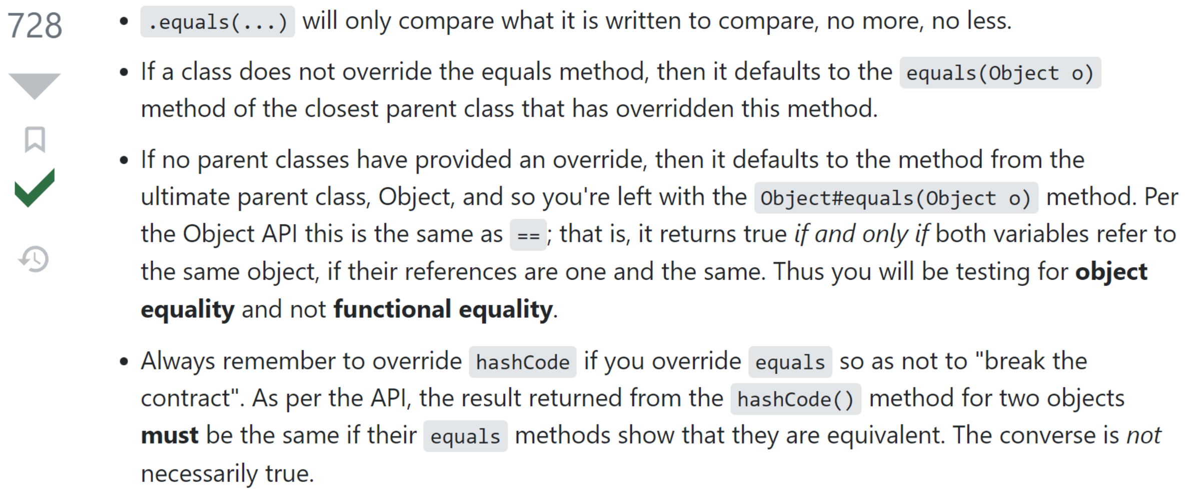
Topics:

Equal() vs ==
Link1 = https://www.scaler.com/topics/difference-between-equals-method-in-java/



- `.equals(...)` will only compare what it is written to compare, no more, no less.
- If a class does not override the equals method, then it defaults to the `equals(Object o)` method of the closest parent class that has overridden this method.
- If no parent classes have provided an override, then it defaults to the method from the ultimate parent class, Object, and so you're left with the `Object#equals(Object o)` method. Per the Object API this is the same as `==`; that is, it returns true *if and only if* both variables refer to the same object, if their references are one and the same. Thus you will be testing for **object equality** and not **functional equality**.
- Always remember to override `hashCode` if you override `equals` so as not to "break the contract". As per the API, the result returned from the `hashCode()` method for two objects **must** be the same if their `equals` methods show that they are equivalent. The converse is *not* necessarily true.