Module 634-1 – Lab8
The Adapter and Façade Patterns

Prof. Dr. Michael Schumacher

Haute Ecole Spécialisée de Suisse occidentale
University of Applied Sciences Western Switzerland

Ex1

Stacks and queues are examples of containers with special insertion and removal behaviors and a special access behavior.

Stacks: Insertion and removal in a stack must be carried out in such a way that the last data inserted is the first one to be removed. One can only retrieve and remove a data element from a stack by way of special access point called the "top". Traditionally, the insertion and removal methods for a stack are called `push` and `pop`, respectively. `push` inserts a data element at the top of the stack. `pop` removes and returns the data element at the top of the stack. A stack is used to model systems that exhibit LIFO (Last In First Out) insert/removal behavior.

Queues: Data insertion and removal in a queue must be carried out in such a way that the first one to be inserted is the first one to be removed. One can only retrieve and remove a data element from a queue by way of special access point called the "front". Traditionally, the insertion and removal methods for a queue are called `enqueue` and `dequeue`, respectively. `enqueue` inserts a data element at the "end" of the queue. `dequeue` removes and returns the data element at the front of the queue. A queue is used to model systems that exhibit FIFO (First In First Out) insertion/removal behavior.

Stacks and queues can be easily implemented by adapting with a list. Using the adapter pattern, design and implement those classes. Our starting interfaces are the following:

```java
public interface MyQueue<E> {
	public void enQueue(E element);
	public E deQueue();
}

public interface MyStack<E> {
	public void push(E element);
	public E pop();
}
```

Ex2

Actually, they are two versions of the adapter pattern: object adapter and class adapter. In the course, we presented the object adapter. Explain shortly what a class adapter is. Give a general class diagram for that.