```
    cout << "Creating an empty queue and empty stack with type double" << endl;
    LinkedQueue<double> lq;
    LinkedStack<double> ls;
    cout << ls << endl;
    cout << lq << endl;
    cout << endl;

    cout << "Filling queue and stack with numbers 0.1-9.1" << endl;
    for (double i = 0.1; i < 10.1; ++i) {
        lq.enqueue(i);
        ls.push(i);
    }
    cout << ls << endl;
    cout << lq << endl;
    cout << endl;

    cout << "checking top of both stack and queue" << endl;
    cout << ls.top() << endl;
    cout << lq.first() << endl;
    cout << endl;

    cout << "popping/dequeueing one number from stack and queue" << endl;
    ls.pop();
    lq.dequeue();
    cout << ls << endl;
    cout << lq << endl;
    cout << endl;

    cout << "Checking if stack/queue is empty" << endl;
    cout << boolalpha << lq.isEmpty() << endl;
    cout << boolalpha << ls.isEmpty() << endl;
    cout << endl;

    cout << "Clearing both queue/stack" << endl;
    ls.~LinkedStack();
    lq.~LinkedQueue();
    cout << ls << endl;
    cout << lq << endl;
    cout << endl;

    cout << "Checking if stack/queue is empty" << endl;
    cout << boolalpha << lq.isEmpty() << endl;
    cout << boolalpha << ls.isEmpty() << endl;
    cout << endl;

    cout << "Creating an empty queue and empty stack with type string" << endl;
    LinkedQueue<string> lq2;
    LinkedStack<string> ls2;
    cout << ls2 << endl;
    cout << lq2 << endl;
    cout << endl;

    cout << "Filling queue and stack with strings 0-9" << endl;
    for (int i = 0; i < 10; ++i) {
        stringstream ss;
        ss << i;
        lq2.enqueue(ss.str());
        ls2.push(ss.str());
    }
    cout << ls2 << endl;
    cout << lq2 << endl;
    cout << endl;

    cout << "checking top of both stack and queue" << endl;
    cout << ls2.top() << endl;
    cout << lq2.first() << endl;
    cout << endl;

    cout << "popping/dequeueing one string from stack and queue" << endl;
    ls2.pop();
    lq2.dequeue();
    cout << ls2 << endl;
    cout << lq2 << endl;
    cout << endl;

    cout << "Checking if stack/queue is empty" << endl;
    cout << boolalpha << lq2.isEmpty() << endl;
    cout << boolalpha << ls2.isEmpty() << endl;
    cout << endl;

    cout << "Clearing both queue/stack" << endl;
    ls2.~LinkedStack();
    lq2.~LinkedQueue();
    cout << ls2 << endl;
    cout << lq2 << endl;
    cout << endl;

    cout << "Checking if stack/queue is empty" << endl;
    cout << boolalpha << lq2.isEmpty() << endl;
    cout << boolalpha << ls2.isEmpty() << endl;
    cout << endl;

    system("pause");
    return 0;
}
```