

Question 1:
Stack Implementatio:

```
function Stack() {  
  this.items = [];  
  
  this.push = function(element) {  
    this.items.push(element);  
  };  
  
  this.pop = function() {  
    if (this.isEmpty()) {  
      return "Stack is empty";  
    }  
    return this.items.pop();  
  };  
  
  this.isEmpty = function() {  
    return this.items.length === 0;  
  };  
}
```

```
// Example usage  
var stack = new Stack();  
console.log(stack.isEmpty()); // Output: true  
stack.push(10);  
stack.push(20);  
stack.push(30);  
console.log(stack.pop()); // Output: 30  
console.log(stack.isEmpty()); // Output: false  
Queue Implementation:
```

Question 2.

```
function Queue() {  
  this.items = [];  
  
  this.enqueue = function(element) {  
    this.items.push(element);  
  };  
  
  this.dequeue = function() {  
    if (this.isEmpty()) {  
      return "Queue is empty";  
    }  
    return this.items.shift();  
  };  
};
```

```
this.isEmpty = function() {  
    return this.items.length === 0;  
};  
}
```

```
// Example usage  
var queue = new Queue();  
console.log(queue.isEmpty()); // Output: true  
queue.enqueue(10);  
queue.enqueue(20);  
queue.enqueue(30);  
console.log(queue.dequeue()); // Output: 10  
console.log(queue.isEmpty()); //
```