Saturday, May 7, 2022 9:57 AM

Serializable ----------

Pillars of OOP ----------
Major

1. Abstraction ----selecting relevant properties
2. Encapsulation --- access specifiers, getters setters
3. Inheritance ---- IS A relation between classes and Interfaces
4. Polymorphism ----- static and dynamic polymorphism

Minor

1. Strong type --- data type of a reference or a variable is fixed , type casting--- Upcasting/Downcasting
2. Concurrency --- Multithreading, data sharing, synchronized
3. Persistence -------- preserving the state of the object

Where is the object created ? On the Heap in the RAM

RAM	HDD
objects are created	
volatile memory	Non volatile memory
Therefore objects are destroyed after program ends or shutdown	If we want objects to be preserved ||
	Object persistence ----- then the objects must be transferred to HDD --- **file**

When we say we want to persist the object --- we want to save the STATE of the object!!!

What is the STATE of the object ???
current value of properties is called as current state of the object !!

Object persistence is a complex activity ------- there could be many states and those states may belong many objects
When we persist the object we store class information along with the property value, relationships between classes !!!!!!

This complex task of object persistence is called as SERIALIZATION
it includes storing the state of the entire object along with class info

---

The logic of serialization and deserialization -------is internally handled by the JVM!!!!

transient keyword --- it can be written as a qualifier to the property/attribute/data members of a Serializable class
it means that the Serialization **should not save the current value** of the transient property !!!

So during deserialization the default value of that **property-type** is taken!!

HW --- try out

IS A Serialization

Sub class is serializable super is not serializable what happens

HAS A serialization

Class is serializable but the HAS A class is not serializable what happens

---

public static final long serialVersionUID=9l;

This is the ID given for a class

When a class is DESERIALIZED the Deserializer checks if the serialVersionUID of saved class and current class is SAME. If same it will deserialize else NOT deserialize !!!!

If we don't give the serialVersionUID then the serializer autogenerates it for every changed class !!!

File NIO = New IO

java.nio package ----------

Files class

Paths

Path

Dates in Java ----------

java.util.Date

to DB dates you will use java.sql.Date

Date related activities

java.util.Calendar

java.util.GregorianCalendar

SimpleDateFormat

Ex1 - -- print todays date ---- system date

Use java.util.Date

OR

GregorianCalendar

---

how many days are between 30 june2022 and today ?

What is today

current month

EPOCH -----

milliseconds are calculated from 1-1-1970 12 am to todays date

How many millis one sec ?

1000 millis 1 sec
60 sec 1min
60min 1 hour
24 hrs 1 day
