CIRCULAR LINKED LIST :

**package** circularlinkedlist27;

**public** **class** Linkedlist {

**static** **class** Node

{

**int** data;

Node next;

};

**static** Node addToEmpty(Node last, **int** data)

{

**if** (last != **null**)

**return** last;

Node temp = **new** Node();

temp.data = data;

last = temp;

last.next = last;

**return** last;

}

**static** Node addBegin(Node last, **int** data)

{

**if** (last == **null**)

**return** *addToEmpty*(last, data);

Node temp = **new** Node();

temp.data = data;

temp.next = last.next;

last.next = temp;

**return** last;

}

**static** Node addEnd(Node last, **int** data)

{

**if** (last == **null**)

**return** *addToEmpty*(last, data);

Node temp = **new** Node();

temp.data = data;

temp.next = last.next;

last.next = temp;

last = temp;

**return** last;

}

**static** Node addAfter(Node last, **int** data, **int** item)

{

**if** (last == **null**)

**return** **null**;

Node temp, p;

p = last.next;

**do**

{

**if** (p.data == item)

{

temp = **new** Node();

temp.data = data;

temp.next = p.next;

p.next = temp;

**if** (p == last)

last = temp;

**return** last;

}

p = p.next;

} **while**(p != last.next);

System.***out***.println(item + " not present in the list.");

**return** last;

}

**static** **void** traverse(Node last)

{

Node p;

**if** (last == **null**)

{

System.***out***.println("List is empty.");

**return**;

}

p = last.next;

**do**

{

System.***out***.print(p.data + " ");

p = p.next;

}

**while**(p != last.next);

}

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

{

Node last = **null**;

last = *addToEmpty*(last, 6);

last = *addBegin*(last, 4);

last = *addBegin*(last, 2);

last = *addEnd*(last, 8);

last = *addEnd*(last, 12);

last = *addAfter*(last, 10, 8);

*traverse*(last);

}

}