Checklist review today:

* final finally
  + variable - constant, must be initialized
  + method - can’t be overridden
  + class - can’t be extended
  + container (like list) – can modify the content, can’t address to different reference (new container).
* Immutable class
  + final class
  + private final fields
  + no setter
  + return deep copy of the collections for getter
* static (method area)
  + block
  + variable
  + methods
  + classes
* Volatile (fields)
  + Visibility
    - guarantee
    - changes made by one thread, data are visible to other threads
  + Atomicity
    - NOT guarantee
    - + CAS (Compare and Swap ->thread safety)
      * Lock free
      * AtomicReference
        + May occur ABA problem
        + AtomicStampedReference can solve
        + compare not only content but also version # can solve.
      * unsafe
      * while loop (for compare)
  + Instruction reordering
    - guarantee
    - may increase parallel execution (by JVM)

New things learned today:

Microservices (advantages\*\*\*\*\*\*\* & disadvantages \*\*\*\*\*)

* divide a big project to small services
* # of services work together to achieve a purpose
* Service Discovery (Eureka)
* @EnableEurekaServer
* Client-side LB(load balancer) (Ribbon)
* Fault Tolerance
  + resilience4j
    - circuitbreaker
    - bulkhead
    - ratelimiter
    - retry
* Api Gateway
  + gate for verify JWT
  + AWS
* Tracing / Security
  + sleuth
    - client
    - track id (same in one request)
    - span id (different)
  + zipkin
    - server
    - docker

flow

* get JWT by OAuth2
* verify JET by gateway
* sent JWT to services depends on different scenarios

deploy

* Java -> .class -> .jar(s) -JDK-> Docker image(s) to Repo

-> Services

Learned Plan tomorrow:

* + Review OOP
    - Abstract
    - Encapsulation
    - Inheritance
    - Polymorphism
  + Review Exception
    - checked exception
    - unchecked exception