summarize missing points:

template -> access elements from template

immutable class:

1. final -> class
2. private final fields
3. getter only, no setter
4. in getter, given referenced data type field, always return deep copy dummies

double vs Double -> primitive vs referenced data type -> 8 primitive data type byte < short < int < long，float < double，boolean，char （1 byte < 2 byte < 4 byte < 8 byte，4 byte < 8 byte，1 bit{1 byte}，2 byte）

auto boxing (primitive type to its wrapper class)

auto unboxing

Throwable: Exception vs Error

========================================================

maxHeap : comparable / comparator (either one)

ArrayDeque -> Circle array

LinkedHastSet

- TreeSet<Employee> ? Comparator(sorting age) vs Comparable(sorting ID),  
=> only use Comparator

override: hashcode + equals -> Employee extends Object -> e1.equals(e2) -> euqlas check each and every fields

Polymorphism

Abstraction -> interface + abstract class

Liskov

D -> Dependency Inversion (modules depend on abstractions , not high on low)