Concrete Collections

	ArrayList	*Stack*	*Queue*
Create	*ArrayList arList = new ArrayList()* *{* *1,* *null,* *"Bill",* *300,* *" ",* *4.5f,* *300,* *};*	Stack myStack = new Stack();	*Queue callerIds = new Queue();*
Delete	*arList.Remove(null);*	*myStack.Pop()*	*callerIds.Dequeue();*
Update	*arList.Insert(1, "Second Item");*	myStack.Push(2);	*callerIds.Enqueue(2);*
Read	*foreach (var val in arList)* *Console.WriteLine(val);*	*foreach (var item in myStack)* *Console.Write(item + ",");*	*foreach (var id in callerIds)///read q* *Console.Write(id);*

Generic Collections

	List	*Stack*	*Queue*
Create	*public void ListExample()* *{* *// Create a list of strings by using a* *// collection initializer.* *var salmons = new List<string> { "chinook", "coho", "pink", "sockeye" };*	Stack<int> myStack = new Stack<int>();	*Queue<int> callerIds = new Queue<int>();*
Delete	*// Remove an element from the list by specifying* *// the object.* *salmons.Remove("coho");*	*myStack.Pop()*	*callerIds.Dequeue();*
Update	*salmons.Add("Added");///Update*	myStack.Push(2);	*callerIds.Enqueue(2);*
Read	*// Iterate through the list.* *foreach (var salmon in salmons)* *{* *Console.Write(salmon + " ");* *}*	*foreach (var item in myStack)* *Console.Write(item + ",");*	*foreach (var id in callerIds)///read q* *Console.Write(id);*