

Example of Augmented Input

```
@Override public void run()
{
    ConcurrentHashMap<URL,ServiceListener> listeners =
        serviceListeners.get(service);
    if(listeners!=null)
    {
        synchronized(listeners){
            for(Map.Entry<URL,ServiceListener> entry:
                listeners.entrySet()){
                ServiceListener serviceListener=entry.getValue();
                serviceListener.notifyService(entry.getKey(),
                    getUrl(), urls);
            }
        }
    }
    else{
        Log Statement to be generated
        LoggerUtil.debug("need not notify service:" + service);
    }
}
```

<log_message> "need not notify service:" </log_message>
<similarity> 0.79 </similarity>
<log_message> "Removing expired resumption token {}" </log_message>
<similarity> 0.55 </similarity>
<log_message> "State store {} ... at changelog {}" </log_message>
<similarity> 0.52 </similarity>

```
@Override public void run(){
    ConcurrentHashMap<URL,NotifyListener> listeners =
        notifyListeners.get(service);
    if(listeners!=null)
    {
        synchronized(listeners){
            for(Map.Entry<URL,NotifyListener> entry:
                listeners.entrySet()){
                NotifyListener listener=entry.getValue();
                listener.notify(getUrl(), urls);
            }
        }
    }
    else{
        logger.debug("need not notify service:" + service);
    }
}
```

M
e
t
h
o
d
1

Similarity: 0.79

```
@Override public void run(){
    for(Map.Entry<String,MCR0AISearcher> entry:
        resultMap.entrySet()){
        String searchId=entry.getKey();
        MCR0AISearcher searcher=entry.getValue();
        if((searcher != null) && searcher.isExpired()){
            LOGGER.info("Removing expired resumption token {}",
                searchId );
            resultMap.remove(searchId);
        }
    }
}
```

M
e
t
h
o
d
2

Similarity: 0.55

```
@Override public void updateChangelogOffsets(
    final Map<TopicPartition,Long> writtenOffsets)
{
    for(final Map.Entry<TopicPartition,Long> entry:
        writtenOffsets.entrySet())
    {
        final StateStoreMetadata store=findStore(
            entry.getKey());
        if(store!=null){
            store.setOffset(entry.getValue());
            log.debug("State store {} updated to written
                offset {} at changelog {}",
                store.stateStore.name(),
                store.offset,
                store.changelogPartition);
        }
    }
}
```

M
e
t
h
o
d
3

Similarity: 0.52