(Part 3)

transient → keyword very important in serialisation.

- It is an access modifier.
- only for variables.

Serialisation is saving state of an Object in a file. File is present in hard disk. It is permanent storage

In serialisation we are storing data permanently for using in future purpose.

But this might contain sensitive information and is exposed

imagine

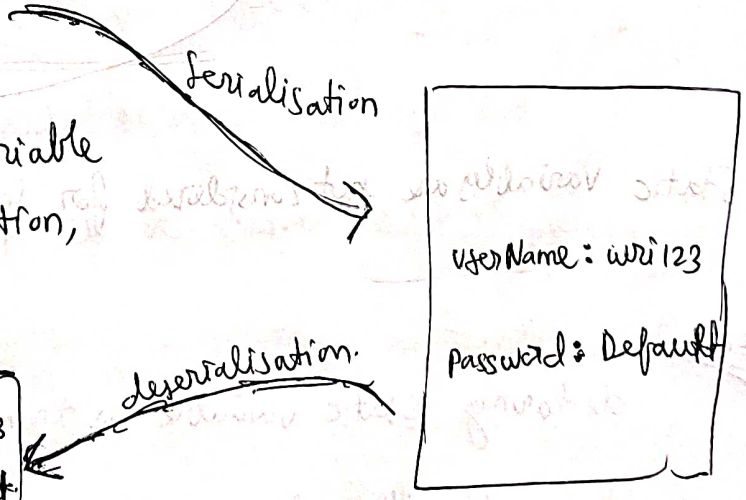
```
class Account{
    userName : uri123
    password : #134KK
```

transient

3 → If we don't want to save the value of a variable during Serialisation,

we will

→ JVM will during serialisation check which variables are transient and use default value for them.



transient means → [Not to Serialize] (For Security Purposes)

[Some Loop Holes for transient]

[static v/s transient] → [static is not part of Object. Hence will not participate in Serialisation]

```
class Dog implements Serializable
{
    int i=10;
    static int j=20;
}
```

(not participating in serialisation)

class level Variable (j=20) hence won't participate in Serialisation.

Object level Variable

```
Dog d1=new Dog();
```

d1 → (i=10)

psuedo code → Serialize d1;

```
Dog d2=dSerialize();
```

Fetched from class level

File (i=10)

```
Sout(d2.i+"---"+d2.j)
```

d2 (i=10)

Static Variables are not considered for Serialisation

(~~Imp~~)

declaring static variable as transient has no use.

It will return the same result

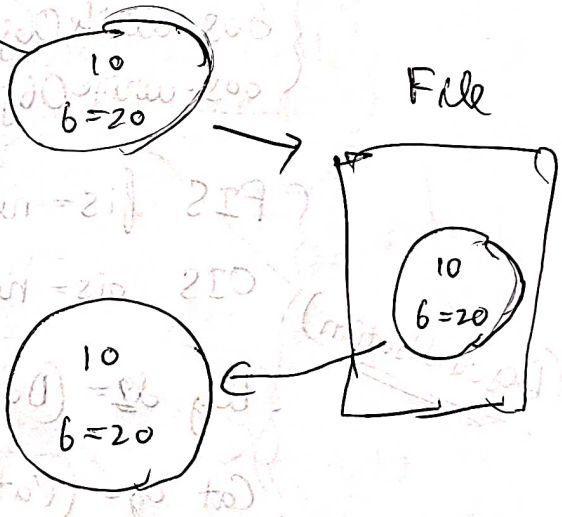
[final v/s transient] → (final variable never in variable format during runtime. It is in value format. It will participate in serialisation)

```
final int a=10;
int b= 20;
```

```
sout(a);
sout(b);
```
} (this code JVM changes) (since value of final does not change)
```
sout(10);
sout(b);
```

```
Class Dog implements Serializeable{
    final int a=10;
    int b= 20;
}
```

(Pass) final variable will participate in serialization by value.




```
Dog d1= new Dog();
serialize ------
Dog d2= deserialize ------
sout(d2.a+"..."+d2.b)
```

If we make final variable as transient., still it will be passed as value.

```
transient final int a=10;
```

So no use of transient on (static & final variables)