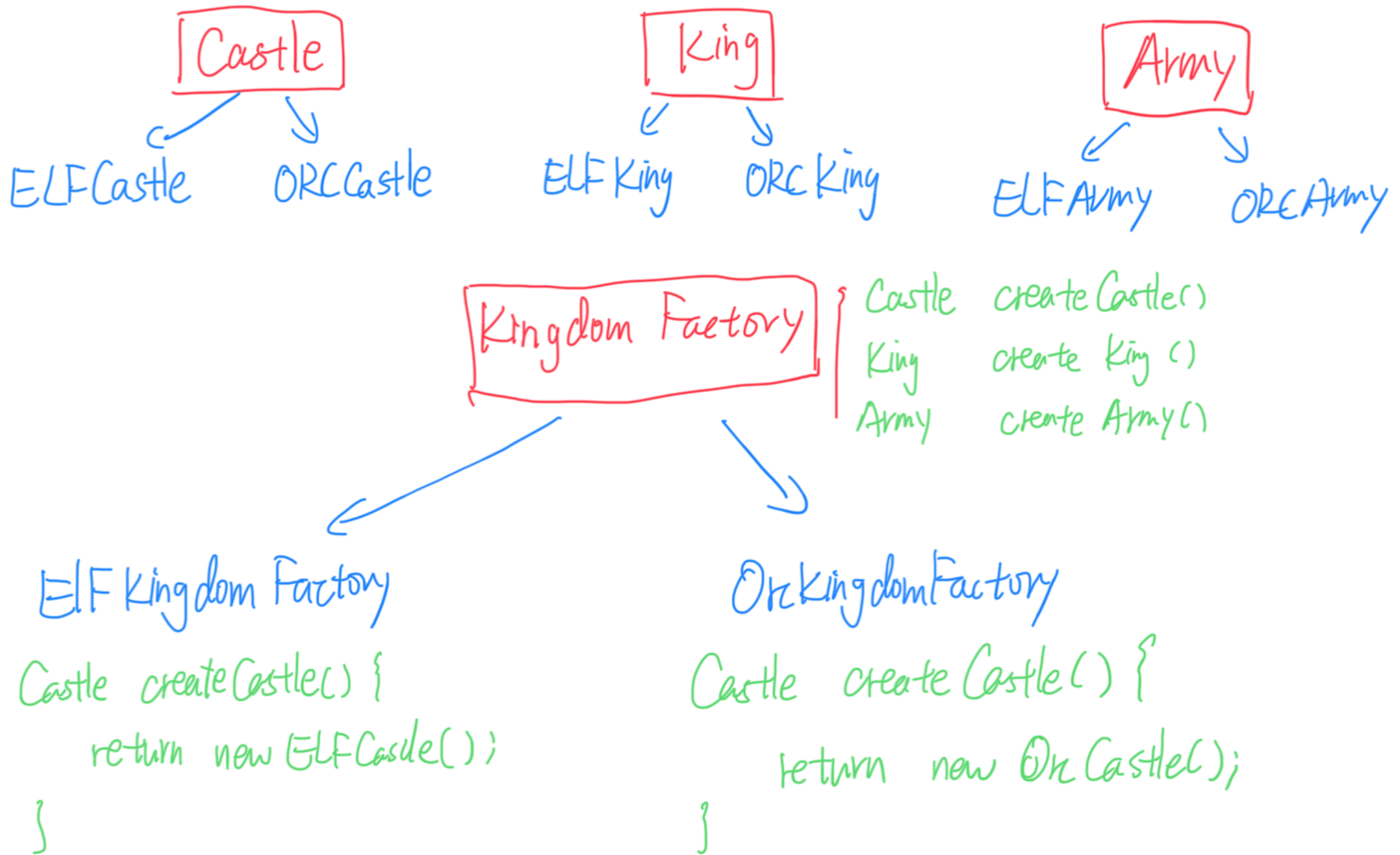
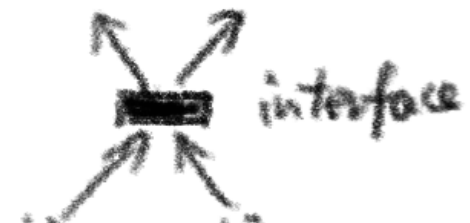


Abstract Factory : Factory Maker



Note: { return type → interface



| return value → object

obj1 obj2

Usage :

① ^{↗ interface} KingdomFactory elfFactory = new ^{↗ object} ElfKingdomFactory();
King elfKing = elfFactory.createKing();

② public ^① static class FactoryMaker {

public enum KingdomType {

ELF, ORC

}

public ^② static ^{③ interface} KingdomFactory makeFactory(KingdomType type){

switch (type) {

④ object.

case ELF: return new ELFKingdomFactory();

case ORC: return new ORCKingdomFactory();

default: throw new Exception();

}

}

}