MINHEAP

```
package DAY_7;
import java.util.ArrayList;
import java.util.List;

public class MinHeap {
    private List<Integer> heap;

    public MinHeap() {
        heap = new ArrayList<>();
    }

    public void insert(int value) {
        heap.add(value);
        heapifyUp(heap.size() - 1);
    }

    public int deleteMin() {
        if (heap.size() == 0) {
            throw new IllegalStateException("Heap is empty");
        }
        if (heap.size() == 1) {
            return heap.remove(0);
        }

        int minValue = heap.get(0);
        heap.set(0, heap.remove(heap.size() - 1));
        heapifyDown(0);
        return minValue;
    }

    public int getMin() {
        if (heap.size() == 0) {
            throw new IllegalStateException("Heap is empty");
        }
        return heap.get(0);
    }

    private void heapifyUp(int index) {
        int parentIndex = (index - 1) / 2;
        if (index > 0 && heap.get(index) < heap.get(parentIndex)) {

             swap(index, parentIndex);
            heapifyUp(parentIndex);
        }
    }

    private void heapifyDown(int index) {
        int leftChild = 2 * index + 1;
        int rightChild = 2 * index + 2;
        int smallest = index;

        if (leftChild < heap.size() && heap.get(leftChild) < heap.get(smallest)) {
            smallest = leftChild;
        }
        if (rightChild < heap.size() && heap.get(rightChild) < heap.get(smallest)) {
            smallest = rightChild;
```

```
        }
        if (smallest != index) {
            swap(index, smallest);
            heapifyDown(smallest);
        }
    }

    private void swap(int index1, int index2) {
        int temp = heap.get(index1);
        heap.set(index1, heap.get(index2));
        heap.set(index2, temp);
    }

    public static void main(String[] args) {
        MinHeap minHeap = new MinHeap();
        minHeap.insert(3);
        minHeap.insert(1);
        minHeap.insert(6);
        minHeap.insert(5);
        minHeap.insert(2);
        minHeap.insert(4);
        System.out.println(minHeap.heap);
        System.out.println("Min value: " + minHeap.getMin());
        System.out.println("Removed min value: " + minHeap.deleteMin());
        System.out.println("New min value: " + minHeap.getMin());

    }
}
```