

# Message System

- Thread local
- Message
- Addressee
- Message system
- Examples

## **Tread local**

If an object can be accessed only from one thread it is thread safe.

Object which can be accessed only from one thread is Thread Local object.

## Message

Message is an object which is created by one thread and processed by another one.

It is effectively Thread Local.

```
public abstract class Message {  
    private final Address from;  
    private final Address to;  
    public Message(Address from, Address to) {  
        this.from = from;  
        this.to = to;  
    }  
    public Address getFrom() {  
        return from;  
    }  
    public Address getTo() {  
        return to;  
    }  
    public abstract void exec(Addressee addressee);  
}
```

## Addressee

Addressee is an object which can process messages.

It can be subclass of Thread or Runnable.

It process messages in a loop every N milliseconds.

```
public interface Addressee {  
    Address getAddress();  
}
```

```
public class DBService extends Thread implements Addressee {  
    ...  
    public void run() {  
        while (true) {  
            MessageSystem.instance().exec(this);  
            try {  
                Thread.sleep(MessageSystem.STEP_TIME);  
            } catch (InterruptedException e) {  
                e.printStackTrace();  
            }  
        }  
    }  
}
```

## MessageSystem

Message system is an object which contains a queue for messages for each Addressee.

Can add messages to a queue.

And execute scheduled messages.

```
public final class MessageSystem {
    private final Map<Address, ConcurrentLinkedListQueue<Message>> messages = new HashMap<>();

    public void addAddressee(Addressee addressee) {
        messages.put(addressee.getAddress(), new ConcurrentLinkedListQueue<>());
    }

    public void sendMessage(Message message) {
        messages.get(message.getTo()).add(message);
    }

    public void exec(Addressee addressee) {
        ConcurrentLinkedListQueue<Message> queue = messages.get(addressee.getAddress());
        while (!queue.isEmpty()) {
            Message message = queue.poll();
            message.exec(addressee);
        }
    }
}
```