scenario on the described ADT and decide which ADT should be used for each concrete class, also implement the whole scenario in multiple java classes.

**CODE:**

INTERFACE BANKING SYSTEM:

**package** lab9scd;

**public** **interface** Banking\_system {

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

}

BANK CLASS:

**package** lab9scd;

**public** **class** Bank {

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

loan l =**new** loan();

l.print();

bill\_payment p = **new** bill\_payment();

p.print();

new\_account a = **new** new\_account ();

a.print();

}}

BILL PAYMENT CLASS:

**package** lab9scd;

**import** java.util.ArrayList;

**import** java.util.List;

**public** **class** bill\_payment **implements** Banking\_system{

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

List <String> customers =**new** ArrayList<String>();

customers.add("Saraah");

customers.add("khan");

customers.add("aliza");

customers.add("noor");

System.***out***.println("\nCustomers List for payment");

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

System.***out***.println("\nPending Customers");

customers.remove(2);

customers.remove(0);

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

}}

LOAN CLASS:

**package** lab9scd;

**import** java.util.Stack;

**public** **class** loan **implements** Banking\_system {

**static** **void** stack\_push(Stack<Integer> S)

{

**for** (**int** i = 1; i < 5; i++)

{

S.push(i);

}

}

**static** **void** stack\_pop(Stack<Integer> S)

{

System.***out***.println("Customer List According to least limit extension");

**int** j=1;

**for** (**int** i=1; i < 5; i++)

{

Integer y = (Integer) S.pop();

System.***out***.println("Customer "+y+" has Left "+j\*2+" month");

j++;

}

}

**static** **void** stack\_peek (Stack<Integer> S) {

Integer element = (Integer) S.peek();

System.***out***.println("Customer whose extension limit is about to expire: Customer + element");

}

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

Stack<Integer> S= **new** Stack<Integer>();

*stack\_push*(S);

*stack\_pop*(S);

*stack\_push*(S);

*stack\_peek*(S);

}}

**NEW ACCOUNT CLASS:**

**package** lab9scd;

**import** java.util.LinkedList;

**import** java.util.Queue;

**public** **class** new\_account {

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

Queue<String> q = **new** LinkedList<String>();

q.add("nabia");

q.add("sania");

q.add("shazia");

q.add("faiza");

System.***out***.println("\nCustomer Queue for Account Opening");

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

}

}

OUTPUT:

