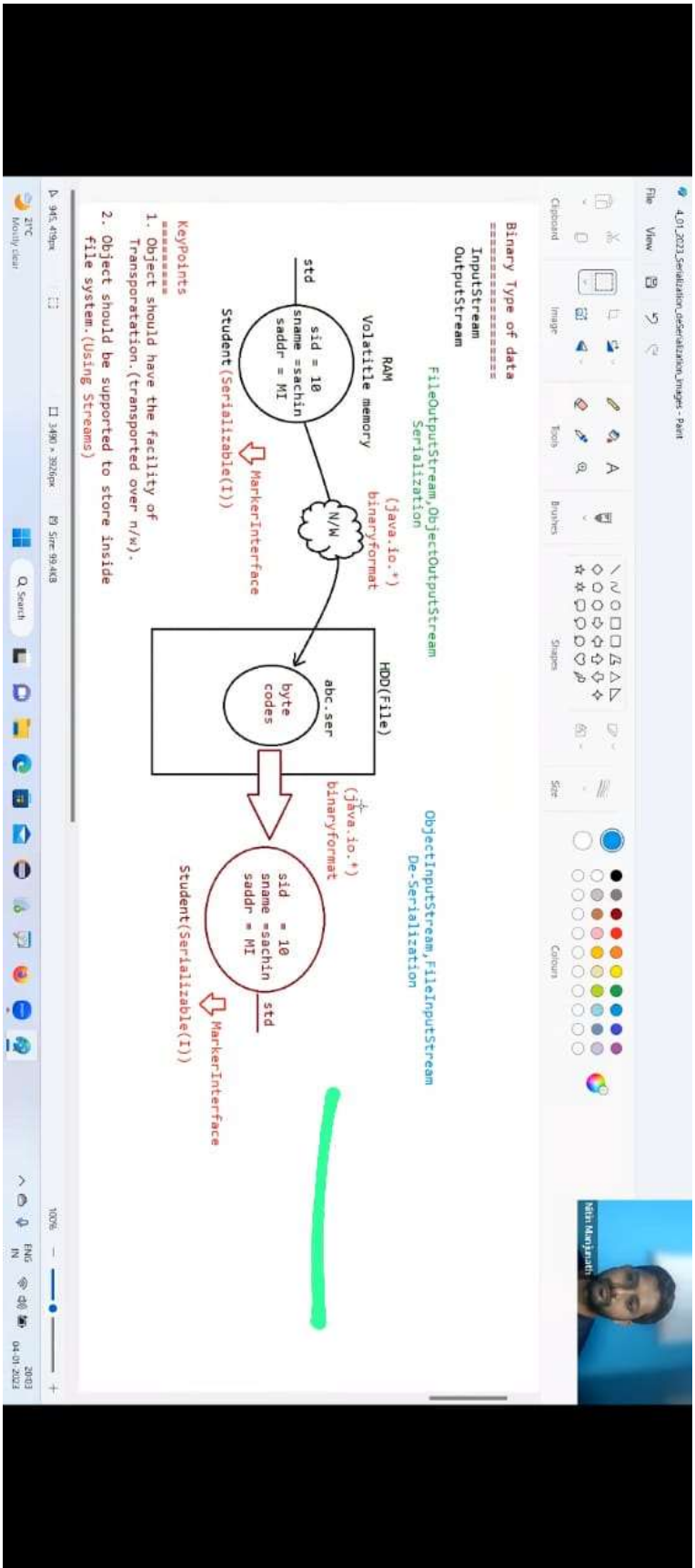
45Java_Serialization_Deserialization



*04_01_2023_Serialization_DeSerialization_notes - Notepad

File Edit View

10. Difference between Serialization & Externalization
11. SerialVersionUID

Serialization: (1.1 v)

=> The process of saving (or) writing state of an object to a file is called serialization but strictly speaking it is the process of converting an
object from java supported form to either network supported form (or) file supported form.

=> By using FileOutputStream and ObjectOutputStream classes we can achieve serialization process.
|=> writeObject(Object obj)

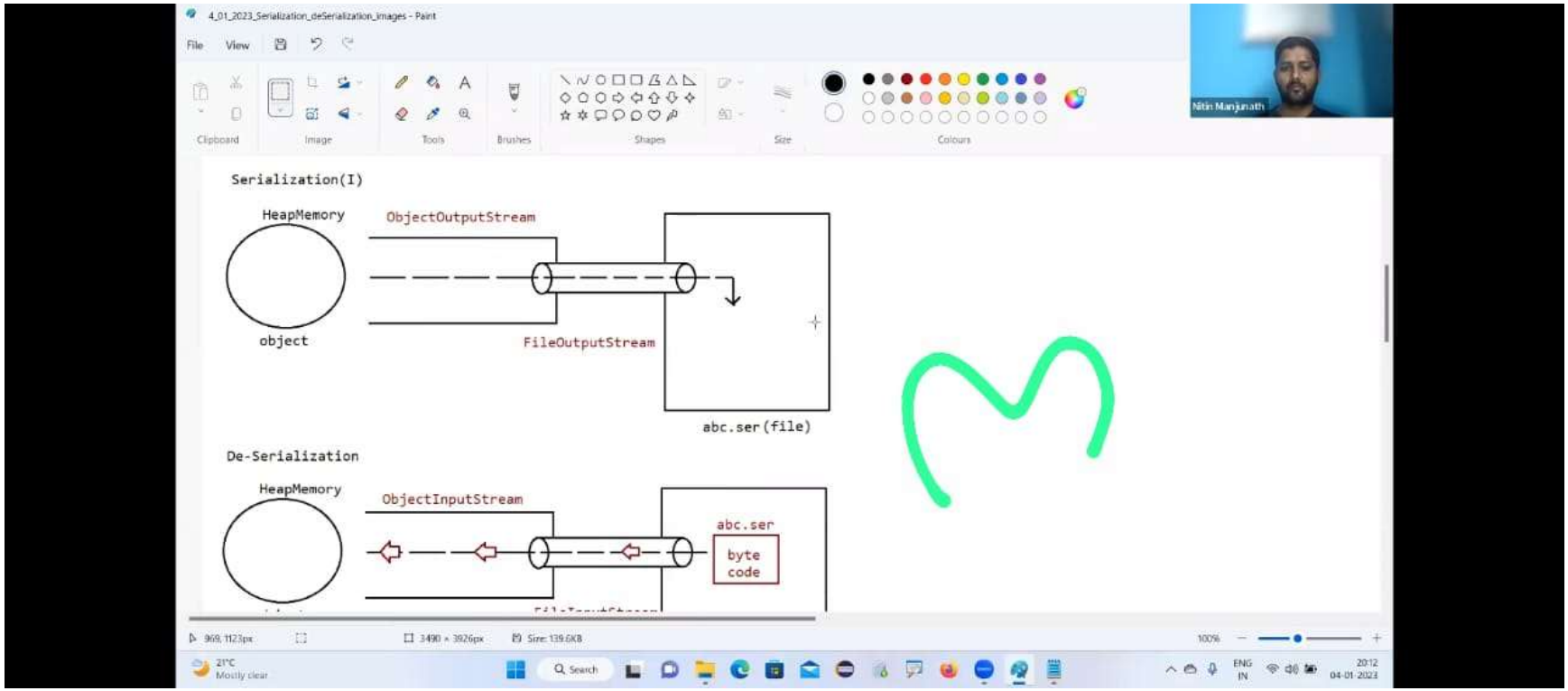
De-Serialization:

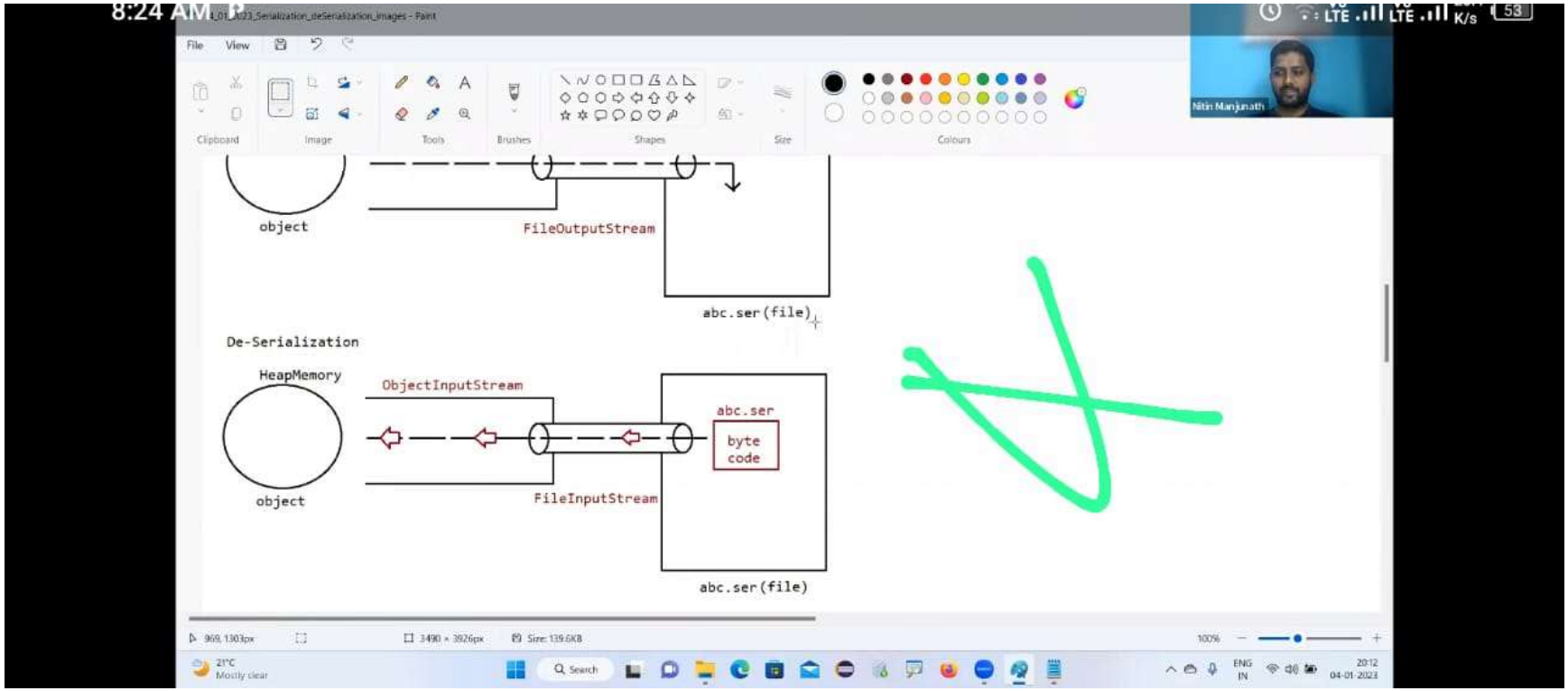
=> The process of reading state of an object from a file is called DeSerialization but strictly speaking it is the process of converti
object from file supported form (or) network supported form to java supported form.

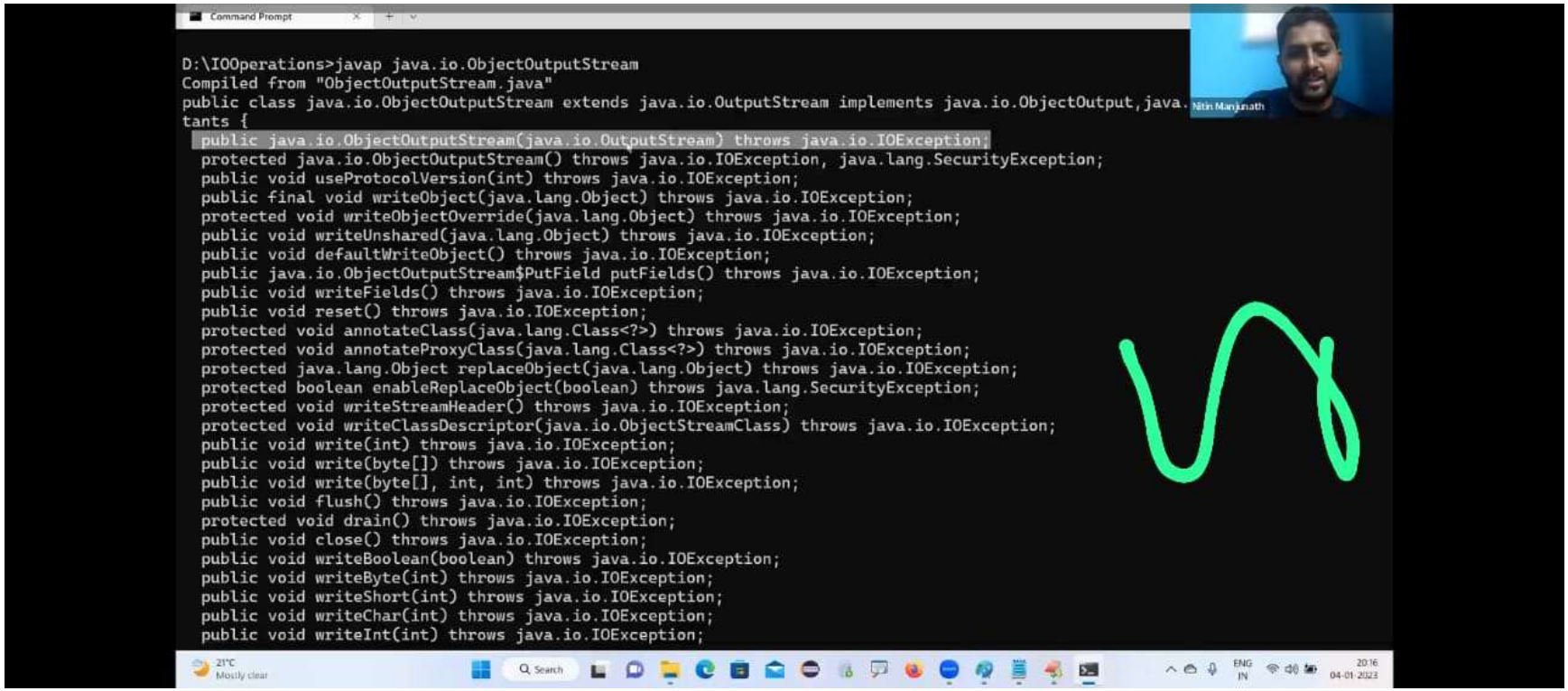
=> By using FileInputStream and ObjectInputStream classes we can achieve DeSerialization.
|=> readObject()

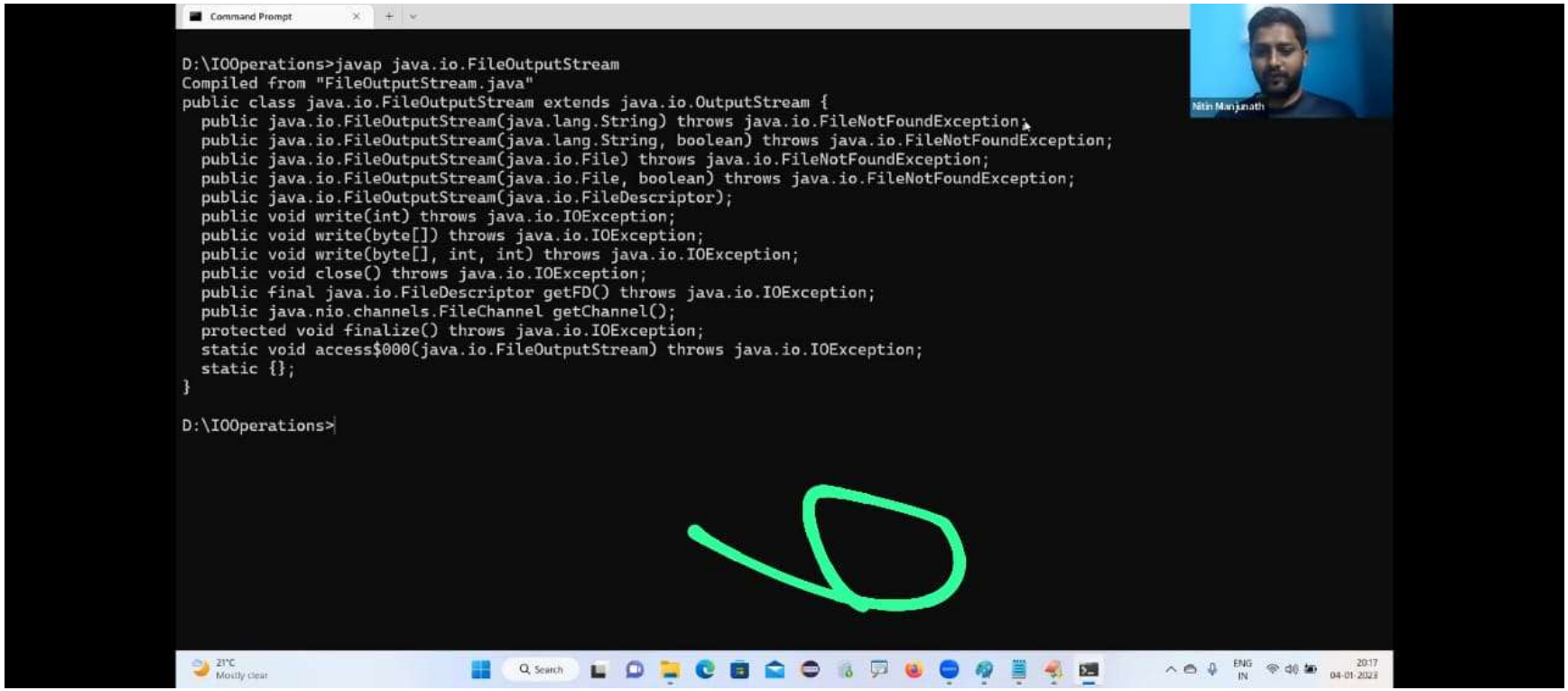
Ln 16, Col 7 110% Windows (CRLF) UTF-8

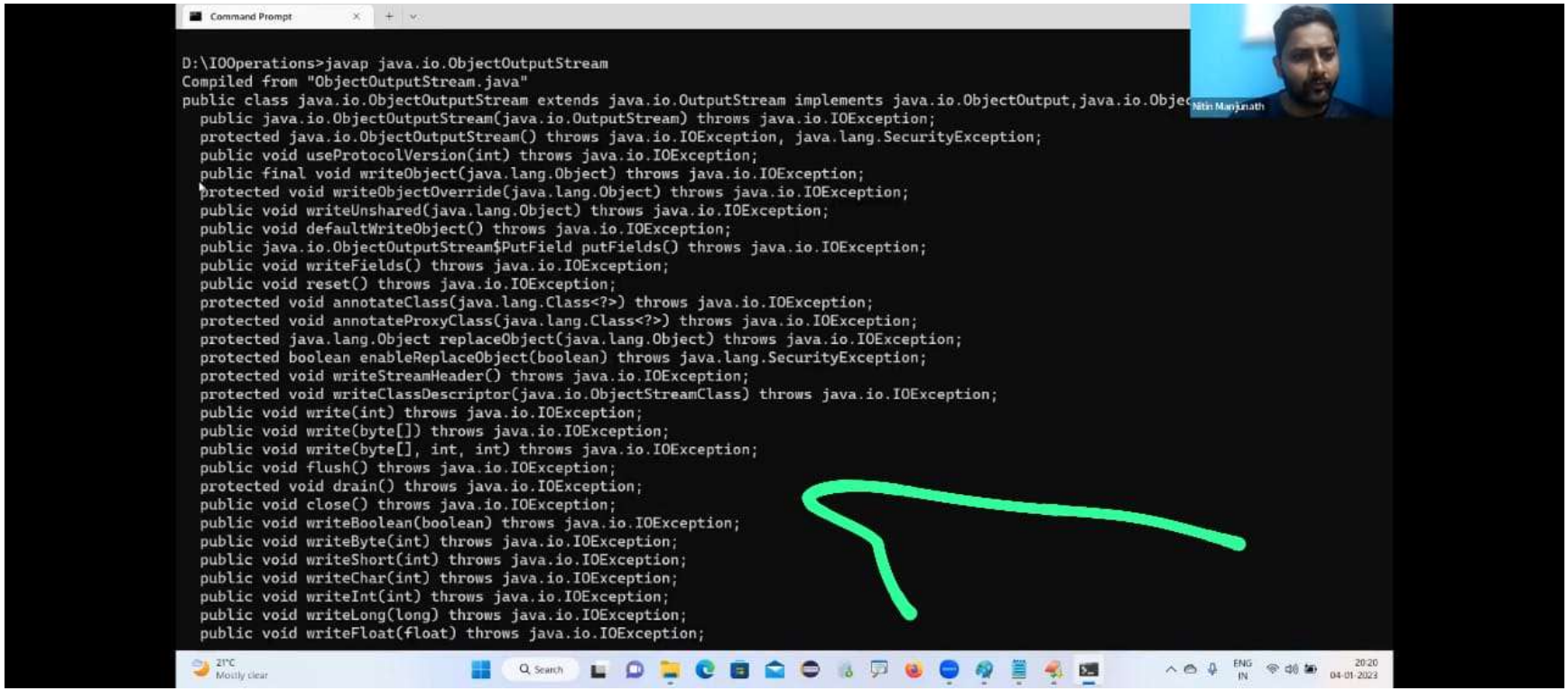
21°C Mostly clear Search ENG IN 20:05 04-01-2023

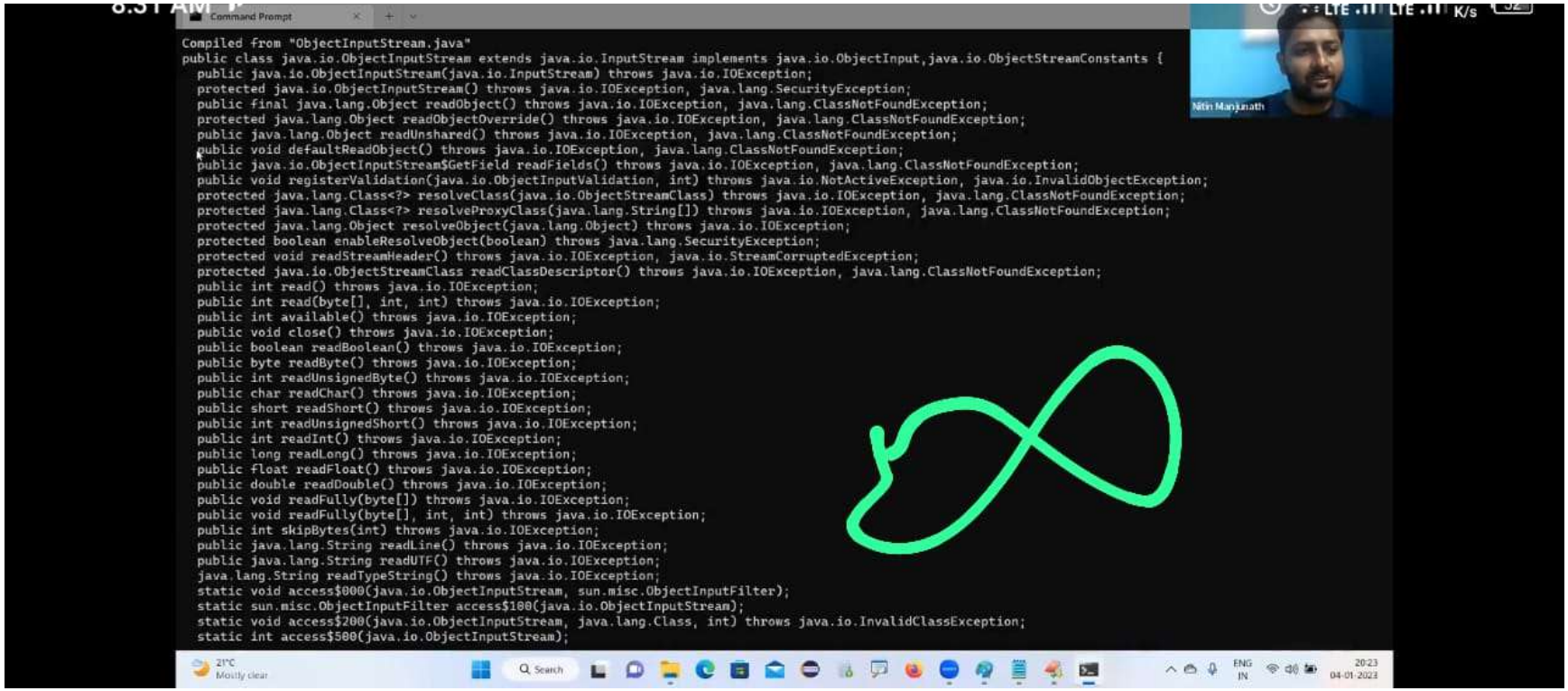

4_01_2023_Serialization_deSerialization_Images - Paint
File
View
Nitin Manjunath
Serialization(I)
HeapMemory
ObjectOutputStream
object
FileOutputStream
abc.ser(file)
De-Serialization
HeapMemory
ObjectInputStream
abc.ser
byte
code

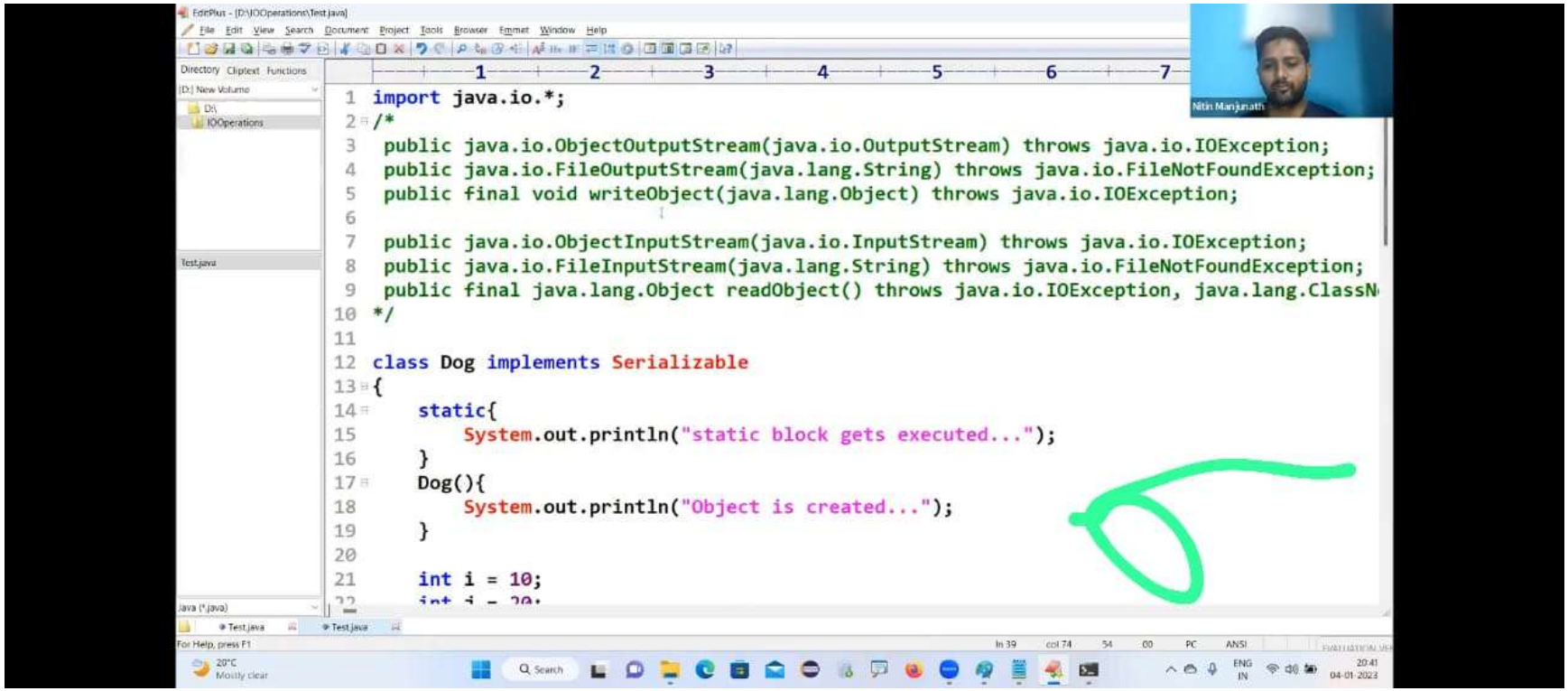

8:24 AM
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
object
FileOutputStream
abc.ser(file)
De-Serialization
HeapMemory
ObjectInputStream
abc.ser
byte code
object
FileInputStream
abc.ser(file)
100%
Search
ENG IN
04-01-2023


Command Prompt
D:\IOOperations>javap java.io.ObjectOutputStream
Compiled from "ObjectOutputStream.java"
public class java.io.ObjectOutputStream extends java.io.OutputStream implements java.io.ObjectOutput,java.
tants {
public java.io.ObjectOutputStream(java.io.OutputStream) throws java.io.IOException;
protected java.io.ObjectOutputStream() throws java.io.IOException, java.lang.SecurityException;
public void useProtocolVersion(int) throws java.io.IOException;
public final void writeObject(java.lang.Object) throws java.io.IOException;
protected void writeObjectOverride(java.lang.Object) throws java.io.IOException;
public void writeUnshared(java.lang.Object) throws java.io.IOException;
public void defaultWriteObject() throws java.io.IOException;
public java.io.ObjectOutputStream$PutField putFields() throws java.io.IOException;
public void writeFields() throws java.io.IOException;
public void reset() throws java.io.IOException;
protected void annotateClass(java.lang.Class<?>) throws java.io.IOException;
protected void annotateProxyClass(java.lang.Class<?>) throws java.io.IOException;
protected java.lang.Object replaceObject(java.lang.Object) throws java.io.IOException;
protected boolean enableReplaceObject(boolean) throws java.lang.SecurityException;
protected void writeStreamHeader() throws java.io.IOException;
protected void writeClassDescriptor(java.io.ObjectStreamClass) throws java.io.IOException;
public void write(int) throws java.io.IOException;
public void write(byte[]) throws java.io.IOException;
public void write(byte[], int, int) throws java.io.IOException;
public void flush() throws java.io.IOException;
protected void drain() throws java.io.IOException;
public void close() throws java.io.IOException;
public void writeBoolean(boolean) throws java.io.IOException;
public void writeByte(int) throws java.io.IOException;
public void writeShort(int) throws java.io.IOException;
public void writeChar(int) throws java.io.IOException;
public void writeInt(int) throws java.io.IOException;
Nitin Manjunath
21°C
Mostly clear
Search
ENG
IN
20:16
04-01-2023

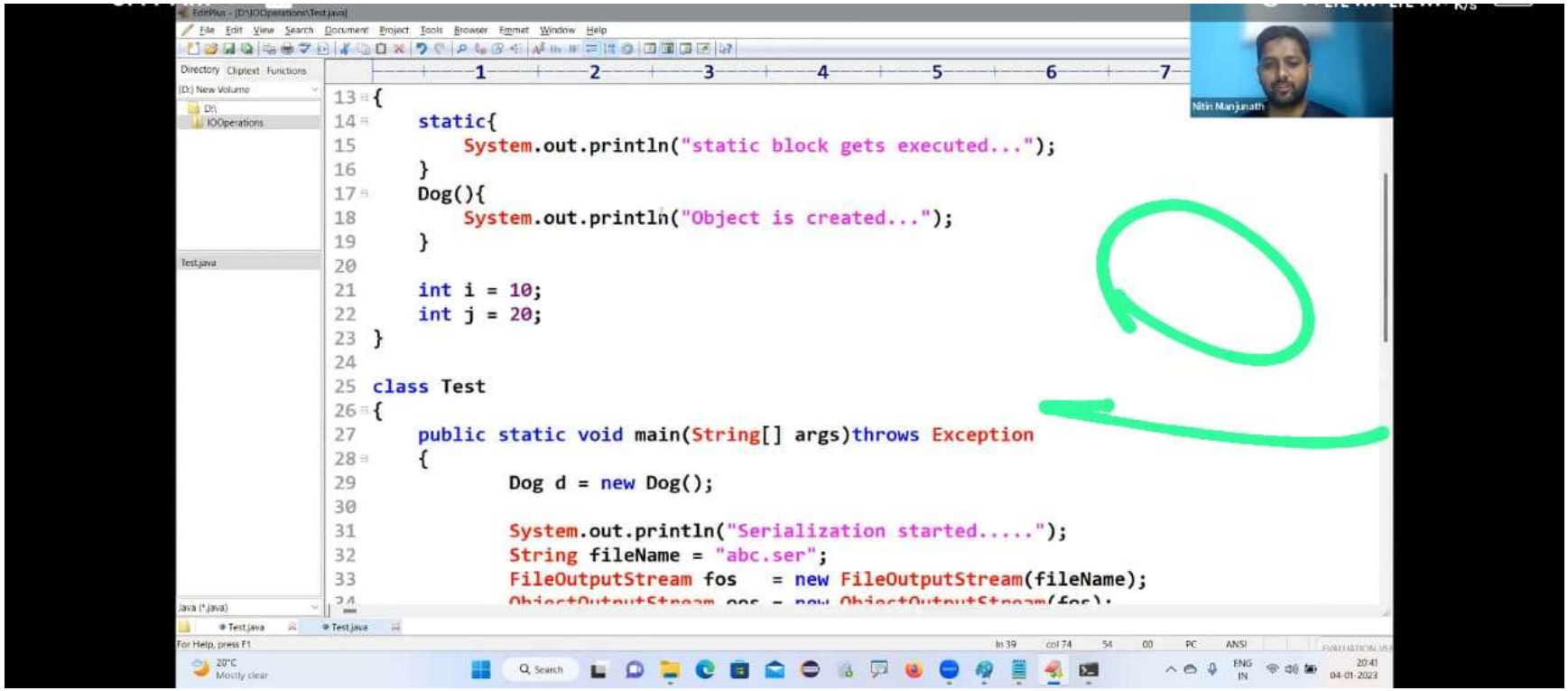

Command Prompt
D:\IOOperations>javap java.io.FileOutputStream
Compiled from "FileOutputStream.java"
public class java.io.FileOutputStream extends java.io.OutputStream {
public java.io.FileOutputStream(java.lang.String) throws java.io.FileNotFoundException;
public java.io.FileOutputStream(java.lang.String, boolean) throws java.io.FileNotFoundException;
public java.io.FileOutputStream(java.io.File) throws java.io.FileNotFoundException;
public java.io.FileOutputStream(java.io.File, boolean) throws java.io.FileNotFoundException;
public java.io.FileOutputStream(java.io.FileDescriptor);
public void write(int) throws java.io.IOException;
public void write(byte[]) throws java.io.IOException;
public void write(byte[], int, int) throws java.io.IOException;
public void close() throws java.io.IOException;
public final java.io.FileDescriptor getFD() throws java.io.IOException;
public java.nio.channels.FileChannel getChannel();
protected void finalize() throws java.io.IOException;
static void access$000(java.io.FileOutputStream) throws java.io.IOException;
static {};
}
D:\IOOperations>
Nitin Manjunath
21°C
Mostly clear
Search
ENG
IN
20:17
04-01-2023

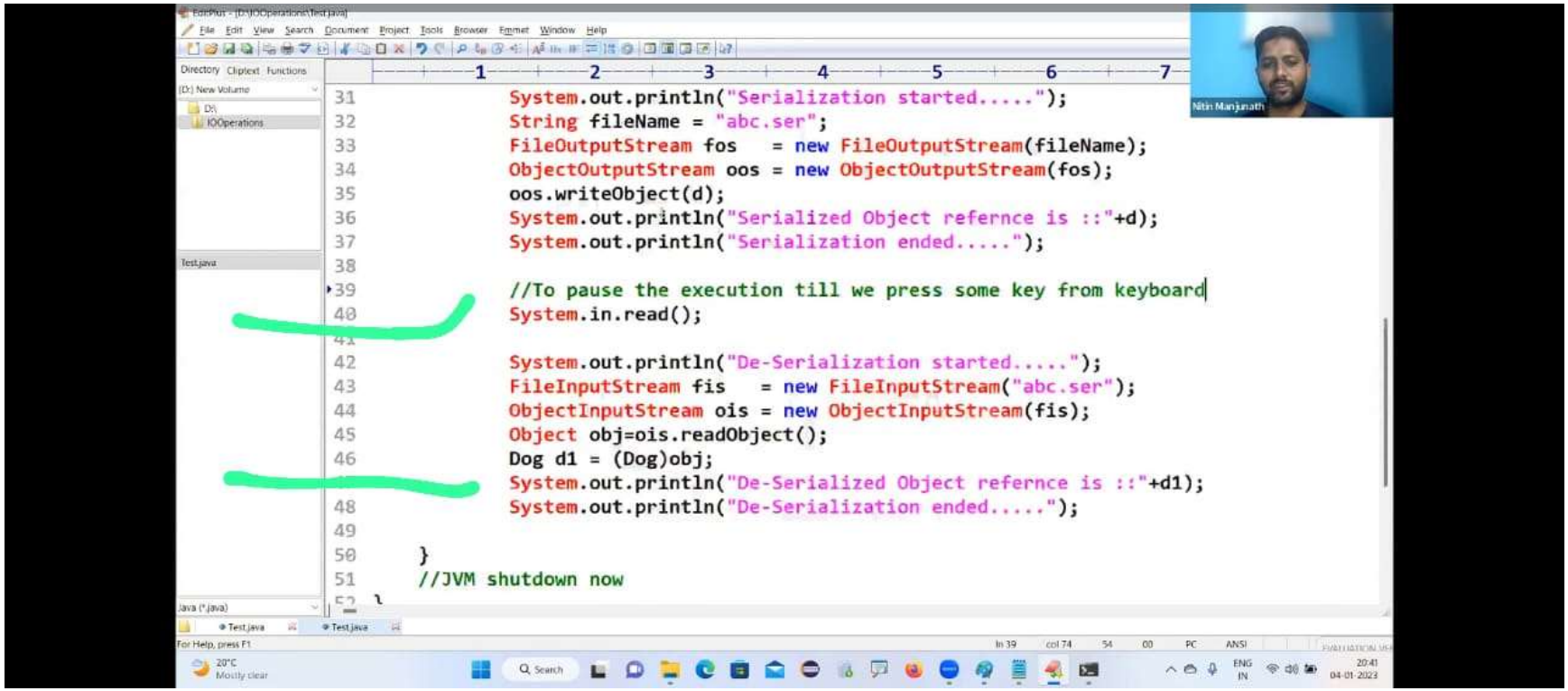

Command Prompt
D:\IOOperations>javap java.io.ObjectOutputStream
Compiled from "ObjectOutputStream.java"
public class java.io.ObjectOutputStream extends java.io.OutputStream implements java.io.ObjectOutput,java.io.Objec
public java.io.ObjectOutputStream(java.io.OutputStream) throws java.io.IOException;
protected java.io.ObjectOutputStream() throws java.io.IOException, java.lang.SecurityException;
public void useProtocolVersion(int) throws java.io.IOException;
public final void writeObject(java.lang.Object) throws java.io.IOException;
protected void writeObjectOverride(java.lang.Object) throws java.io.IOException;
public void writeUnshared(java.lang.Object) throws java.io.IOException;
public void defaultWriteObject() throws java.io.IOException;
public java.io.ObjectOutputStream$PutField putFields() throws java.io.IOException;
public void writeFields() throws java.io.IOException;
public void reset() throws java.io.IOException;
protected void annotateClass(java.lang.Class<?>) throws java.io.IOException;
protected void annotateProxyClass(java.lang.Class<?>) throws java.io.IOException;
protected java.lang.Object replaceObject(java.lang.Object) throws java.io.IOException;
protected boolean enableReplaceObject(boolean) throws java.lang.SecurityException;
protected void writeStreamHeader() throws java.io.IOException;
protected void writeClassDescriptor(java.io.ObjectStreamClass) throws java.io.IOException;
public void write(int) throws java.io.IOException;
public void write(byte[]) throws java.io.IOException;
public void write(byte[], int, int) throws java.io.IOException;
public void flush() throws java.io.IOException;
protected void drain() throws java.io.IOException;
public void close() throws java.io.IOException;
public void writeBoolean(boolean) throws java.io.IOException;
public void writeByte(int) throws java.io.IOException;
public void writeShort(int) throws java.io.IOException;
public void writeChar(int) throws java.io.IOException;
public void writeInt(int) throws java.io.IOException;
public void writeLong(long) throws java.io.IOException;
public void writeFloat(float) throws java.io.IOException;
Nitin Manjunath
21°C
Mostly clear
Search
ENG
IN
20:20
04-01-2023

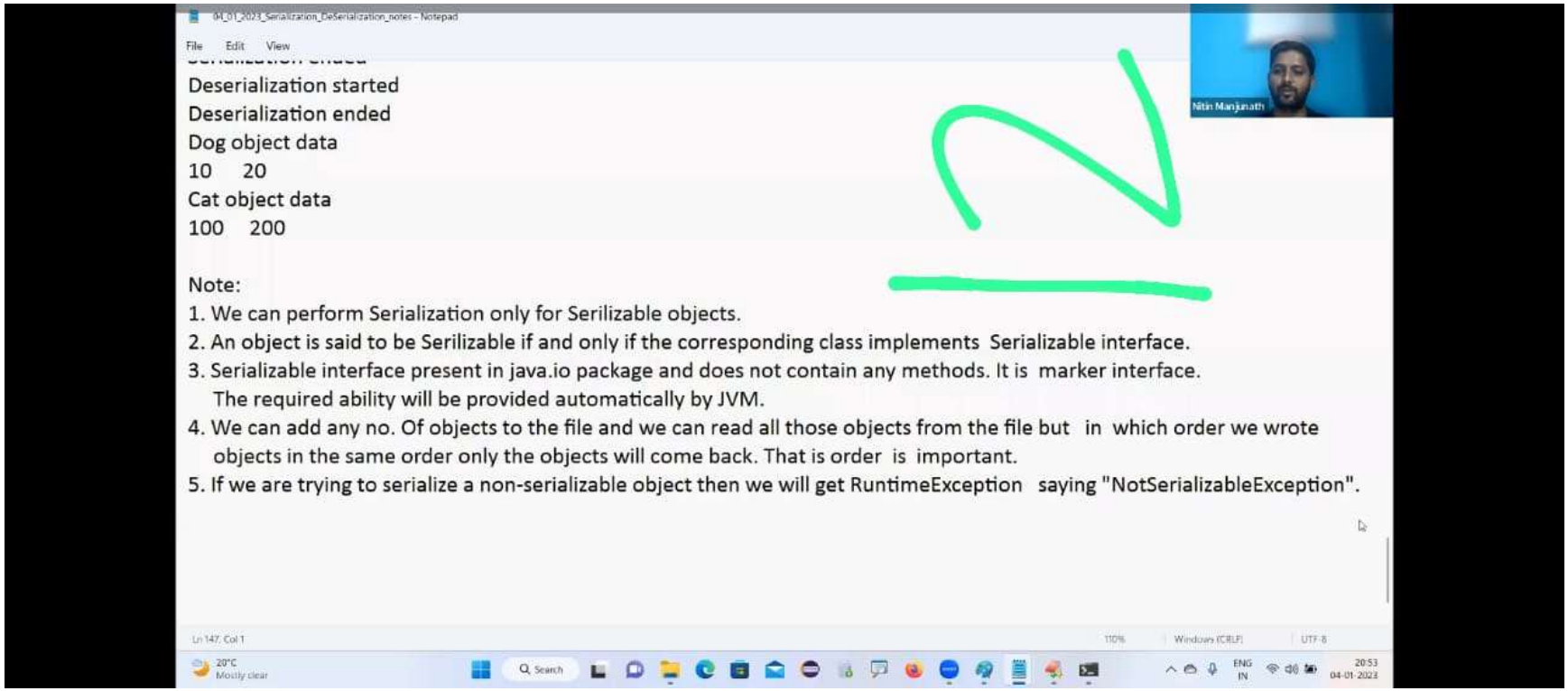

Command Prompt
Compiled from "ObjectInputStream.java"
public class java.io.ObjectInputStream extends java.io.InputStream implements java.io.ObjectInput,java.io.ObjectStreamConstants {
public java.io.ObjectInputStream(java.io.InputStream) throws java.io.IOException;
protected java.io.ObjectInputStream() throws java.io.IOException, java.lang.SecurityException;
public final java.lang.Object readObject() throws java.io.IOException, java.lang.ClassNotFoundException;
protected java.lang.Object readObjectOverride() throws java.io.IOException, java.lang.ClassNotFoundException;
public java.lang.Object readUnshared() throws java.io.IOException, java.lang.ClassNotFoundException;
public void defaultReadObject() throws java.io.IOException, java.lang.ClassNotFoundException;
public java.io.ObjectInputStream$GetField readFields() throws java.io.IOException, java.lang.ClassNotFoundException;
public void registerValidation(java.io.ObjectInputValidation, int) throws java.io.NotActiveException, java.io.InvalidObjectException;
protected java.lang.Class<?> resolveClass(java.io.ObjectStreamClass) throws java.io.IOException, java.lang.ClassNotFoundException;
protected java.lang.Class<?> resolveProxyClass(java.lang.String[]) throws java.io.IOException, java.lang.ClassNotFoundException;
protected java.lang.Object resolveObject(java.lang.Object) throws java.io.IOException;
protected boolean enableResolveObject(boolean) throws java.lang.SecurityException;
protected void readStreamHeader() throws java.io.IOException, java.io.StreamCorruptedException;
protected java.io.ObjectStreamClass readClassDescriptor() throws java.io.IOException, java.lang.ClassNotFoundException;
public int read() throws java.io.IOException;
public int read(byte[], int, int) throws java.io.IOException;
public int available() throws java.io.IOException;
public void close() throws java.io.IOException;
public boolean readBoolean() throws java.io.IOException;
public byte readByte() throws java.io.IOException;
public int readUnsignedByte() throws java.io.IOException;
public char readChar() throws java.io.IOException;
public short readShort() throws java.io.IOException;
public int readUnsignedShort() throws java.io.IOException;
public int readInt() throws java.io.IOException;
public long readLong() throws java.io.IOException;
public float readFloat() throws java.io.IOException;
public double readDouble() throws java.io.IOException;
public void readFully(byte[]) throws java.io.IOException;
public void readFully(byte[], int, int) throws java.io.IOException;
public int skipBytes(int) throws java.io.IOException;
public java.lang.String readLine() throws java.io.IOException;
public java.lang.String readUTF() throws java.io.IOException;
java.lang.String readTypeString() throws java.io.IOException;
static void access$000(java.io.ObjectInputStream, sun.misc.ObjectInputFilter);
static sun.misc.ObjectInputFilter access$100(java.io.ObjectInputStream);
static void access$200(java.io.ObjectInputStream, java.lang.Class, int) throws java.io.InvalidClassException;
static int access$500(java.io.ObjectInputStream);
Nitin Manjunath
21°C
Mostly clear
Search
ENG
IN
20:23
04-01-2023

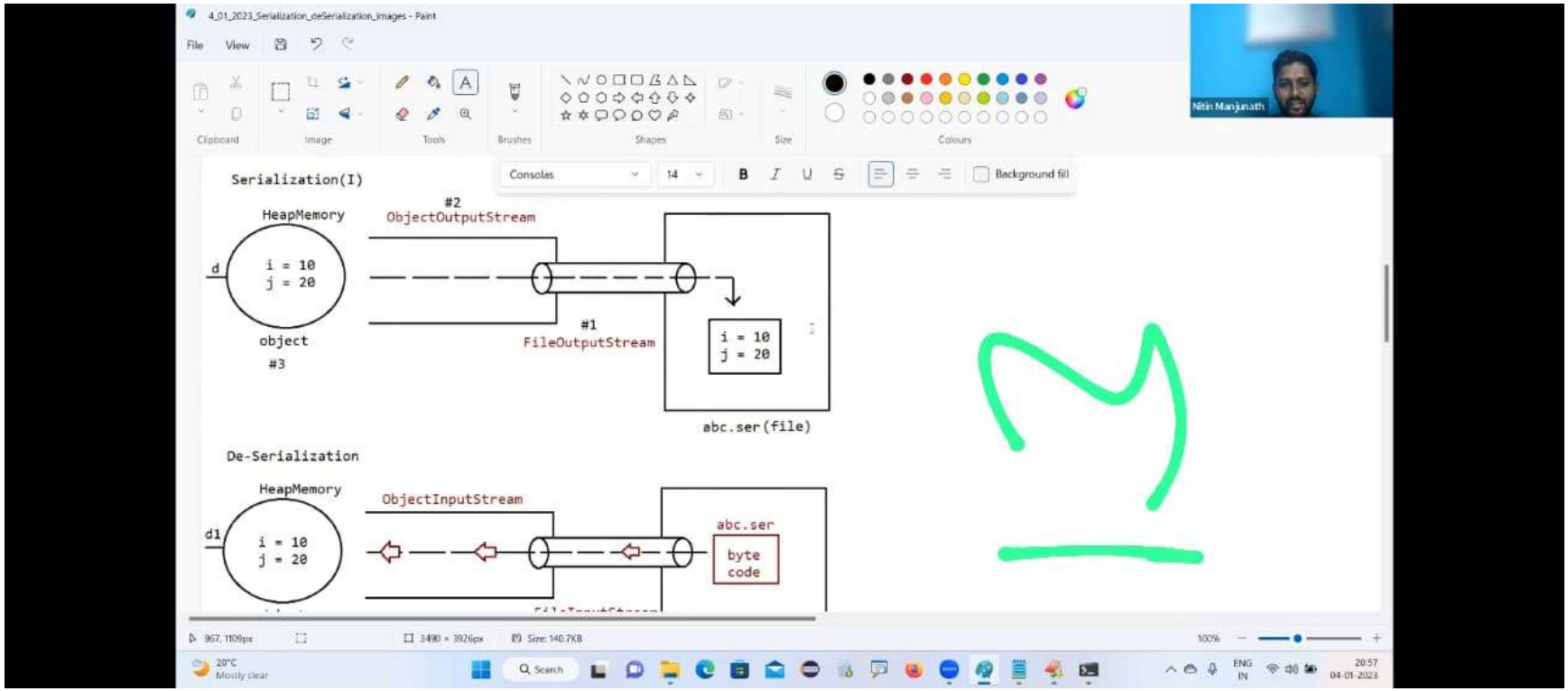


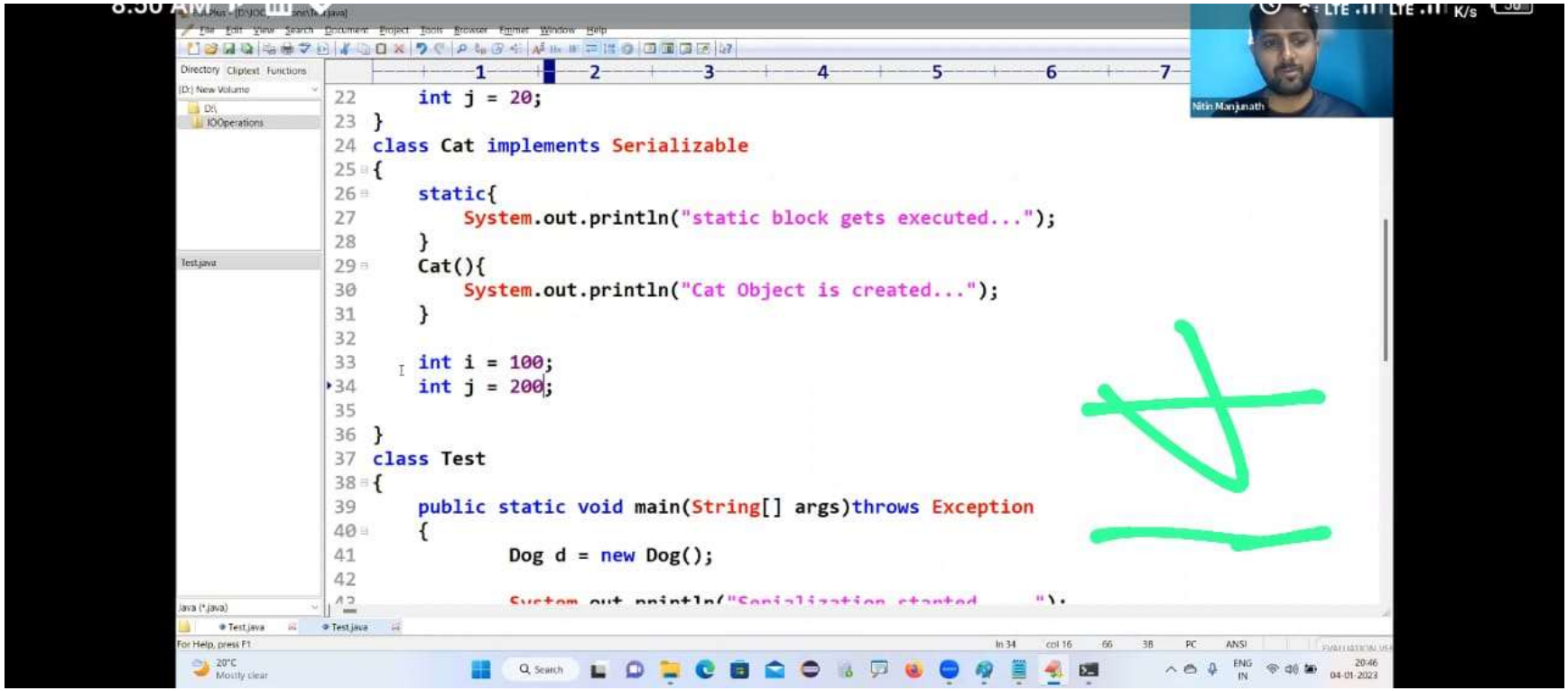

```
1  import java.io.*;
2  /*
3   public java.io.ObjectOutputStream(java.io.OutputStream) throws java.io.IOException;
4   public java.io.FileOutputStream(java.lang.String) throws java.io.FileNotFoundException;
5   public final void writeObject(java.lang.Object) throws java.io.IOException;
6
7   public java.io.ObjectInputStream(java.io.InputStream) throws java.io.IOException;
8   public java.io.FileInputStream(java.lang.String) throws java.io.FileNotFoundException;
9   public final java.lang.Object readObject() throws java.io.IOException, java.lang.ClassN
10 */
11
12 class Dog implements Serializable
13 {
14     static{
15         System.out.println("static block gets executed...");
16     }
17     Dog(){
18         System.out.println("Object is created...");
19     }
20
21     int i = 10;
22     int j = 20;
```

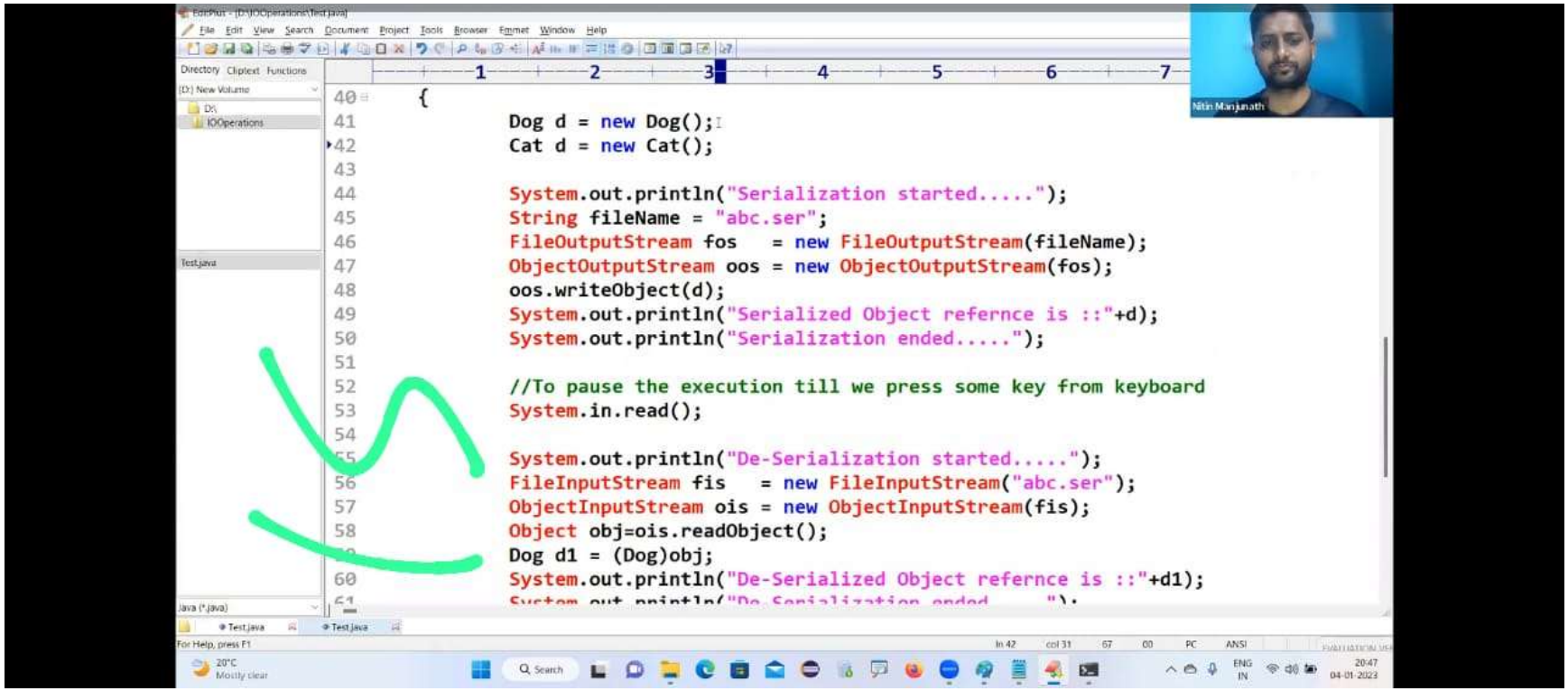

Nitin Manjunath
{
static{
System.out.println("static block gets executed...");
}
Dog(){
System.out.println("Object is created...");
}
int i = 10;
int j = 20;
}
class Test
{
public static void main(String[] args)throws Exception
{
Dog d = new Dog();
System.out.println("Serialization started.....");
String fileName = "abc.ser";
FileOutputStream fos = new FileOutputStream(fileName);

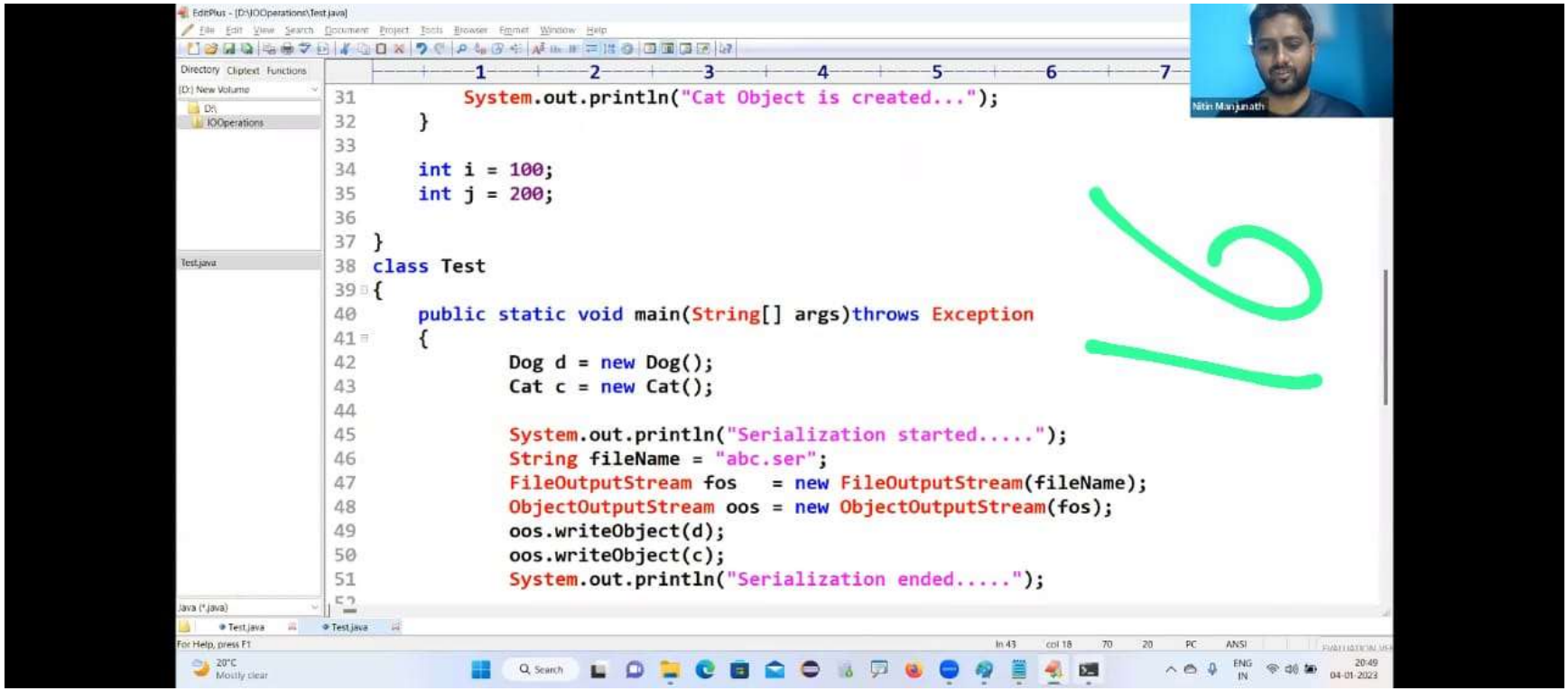

Nitin Manjunath
System.out.println("Serialization started.....");
String fileName = "abc.ser";
FileOutputStream fos = new FileOutputStream(fileName);
ObjectOutputStream oos = new ObjectOutputStream(fos);
oos.writeObject(d);
System.out.println("Serialized Object refernce is ::"+d);
System.out.println("Serialization ended.....");
//To pause the execution till we press some key from keyboard
System.in.read();
System.out.println("De-Serialization started.....");
FileInputStream fis = new FileInputStream("abc.ser");
ObjectInputStream ois = new ObjectInputStream(fis);
Object obj=ois.readObject();
Dog d1 = (Dog)obj;
System.out.println("De-Serialized Object refernce is ::"+d1);
System.out.println("De-Serialization ended.....");
}
//JVM shutdown now

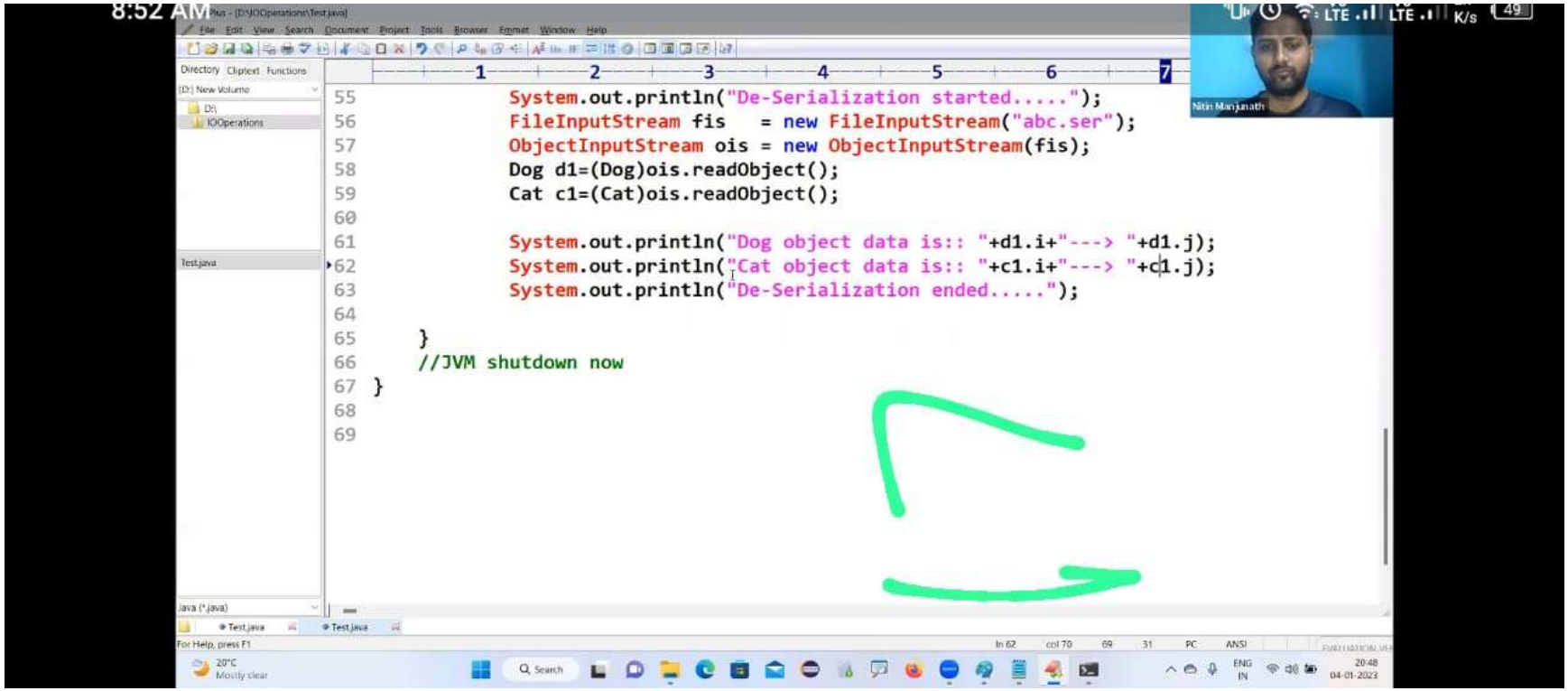

04_01_2023_Serialization_DeSerialization_notes – Notepad
File Edit View
Deserialization started
Deserialization ended
Dog object data
10 20
Cat object data
100 200
Note:
1. We can perform Serialization only for Serilizable objects.
2. An object is said to be Serilizable if and only if the corresponding class implements Serializable interface.
3. Serializable interface present in java.io package and does not contain any methods. It is marker interface.
The required ability will be provided automatically by JVM.
4. We can add any no. Of objects to the file and we can read all those objects from the file but in which order we wrote objects in the same order only the objects will come back. That is order is important.
5. If we are trying to serialize a non-serializable object then we will get RuntimeException saying "NotSerializableException".
Nitin Manjunath
Ln 147, Col 1
110%
Windows (CRLF)
UTF-8
20°C
Mostly clear
Search
ENG
IN
20:53
04-01-2023

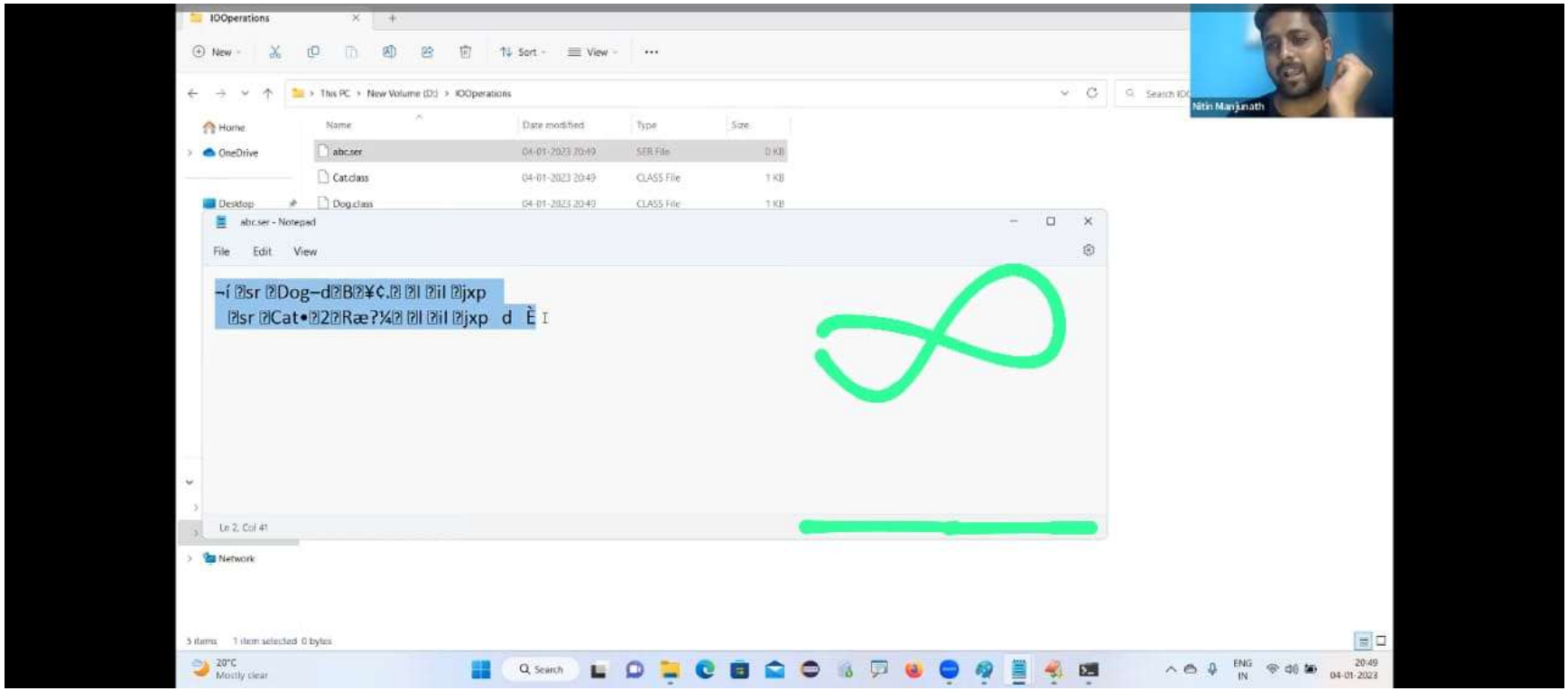

4_01_2023_Serialization_deSerialization_images - Paint
File
View
Nitin Manjunath
Serialization(I)
HeapMemory
d
i = 10
j = 20
object
#3
#2
ObjectOutputStream
#1
FileOutputStream
i = 10
j = 20
abc.ser(file)
De-Serialization
HeapMemory
d1
i = 10
j = 20
ObjectInputStream
abc.ser
byte
code

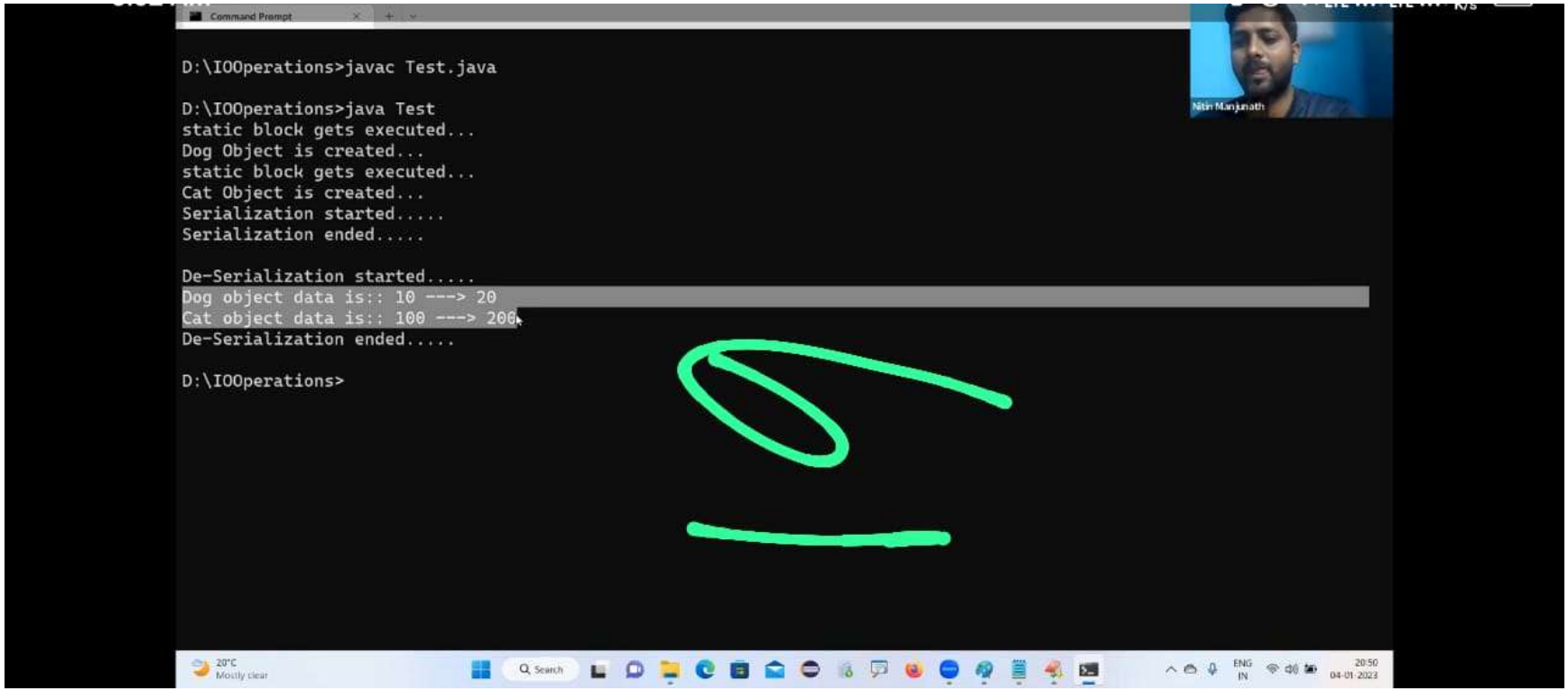

Nitin Manjunath
22 int j = 20;
23 }
24 class Cat implements Serializable
25 {
26 static{
27 System.out.println("static block gets executed...");
28 }
29 Cat(){
30 System.out.println("Cat Object is created...");
31 }
32
33 int i = 100;
34 int j = 200;
35
36 }
37 class Test
38 {
39 public static void main(String[] args)throws Exception
40 {
41 Dog d = new Dog();
42
Directory Cliptext Functions
(D:) New Volume
D:\
IOOperations
Test.java
Java (*.java)
For Help, press F1
20°C
Mostly clear
Search
ENG
IN
20:46
04-01-2023

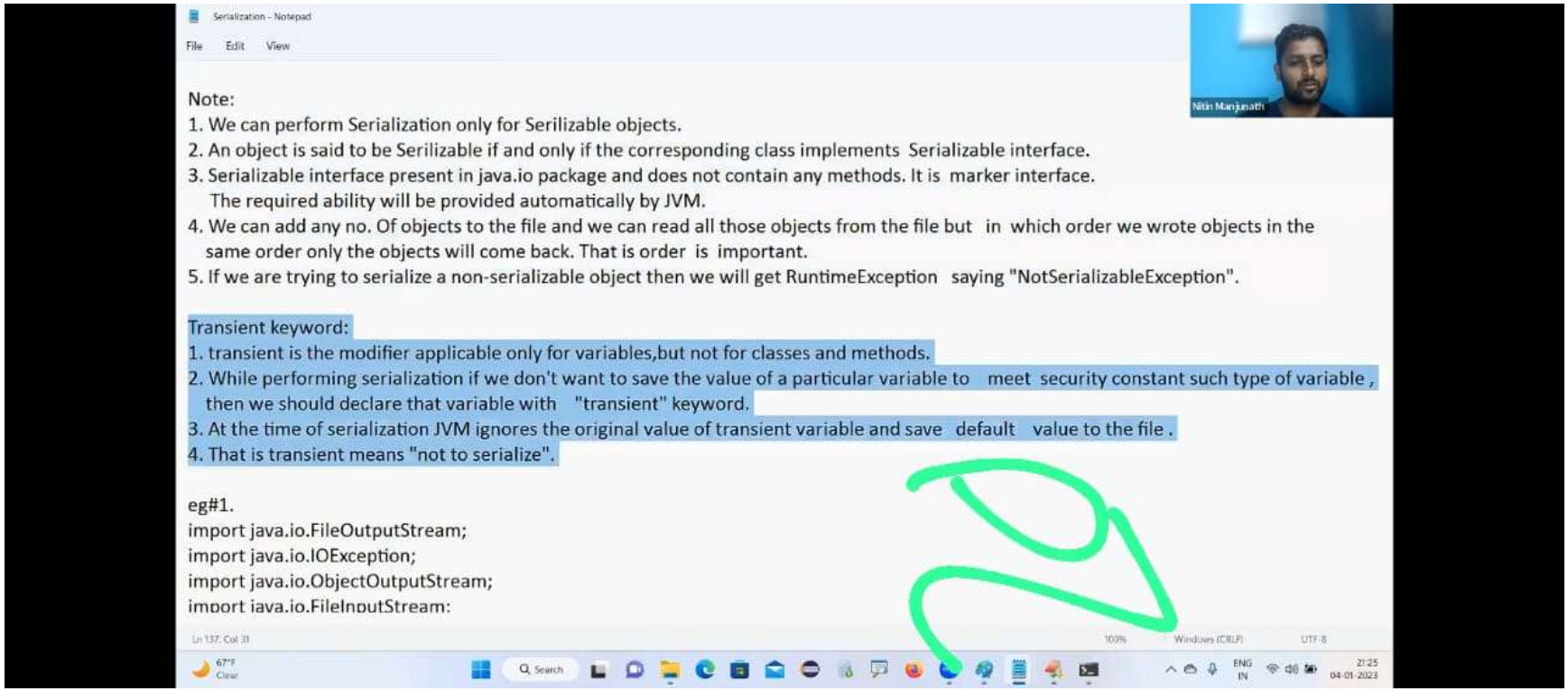

Nitin Manjunath
{
Dog d = new Dog();
Cat d = new Cat();
System.out.println("Serialization started.....");
String fileName = "abc.ser";
FileOutputStream fos = new FileOutputStream(fileName);
ObjectOutputStream oos = new ObjectOutputStream(fos);
oos.writeObject(d);
System.out.println("Serialized Object refernce is ::"+d);
System.out.println("Serialization ended.....");
//To pause the execution till we press some key from keyboard
System.in.read();
System.out.println("De-Serialization started.....");
FileInputStream fis = new FileInputStream("abc.ser");
ObjectInputStream ois = new ObjectInputStream(fis);
Object obj=ois.readObject();
Dog d1 = (Dog)obj;
System.out.println("De-Serialized Object refernce is ::"+d1);

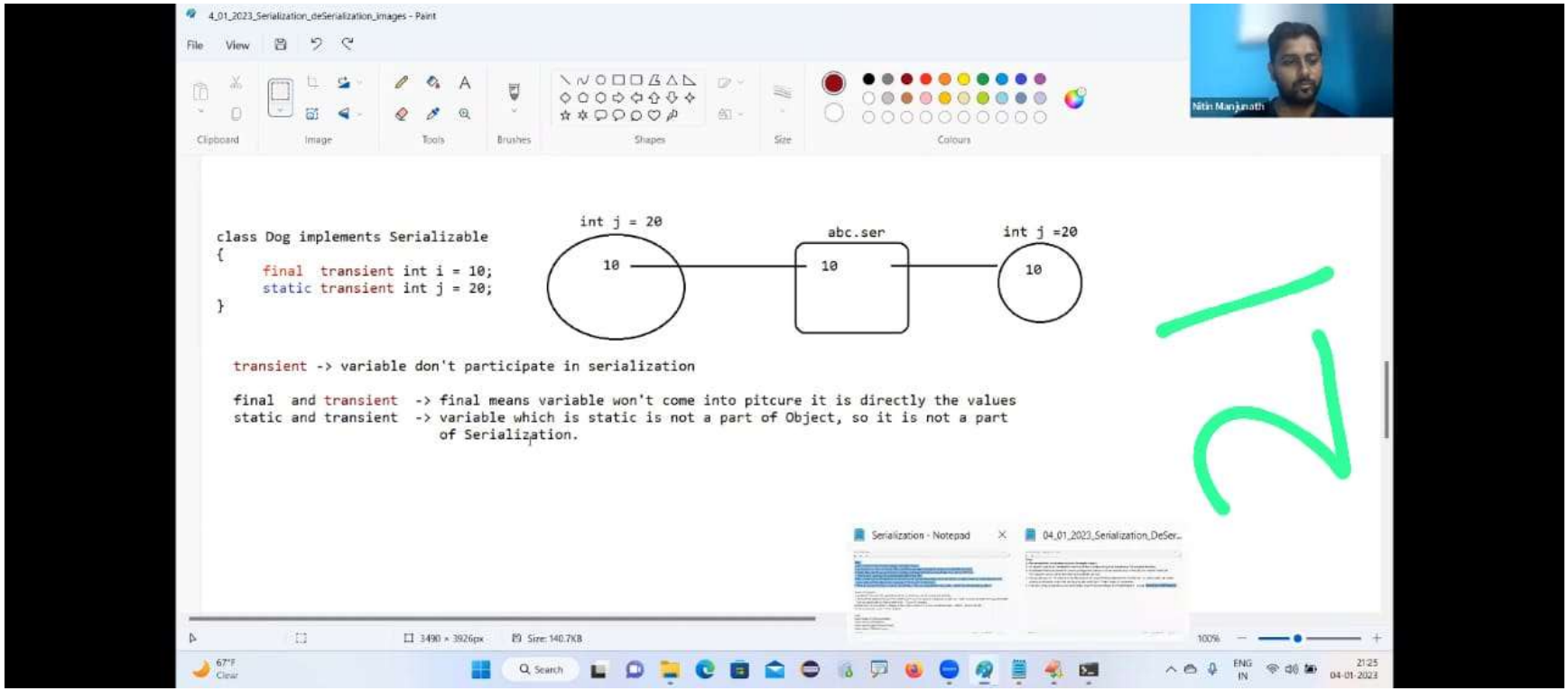

EditPlus - [D:\IOOperations\Test.java]
Directory Cliptext Functions
(D:) New Volume
D:\
IOOperations
Test.java
Java (*.java)
Test.java
Test.java
For Help, press F1
Nitin Manjunath
System.out.println("Cat Object is created...");
}
int i = 100;
int j = 200;
}
class Test
{
public static void main(String[] args)throws Exception
{
Dog d = new Dog();
Cat c = new Cat();
System.out.println("Serialization started.....");
String fileName = "abc.ser";
FileOutputStream fos = new FileOutputStream(fileName);
ObjectOutputStream oos = new ObjectOutputStream(fos);
oos.writeObject(d);
oos.writeObject(c);
System.out.println("Serialization ended.....");
16
20°C
Mostly clear
Search
ENG
IN
20:49
04-01-2023

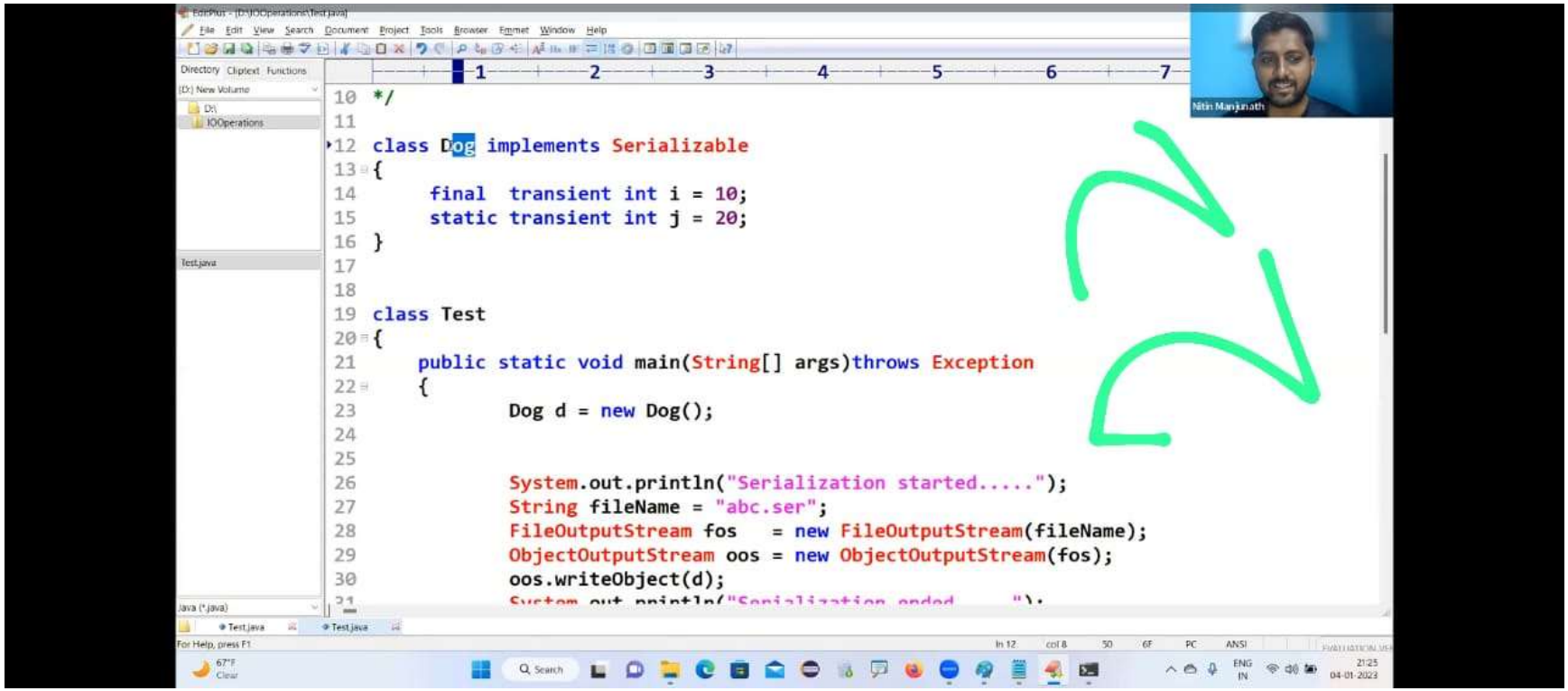

8:52 AM
Nitin Manjunath
System.out.println("De-Serialization started.....");
FileInputStream fis = new FileInputStream("abc.ser");
ObjectInputStream ois = new ObjectInputStream(fis);
Dog d1=(Dog)ois.readObject();
Cat c1=(Cat)ois.readObject();
System.out.println("Dog object data is:: "+d1.i+"---> "+d1.j);
System.out.println("Cat object data is:: "+c1.i+"---> "+c1.j);
System.out.println("De-Serialization ended.....");
}
//JVM shutdown now
}

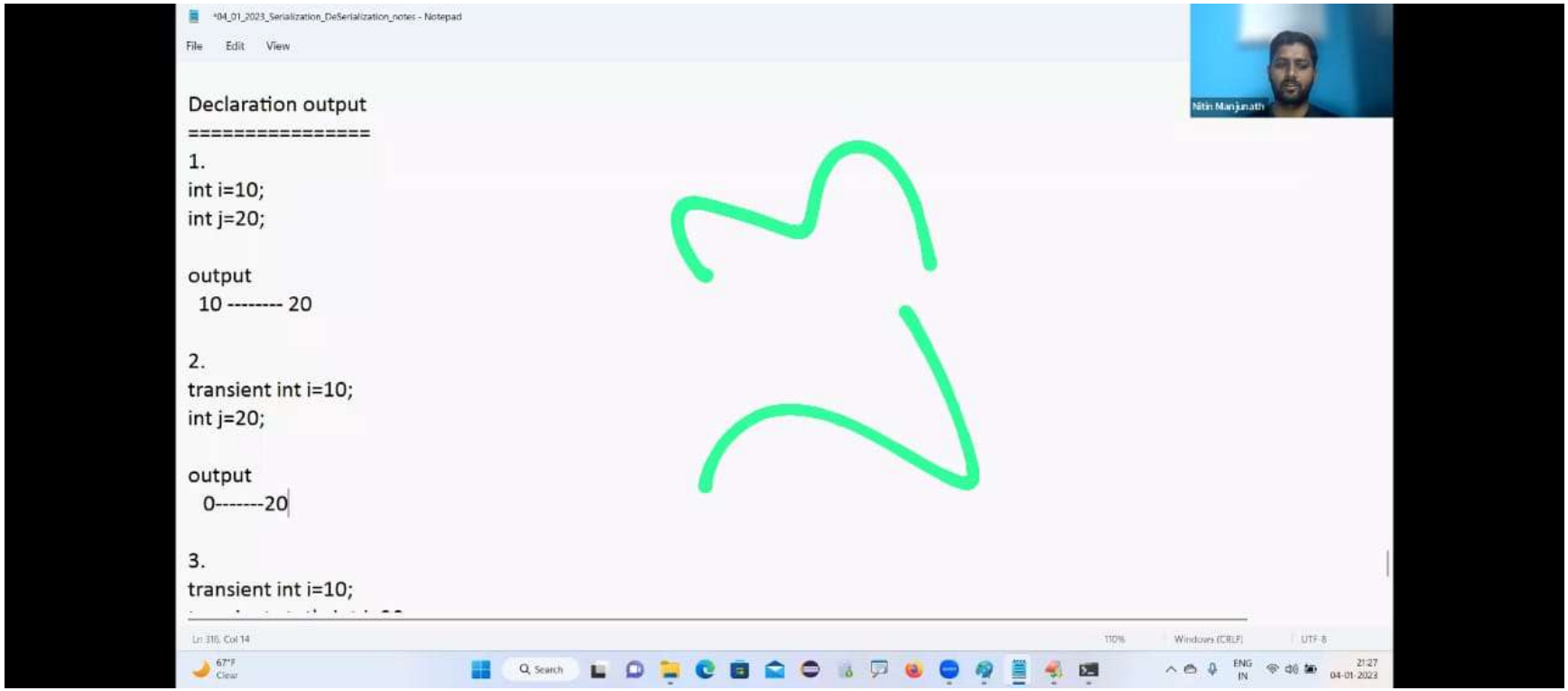

IOOperations
New
Sort
View
This PC > New Volume (D:) > IOOperations
Home
OneDrive
Desktop
Network
Name
Date modified
Type
Size
abc.ser
04-01-2023 20:49
SER File
0 KB
Cat.class
04-01-2023 20:49
CLASS File
1 KB
Dog.class
04-01-2023 20:49
CLASS File
1 KB
abc.ser - Notepad
File
Edit
View
¬í ?sr ?Dog−d?B?¥¢.? ?l ?il ?jxp
?sr ?Cat•?2?Ræ?¼? ?l ?il ?jxp d È
Ln 2, Col 41
5 items
1 item selected 0 bytes
20°C
Mostly clear
Search
ENG
IN
20:49
04-01-2023
Nitin Manjunath


Command Prompt
D:\IOOperations>javac Test.java
D:\IOOperations>java Test
static block gets executed...
Dog Object is created...
static block gets executed...
Cat Object is created...
Serialization started.....
Serialization ended.....
De-Serialization started.....
Dog object data is:: 10 ---> 20
Cat object data is:: 100 ---> 200
De-Serialization ended.....
D:\IOOperations>
Nitin Manjunath
20°C
Mostly clear
Search
ENG
IN
20:50
04-01-2023


Serialization - Notepad
File Edit View
Nitin Manjunath
Note:
1. We can perform Serialization only for Serilizable objects.
2. An object is said to be Serilizable if and only if the corresponding class implements Serializable interface.
3. Serializable interface present in java.io package and does not contain any methods. It is marker interface.
The required ability will be provided automatically by JVM.
4. We can add any no. Of objects to the file and we can read all those objects from the file but in which order we wrote objects in the same order only the objects will come back. That is order is important.
5. If we are trying to serialize a non-serializable object then we will get RuntimeException saying "NotSerializableException".
Transient keyword:
1. transient is the modifier applicable only for variables,but not for classes and methods.
2. While performing serialization if we don't want to save the value of a particular variable to meet security constant such type of variable , then we should declare that variable with "transient" keyword.
3. At the time of serialization JVM ignores the original value of transient variable and save default value to the file .
4. That is transient means "not to serialize".
eg#1.
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.ObjectOutputStream;
import java.io.FileInputStream;
Ln 137, Col 31
100%
Windows (CRLF)
UTF-8
67°F
Clear
Search
ENG
IN
21:25
04-01-2023


4_01_2023_Serialization_deSerialization_images - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
class Dog implements Serializable
{
final transient int i = 10;
static transient int j = 20;
}
int j = 20
10
abc.ser
10
int j =20
10
transient -> variable don't participate in serialization
final and transient -> final means variable won't come into pitcure it is directly the values
static and transient -> variable which is static is not a part of Object, so it is not a part of Serialization.
Serialization - Notepad
04_01_2023_Serialization_DeSer...
3490 × 3926px
Size: 140.7KB
100%
67°F
Clear
Search
ENG
IN
21:25
04-01-2023


Nitin Manjunath
*/
class Dog implements Serializable
{
final transient int i = 10;
static transient int j = 20;
}
class Test
{
public static void main(String[] args)throws Exception
{
Dog d = new Dog();
System.out.println("Serialization started.....");
String fileName = "abc.ser";
FileOutputStream fos = new FileOutputStream(fileName);
ObjectOutputStream oos = new ObjectOutputStream(fos);
oos.writeObject(d);


*04_01_2023_Serialization_DeSerialization_notes - Notepad
File Edit View
Declaration output
==========
1.
int i=10;
int j=20;
output
10 -------- 20
2.
transient int i=10;
int j=20;
output
0-------20
3.
transient int i=10;
Nitin Manjunath
Ln 310, Col 14
110%
Windows (CRLF)
UTF-8
67°F
Clear
Search
ENG
IN
21:27
04-01-2023

04_01_2023_Serialization_DeSerialization_notes - Notepad

File Edit View

int j=20;

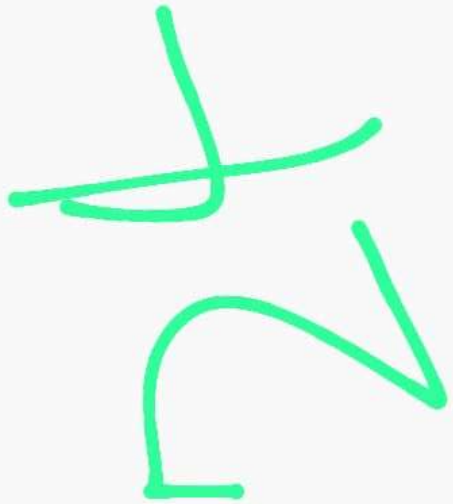
output
0-------20

3.
transient int i=10;
transient static int j=20;

output
0--------20

4.
transient final int i=10;
transient int j=20;

5.
transient final int i=10;
transient static int j=20;



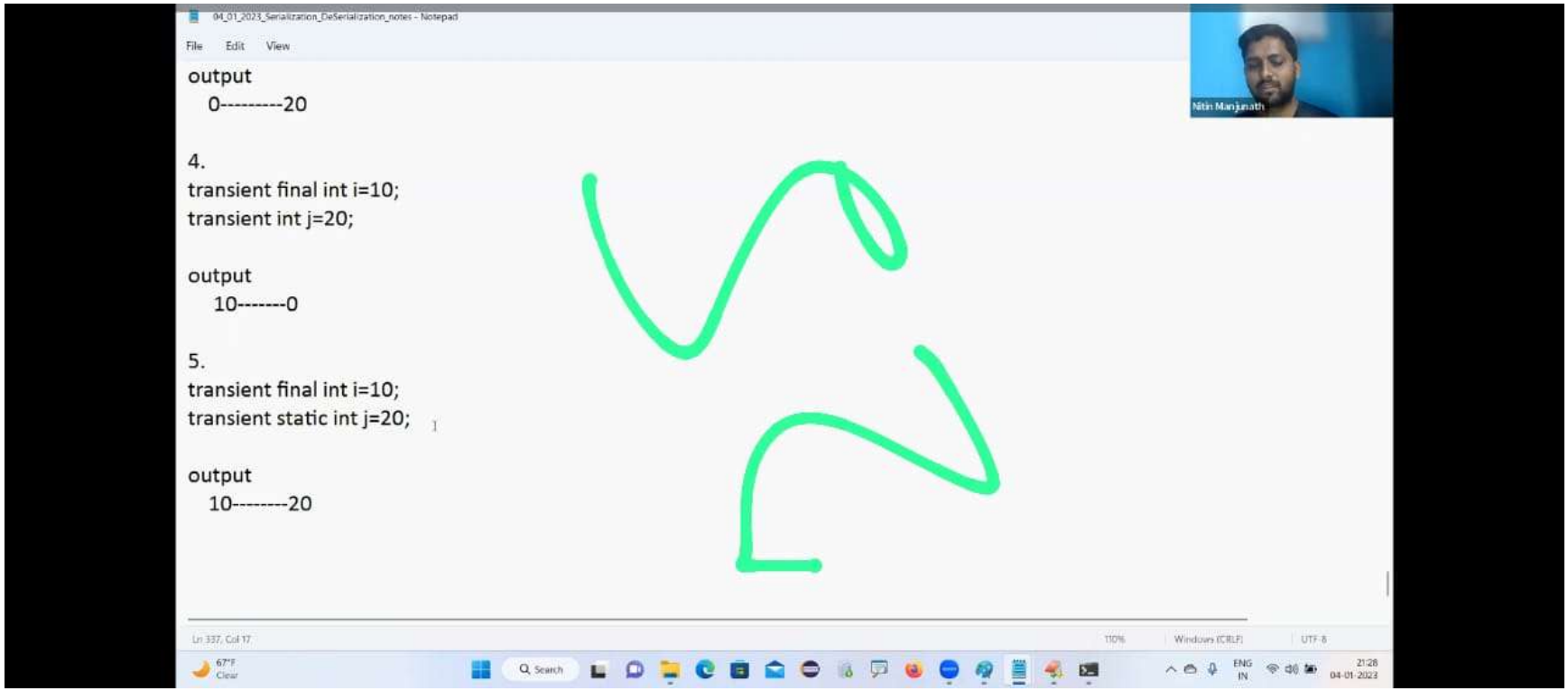
Ln 323, Col 17 110% Windows (CRLF) UTF-8

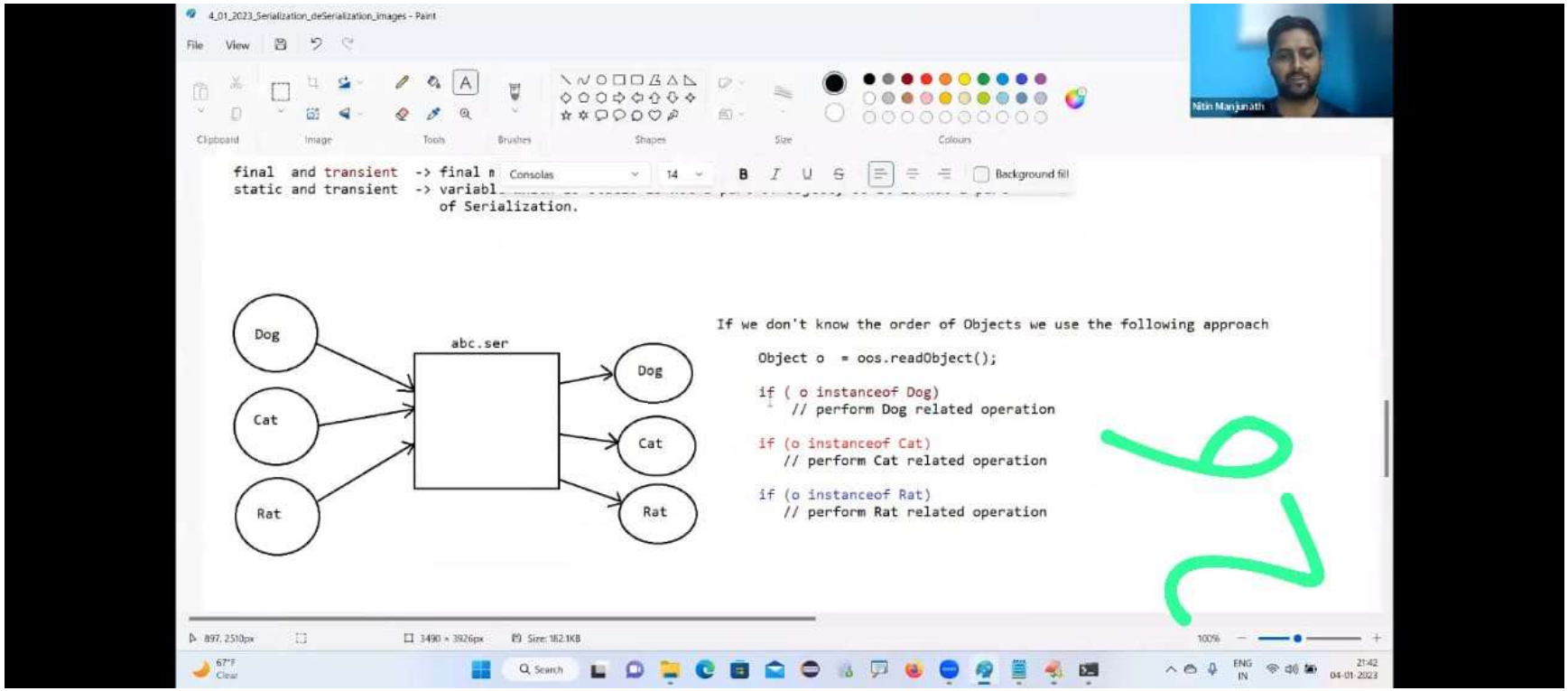
67°F Clear

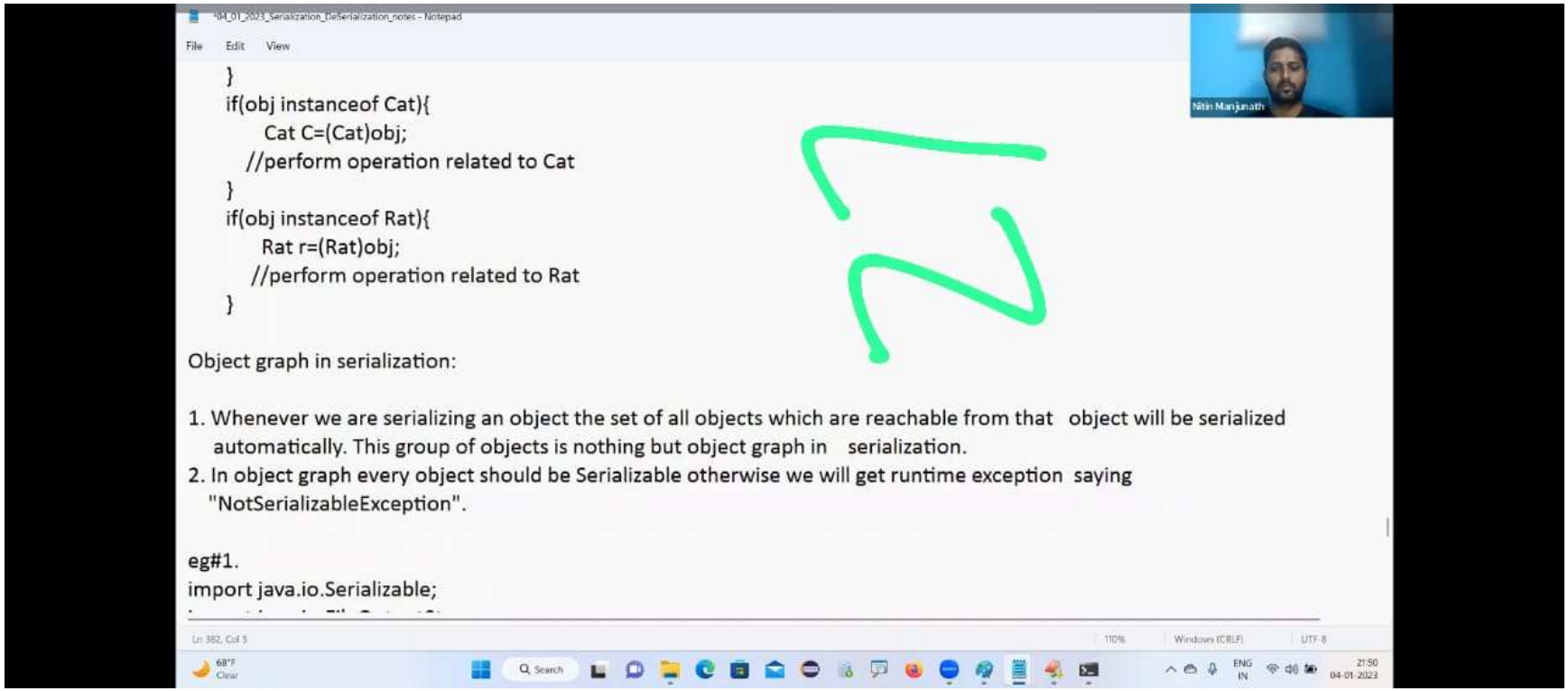
Search

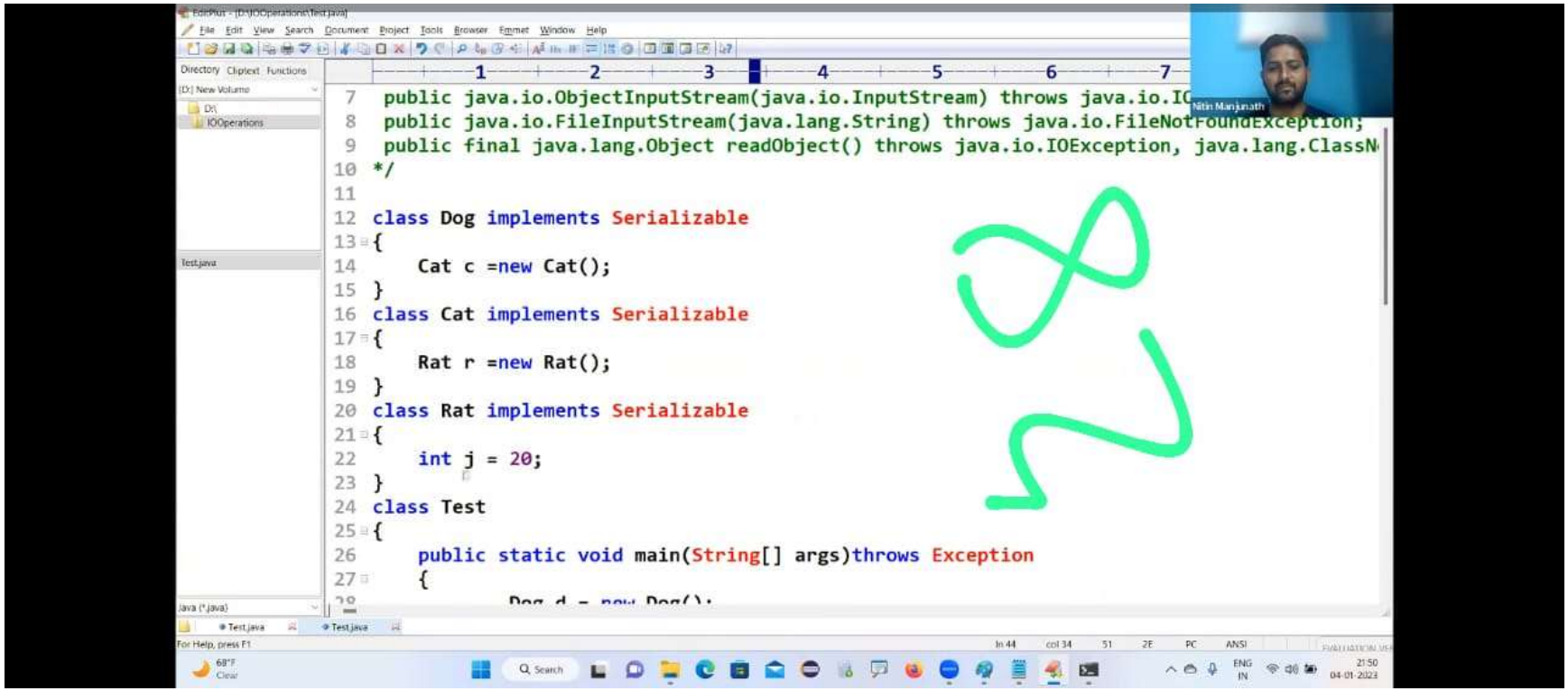
ENG IN

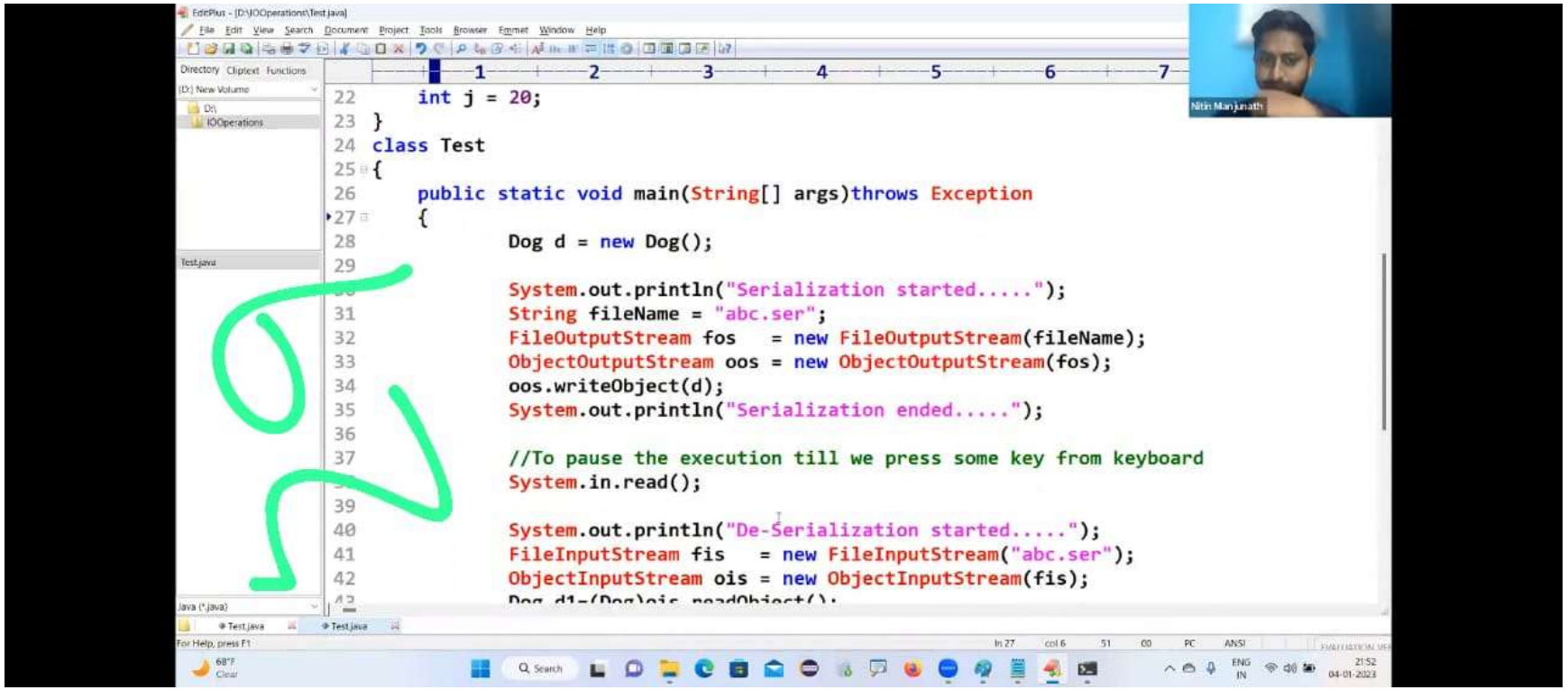
21:28 04-01-2023

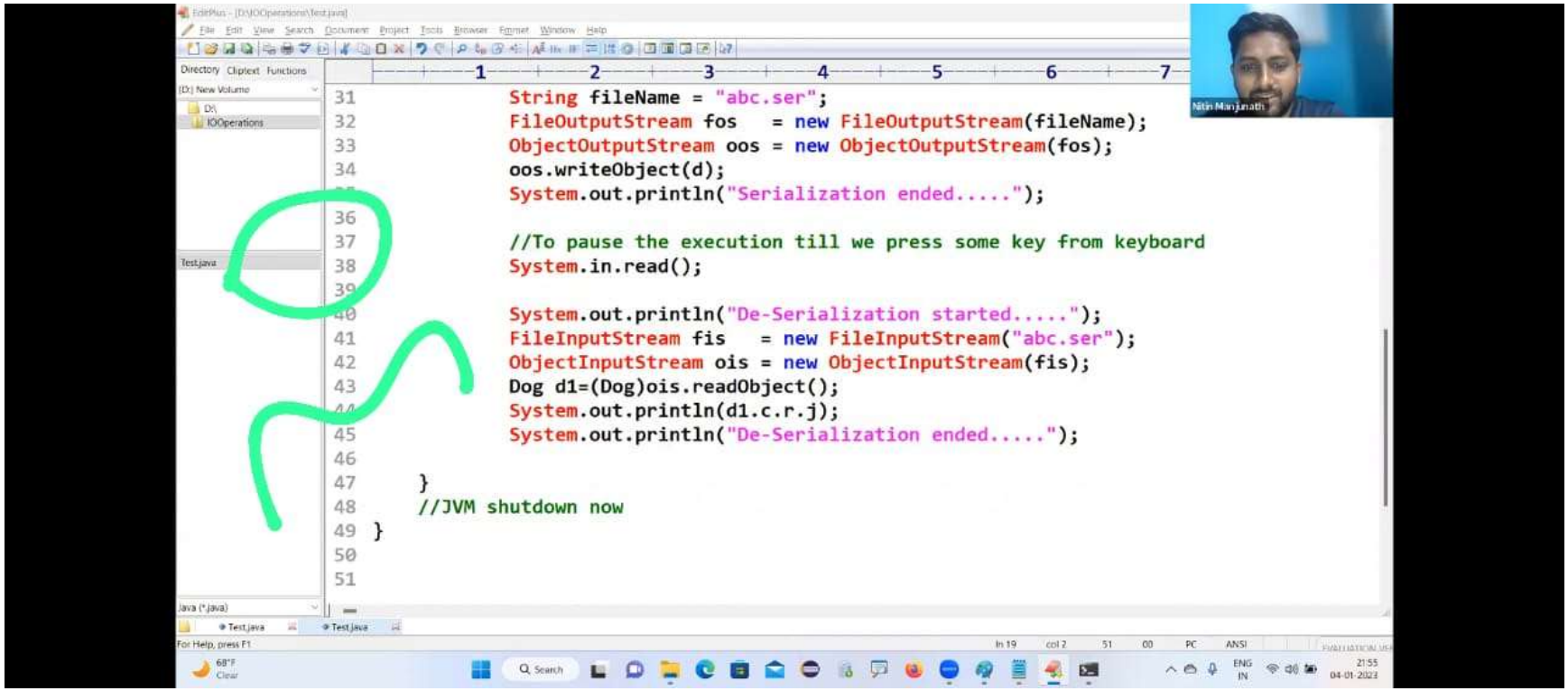

04_01_2023_Serialization_DeSerialization_notes - Notepad
File Edit View
output
0---------20
4.
transient final int i=10;
transient int j=20;
output
10-------0
5.
transient final int i=10;
transient static int j=20;
output
10--------20
Nitin Manjunath
Ln 337, Col 17
110%
Windows (CRLF)
UTF-8
67°F
Clear
Search
ENG
IN
21:28
04-01-2023

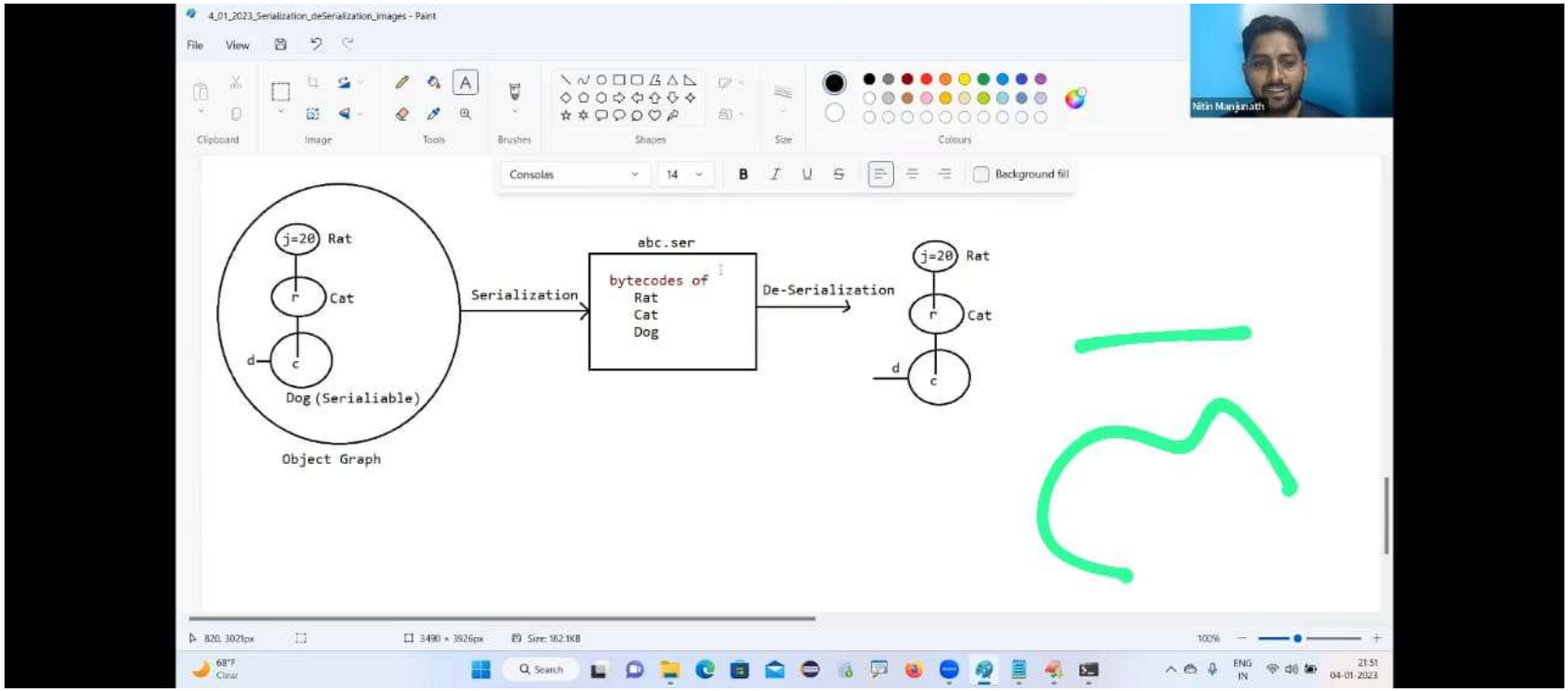

4_01_2023_Serialization_deSerialization_Images - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
final and transient -> final
static and transient -> variabl
of Serialization.
Consolas
14
Background fill
Dog
Cat
Rat
abc.ser
Dog
Cat
Rat
If we don't know the order of Objects we use the following approach
Object o = oos.readObject();
if (o instanceof Dog)
// perform Dog related operation
if (o instanceof Cat)
// perform Cat related operation
if (o instanceof Rat)
// perform Rat related operation
26
897, 2510px
3490 × 3926px
Size: 182.1KB
100%
67°F
Clear
Search
ENG
IN
21:42
04-01-2023

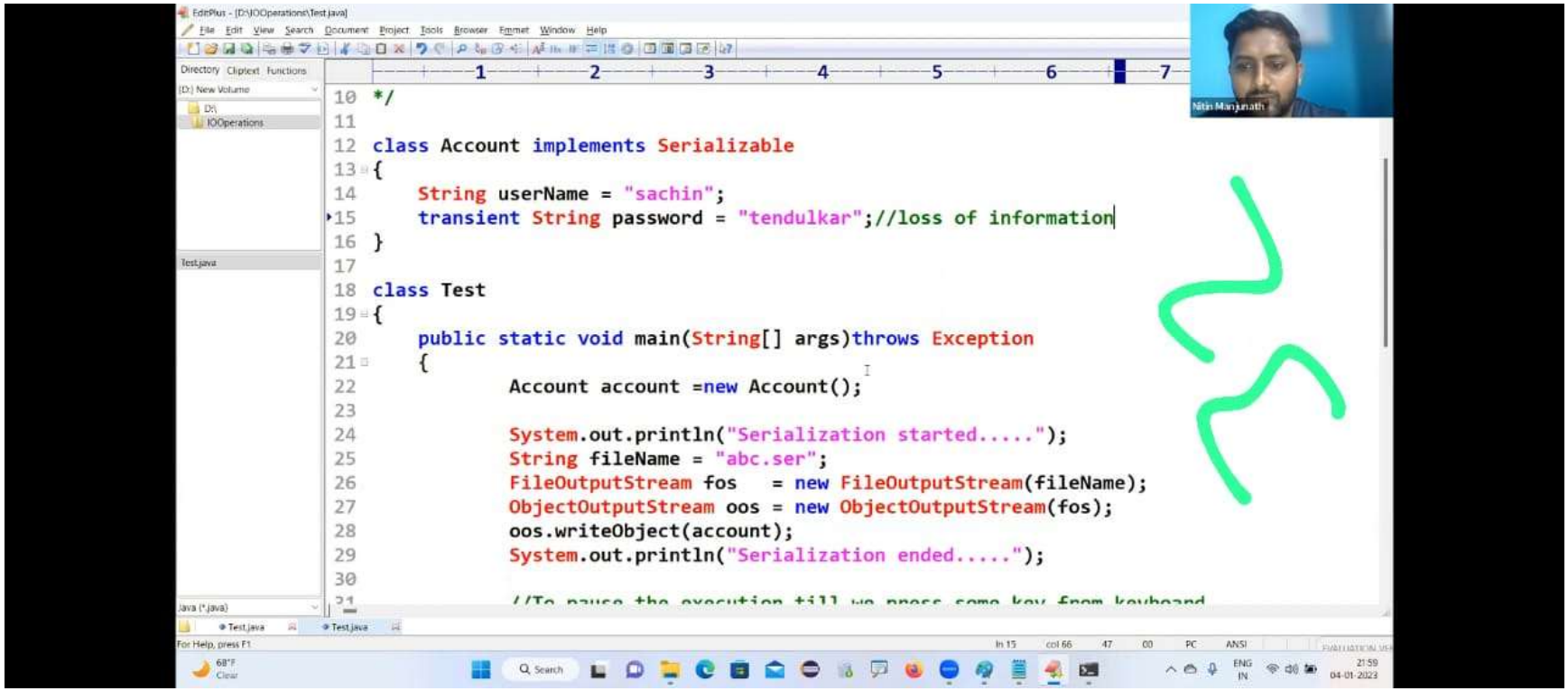

*04_01_2023_Serialization_DeSerialization_notes - Notepad
File Edit View
}
if(obj instanceof Cat){
Cat C=(Cat)obj;
//perform operation related to Cat
}
if(obj instanceof Rat){
Rat r=(Rat)obj;
//perform operation related to Rat
}
Object graph in serialization:
1. Whenever we are serializing an object the set of all objects which are reachable from that object will be serialized automatically. This group of objects is nothing but object graph in serialization.
2. In object graph every object should be Serializable otherwise we will get runtime exception saying "NotSerializableException".
eg#1.
import java.io.Serializable;
Nitin Manjunath
Ln 382, Col 5
110%
Windows (CRLF)
UTF-8
68°F
Clear
Search
ENG
IN
21:50
04-01-2023

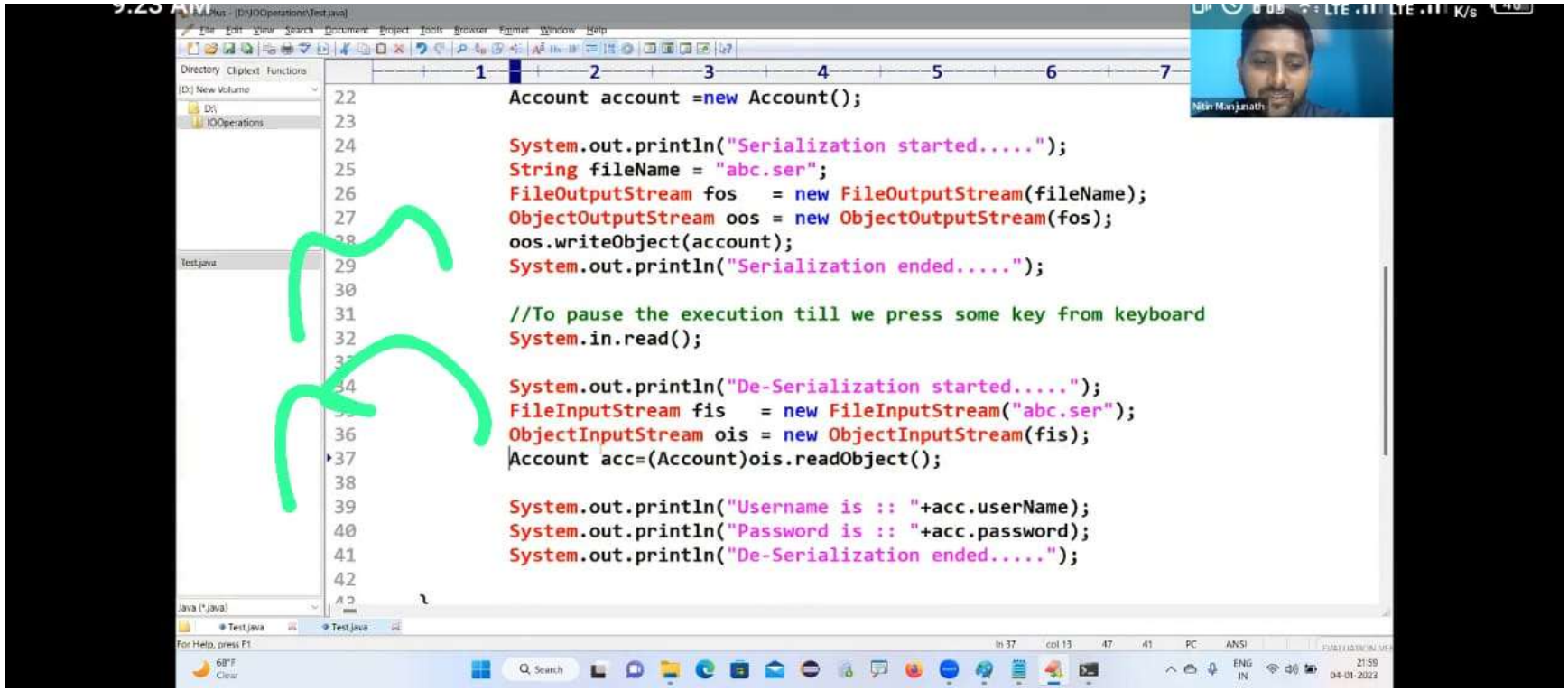

public java.io.ObjectInputStream(java.io.InputStream) throws java.io.IO
public java.io.FileInputStream(java.lang.String) throws java.io.FileNotFoundException;
public final java.lang.Object readObject() throws java.io.IOException, java.lang.ClassN
*/
class Dog implements Serializable
{
Cat c =new Cat();
}
class Cat implements Serializable
{
Rat r =new Rat();
}
class Rat implements Serializable
{
int j = 20;
}
class Test
{
public static void main(String[] args)throws Exception
{
Nitin Manjunath

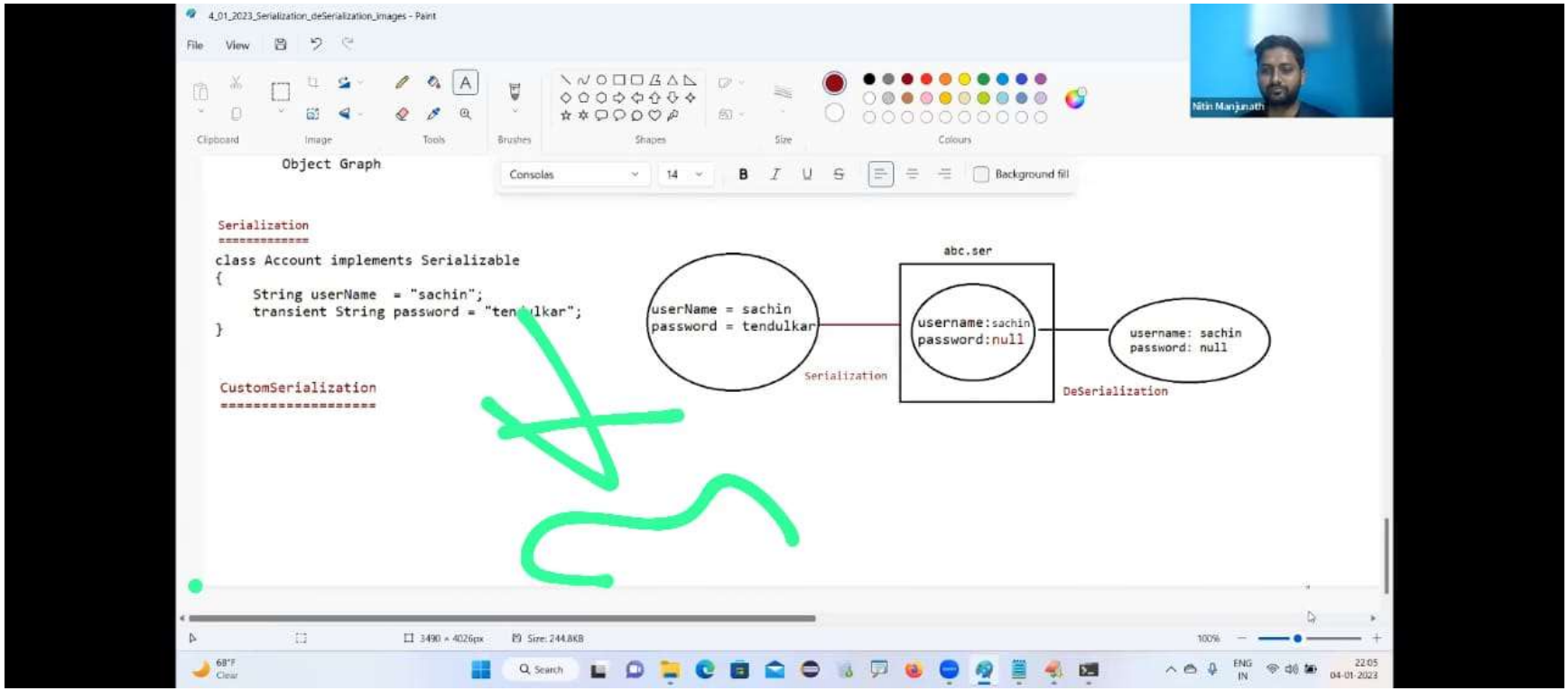

EditPlus - [D:\IOOperations\Test.java]
File Edit View Search Document Project Tools Browser Format Window Help
Directory Cliptext Functions
(D:) New Volume
D:\
IOOperations
Test.java
Nitin Manjunath
22 int j = 20;
23 }
24 class Test
25 {
26 public static void main(String[] args)throws Exception
27 {
28 Dog d = new Dog();
29
31 System.out.println("Serialization started.....");
32 String fileName = "abc.ser";
33 FileOutputStream fos = new FileOutputStream(fileName);
34 ObjectOutputStream oos = new ObjectOutputStream(fos);
35 oos.writeObject(d);
36 System.out.println("Serialization ended.....");
37
39 //To pause the execution till we press some key from keyboard
40 System.in.read();
41
42 System.out.println("De-Serialization started.....");
43 FileInputStream fis = new FileInputStream("abc.ser");
44 ObjectInputStream ois = new ObjectInputStream(fis);
45 Dog d1=(Dog)ois.readObject();
Java (*.java)
Test.java
Test.java
For Help, press F1
ln 27 col 6 51 00 PC ANSI
68°F Clear
Search
ENG IN
21:52 04-01-2023

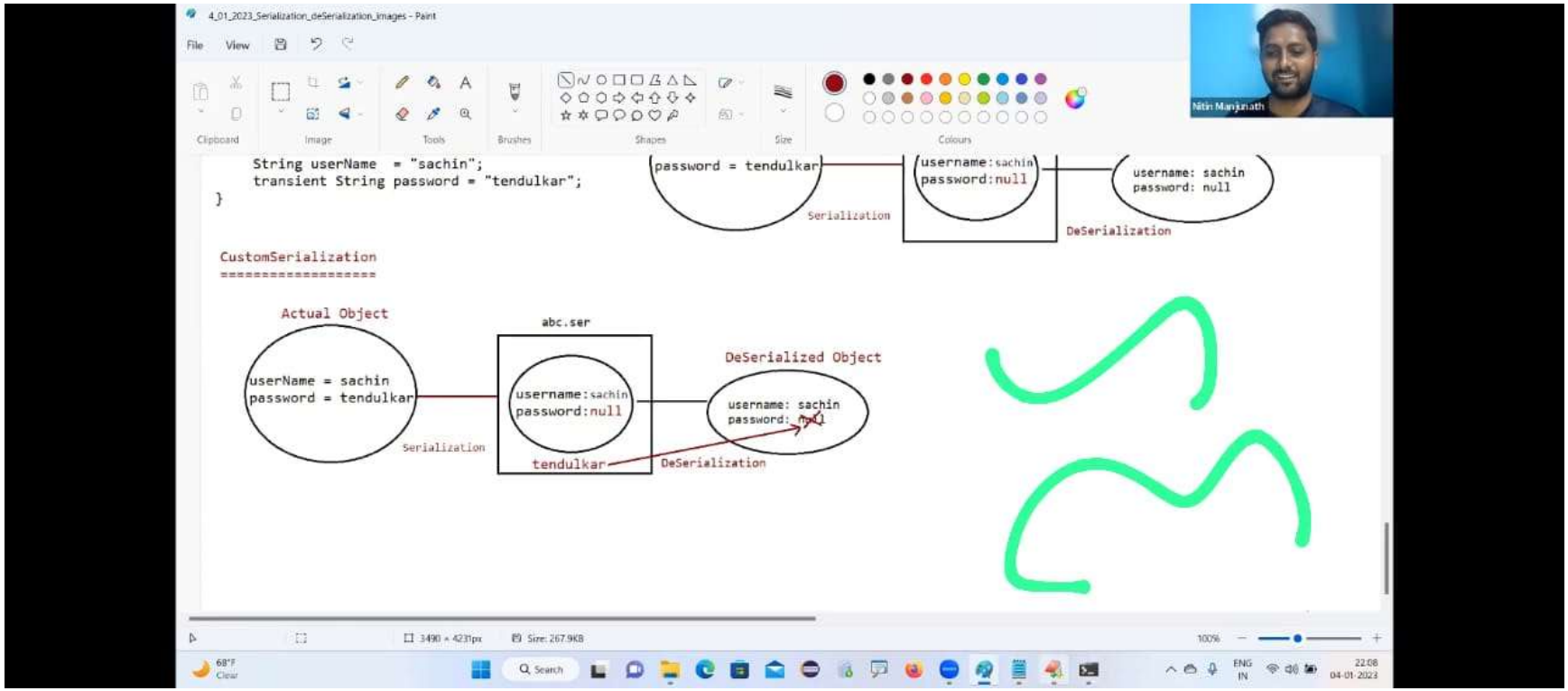

Nitin Manjunath
String fileName = "abc.ser";
FileOutputStream fos = new FileOutputStream(fileName);
ObjectOutputStream oos = new ObjectOutputStream(fos);
oos.writeObject(d);
System.out.println("Serialization ended.....");
//To pause the execution till we press some key from keyboard
System.in.read();
System.out.println("De-Serialization started.....");
FileInputStream fis = new FileInputStream("abc.ser");
ObjectInputStream ois = new ObjectInputStream(fis);
Dog d1=(Dog)ois.readObject();
System.out.println(d1.c.r.j);
System.out.println("De-Serialization ended.....");
}
//JVM shutdown now
}

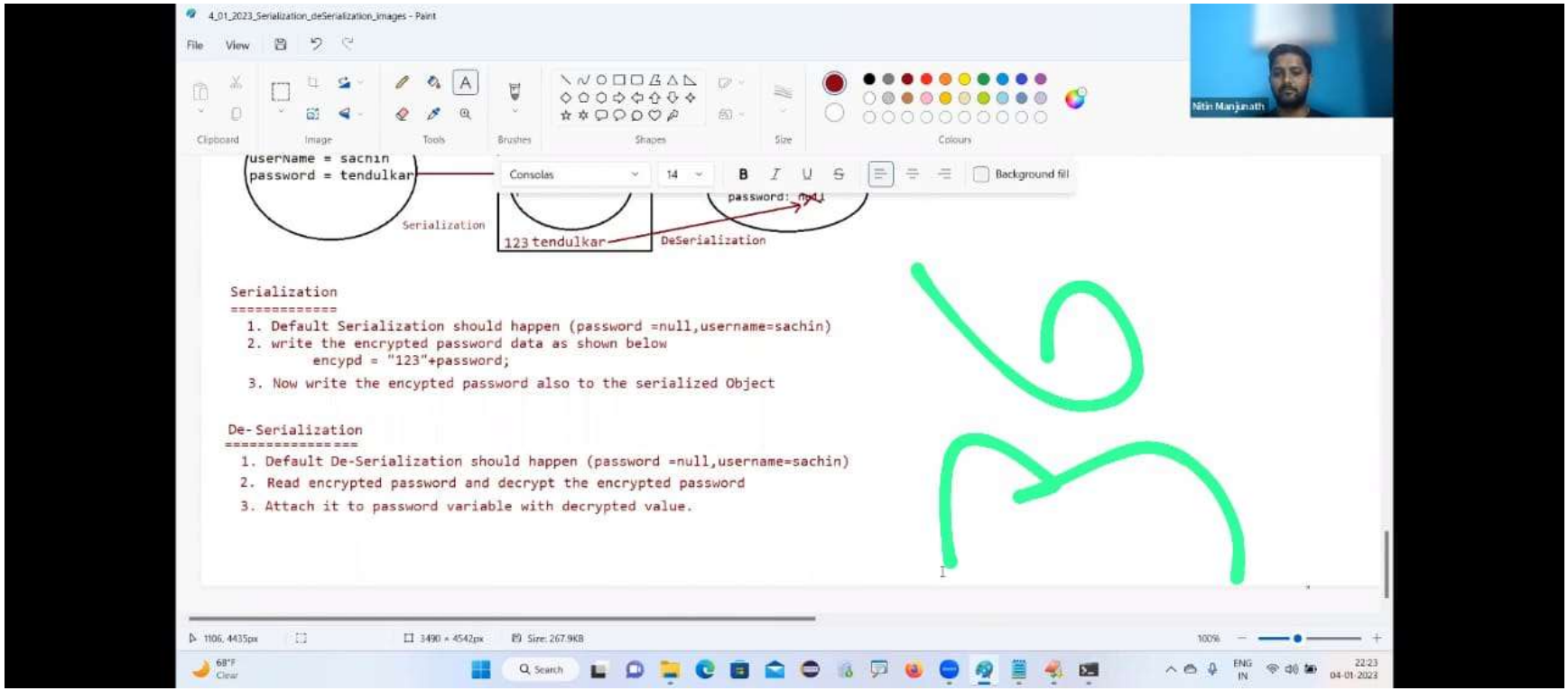

4_01_2023_Serialization_deSerialization_images - Paint
File
View
Nitin Manjunath
Consolas
14
Background fill
j=20
Rat
r
Cat
d
c
Dog (Serialiable)
Object Graph
Serialization
abc.ser
bytecodes of
Rat
Cat
Dog
De-Serialization
j=20
Rat
r
Cat
d
c

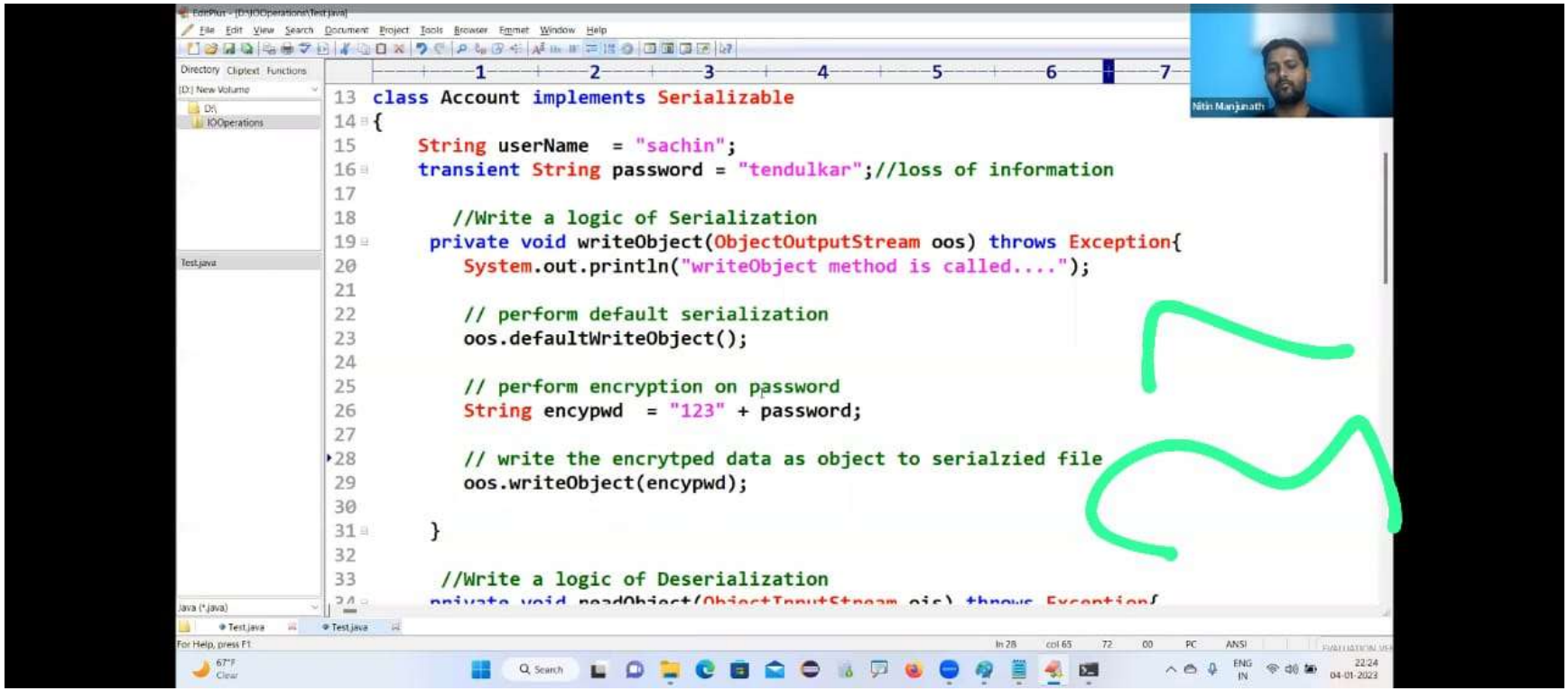

EditPlus - [D:\IOOperations\Test.java]
File Edit View Search Document Project Tools Browser Format Window Help
Directory Cliptext Functions
(D:) New Volume
D:\
IOOperations
Test.java
Java (*.java)
Nitin Manjunath
10 */
11
12 class Account implements Serializable
13 {
14 String userName = "sachin";
15 transient String password = "tendulkar";//loss of information
16 }
17
18 class Test
19 {
20 public static void main(String[] args)throws Exception
21 {
22 Account account =new Account();
23
24 System.out.println("Serialization started.....");
25 String fileName = "abc.ser";
26 FileOutputStream fos = new FileOutputStream(fileName);
27 ObjectOutputStream oos = new ObjectOutputStream(fos);
28 oos.writeObject(account);
29 System.out.println("Serialization ended.....");
30
Test.java
Test.java
For Help, press F1
ln 15 col 66 47 00 PC ANSI
68°F Clear
Search
ENG IN
21:59 04-01-2023

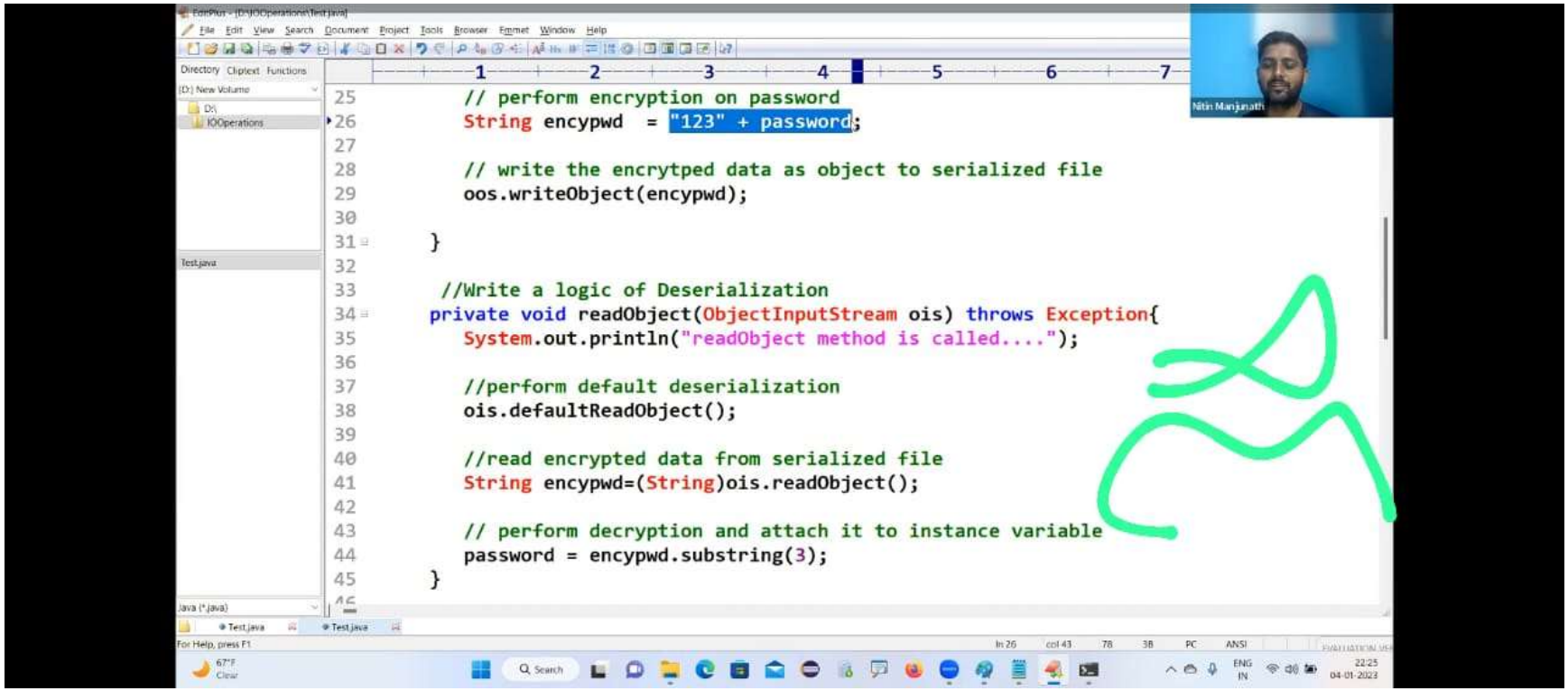

Account account =new Account();
System.out.println("Serialization started.....");
String fileName = "abc.ser";
FileOutputStream fos = new FileOutputStream(fileName);
ObjectOutputStream oos = new ObjectOutputStream(fos);
oos.writeObject(account);
System.out.println("Serialization ended.....");
//To pause the execution till we press some key from keyboard
System.in.read();
System.out.println("De-Serialization started.....");
FileInputStream fis = new FileInputStream("abc.ser");
ObjectInputStream ois = new ObjectInputStream(fis);
Account acc=(Account)ois.readObject();
System.out.println("Username is :: "+acc.userName);
System.out.println("Password is :: "+acc.password);
System.out.println("De-Serialization ended.....");
Nitin Manjunath

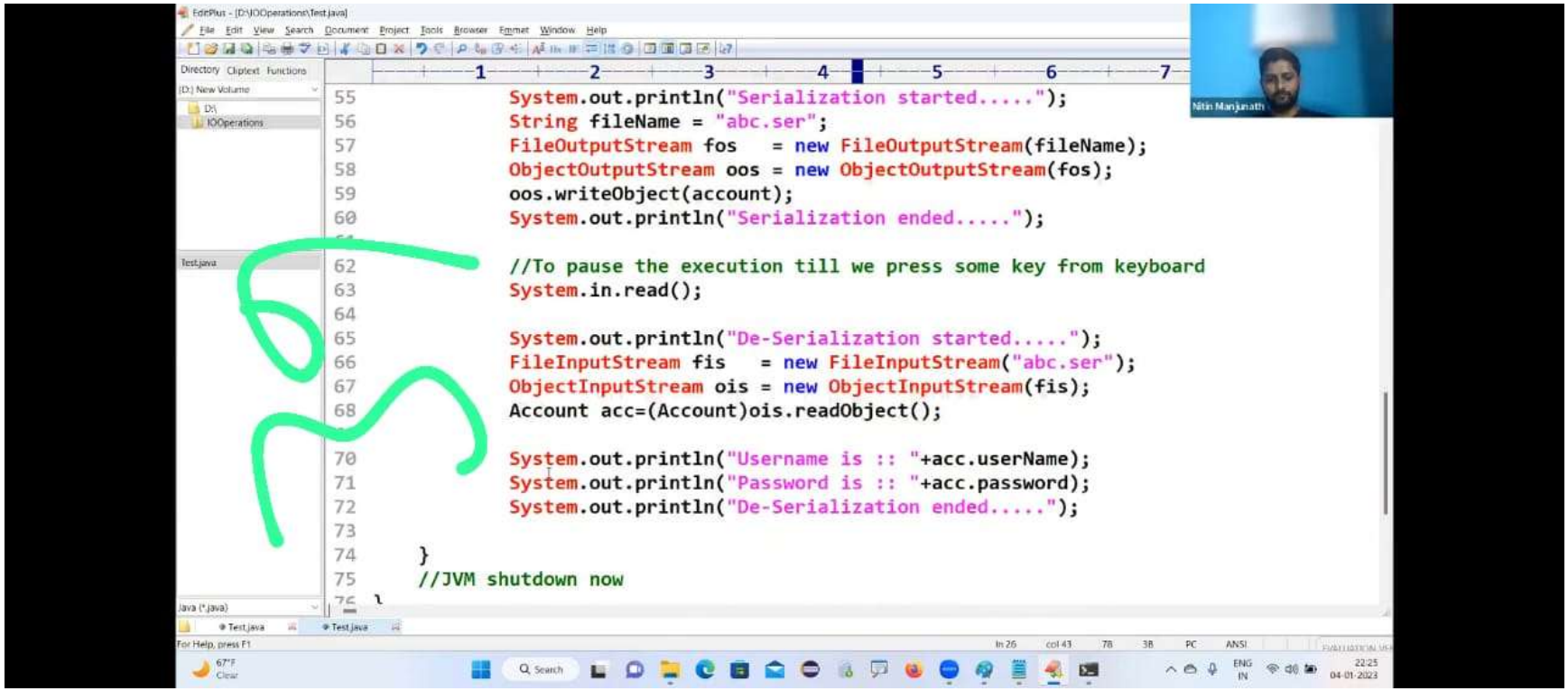

4_01_2023_Serialization_deSerialization_Images - Paint
File
View
Nitin Manjunath
Object Graph
Consolas
14
Background fill
Serialization
==========
class Account implements Serializable
{
String userName = "sachin";
transient String password = "tendulkar";
}
CustomSerialization
==========
abc.ser
userName = sachin
password = tendulkar
Serialization
username:sachin
password:null
username: sachin
password: null
DeSerialization
3490 × 4026px
Size: 244.8KB
100%
Search
ENG
IN
22:05
04-01-2023

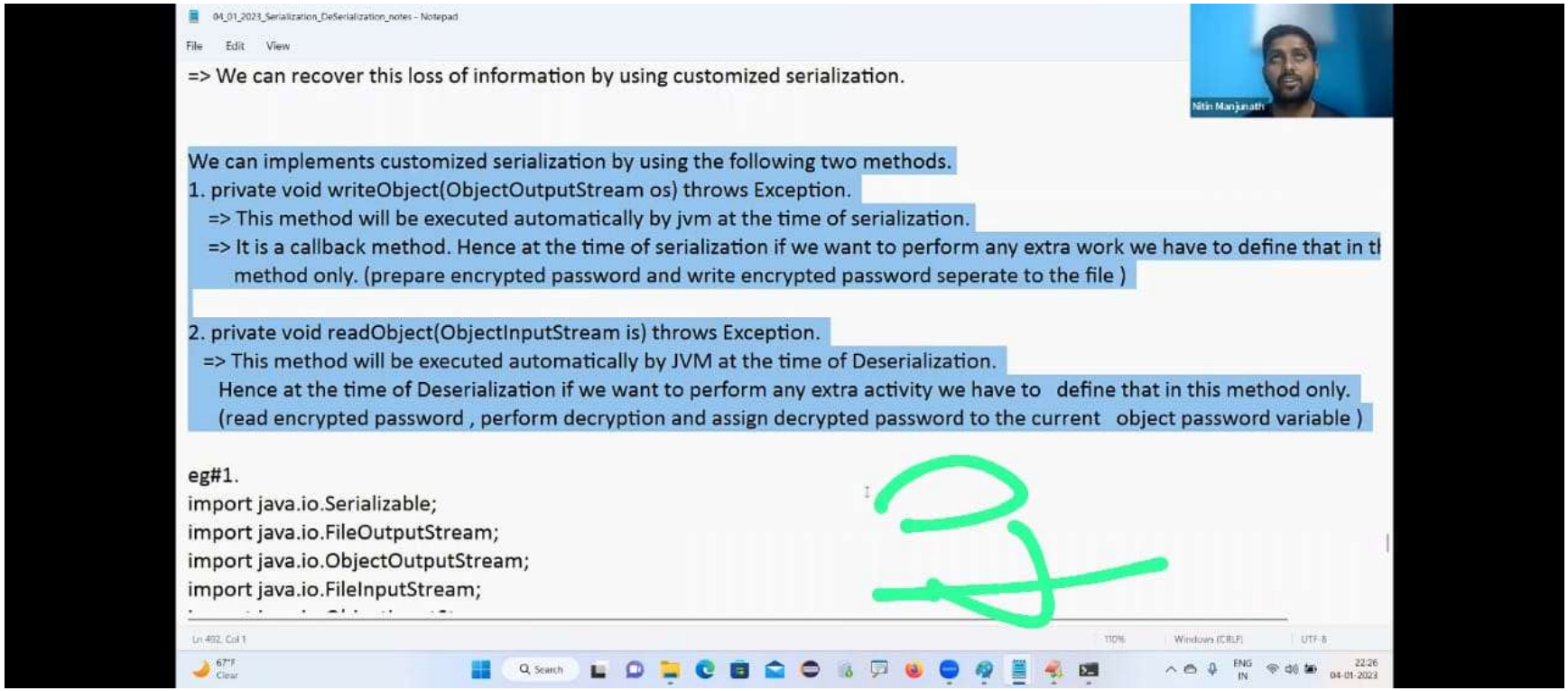

4_01_2023_Serialization_deSerialization_Images - Paint
File
View
Nitin Manjunath
String userName = "sachin";
transient String password = "tendulkar";
}
password = tendulkar
Serialization
username:sachin
password:null
DeSerialization
username: sachin
password: null
CustomSerialization
Actual Object
userName = sachin
password = tendulkar
Serialization
abc.ser
username:sachin
password:null
tendulkar
DeSerialization
DeSerialized Object
username: sachin
password: null
3490 × 4231px
Size: 267.9KB
100%
68°F
Clear
Search
ENG
IN
22:08
04-01-2023

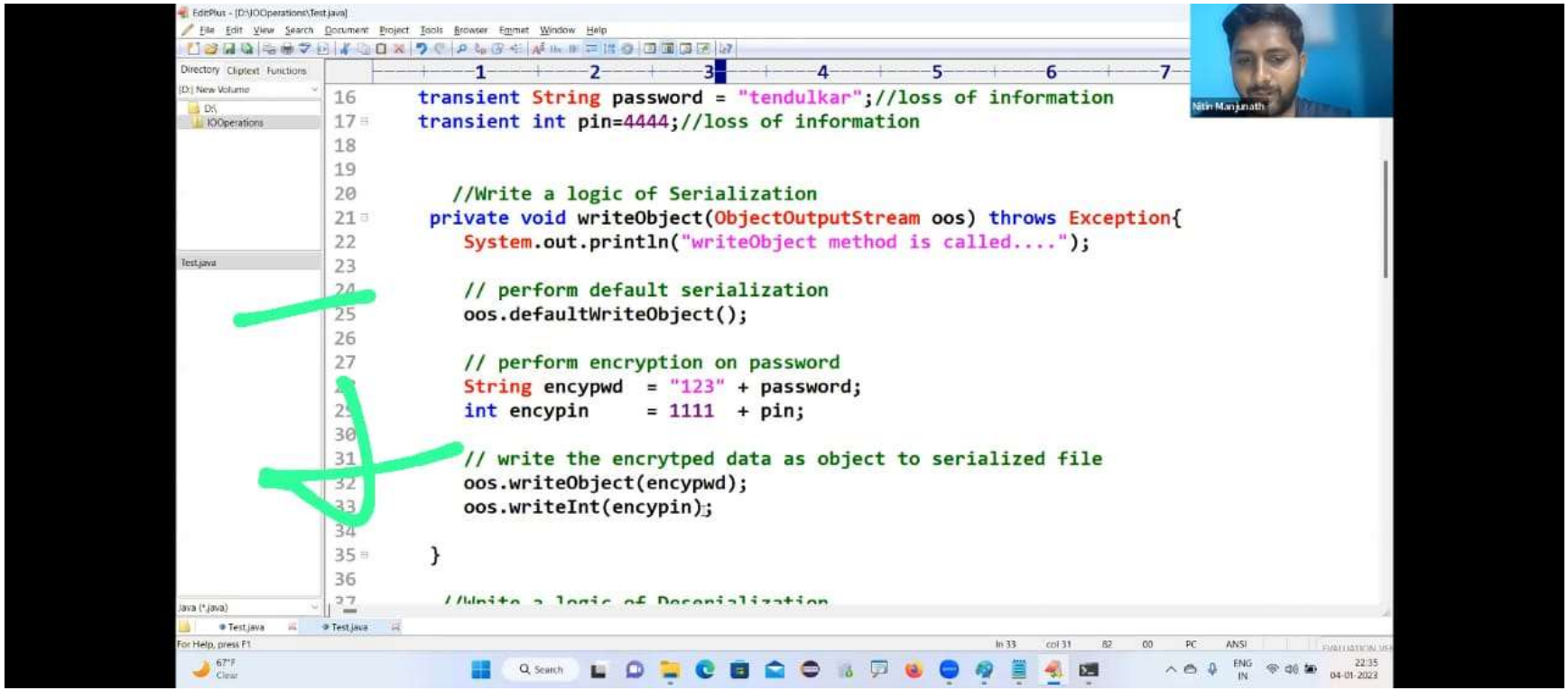

4_01_2023_Serialization_deSerialization_images - Paint
File
View
Nitin Manjunath
userName = sachin
password = tendulkar
Consolas
14
Background fill
Serialization
123 tendulkar
password: null
DeSerialization
Serialization
1. Default Serialization should happen (password =null,username=sachin)
2. write the encrypted password data as shown below
encypd = "123"+password;
3. Now write the encypted password also to the serialized Object
De-Serialization
1. Default De-Serialization should happen (password =null,username=sachin)
2. Read encrypted password and decrypt the encrypted password
3. Attach it to password variable with decrypted value.

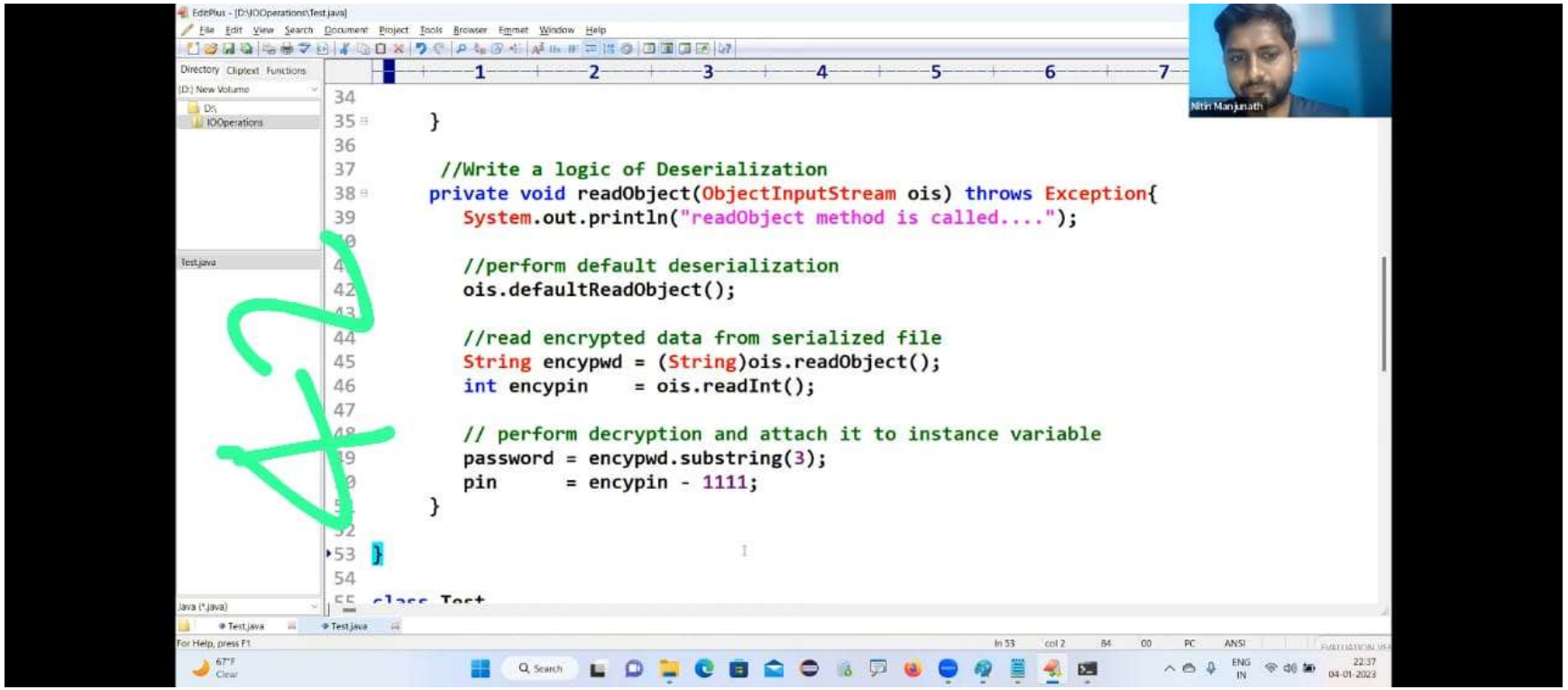

Nitin Manjunath
13 class Account implements Serializable
14 {
15 String userName = "sachin";
16 transient String password = "tendulkar";//loss of information
17
18 //Write a logic of Serialization
19 private void writeObject(ObjectOutputStream oos) throws Exception{
20 System.out.println("writeObject method is called....");
21
22 // perform default serialization
23 oos.defaultWriteObject();
24
25 // perform encryption on password
26 String encypwd = "123" + password;
27
28 // write the encrytped data as object to serialzied file
29 oos.writeObject(encypwd);
30
31 }
32
33 //Write a logic of Deserialization

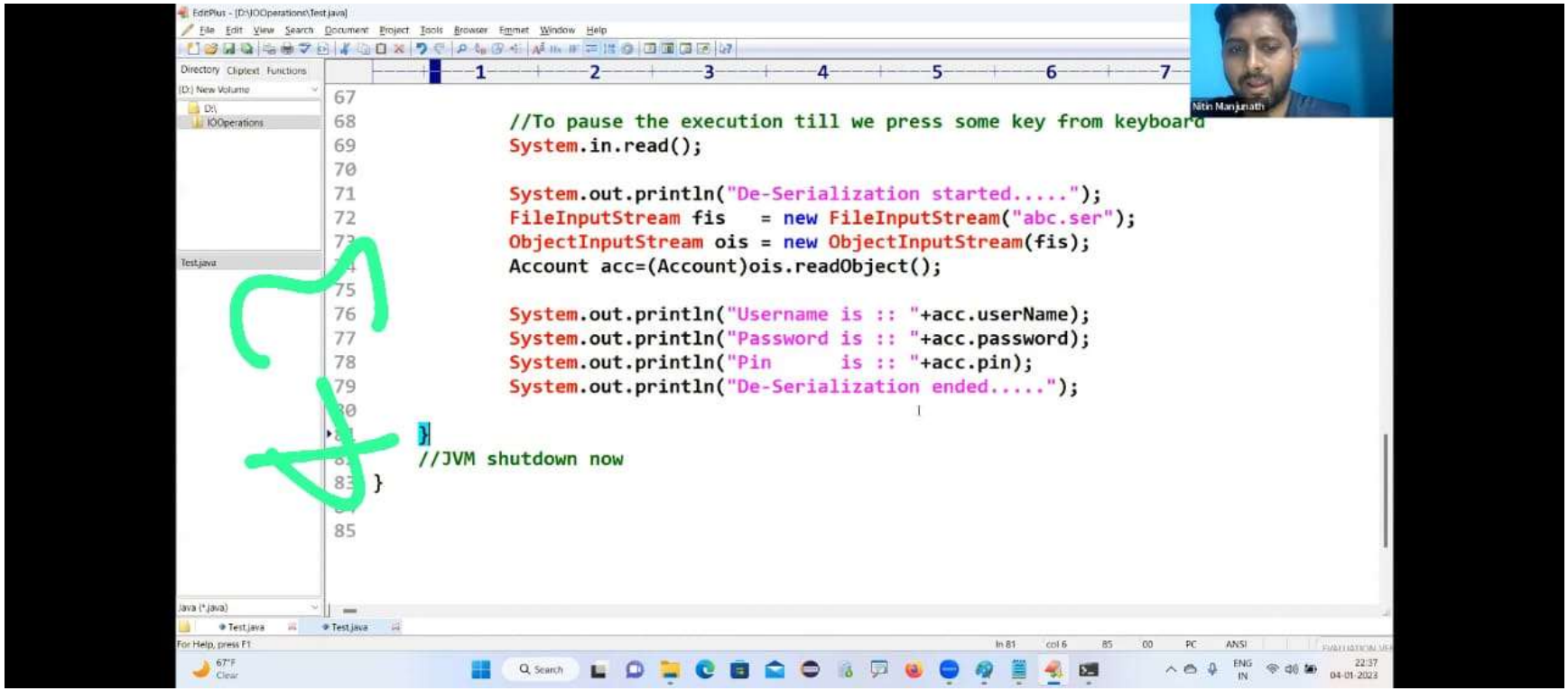

Nitin Manjunath
// perform encryption on password
String encypwd = "123" + password;
// write the encrytped data as object to serialized file
oos.writeObject(encypwd);
}
//Write a logic of Deserialization
private void readObject(ObjectInputStream ois) throws Exception{
System.out.println("readObject method is called....");
//perform default deserialization
ois.defaultReadObject();
//read encrypted data from serialized file
String encypwd=(String)ois.readObject();
// perform decryption and attach it to instance variable
password = encypwd.substring(3);
}

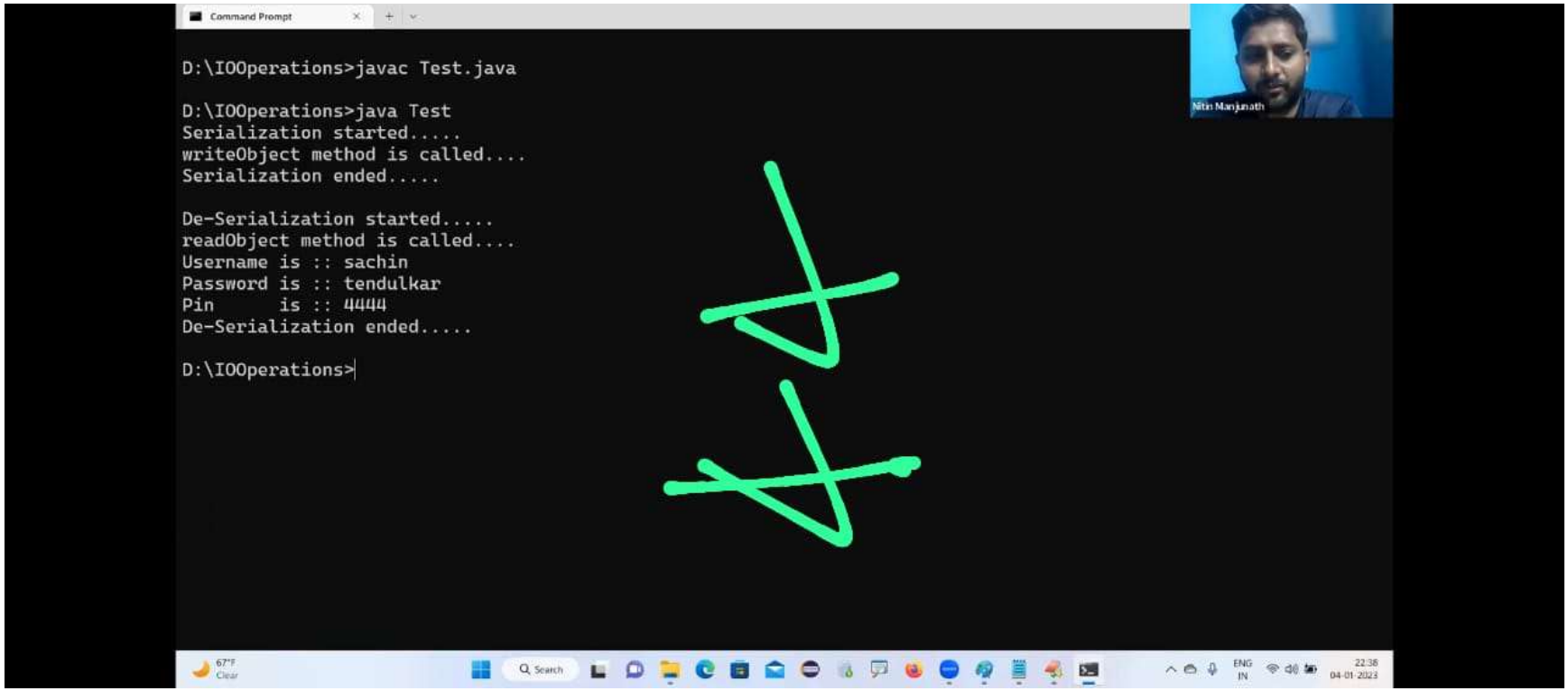

System.out.println("Serialization started.....");
String fileName = "abc.ser";
FileOutputStream fos = new FileOutputStream(fileName);
ObjectOutputStream oos = new ObjectOutputStream(fos);
oos.writeObject(account);
System.out.println("Serialization ended.....");
//To pause the execution till we press some key from keyboard
System.in.read();
System.out.println("De-Serialization started.....");
FileInputStream fis = new FileInputStream("abc.ser");
ObjectInputStream ois = new ObjectInputStream(fis);
Account acc=(Account)ois.readObject();
System.out.println("Username is :: "+acc.userName);
System.out.println("Password is :: "+acc.password);
System.out.println("De-Serialization ended.....");
}
//JVM shutdown now
Nitin Manjunath


04_01_2023_Serialization_DeSerialization_notes - Notepad
File Edit View
=> We can recover this loss of information by using customized serialization.
Nitin Manjunath
We can implements customized serialization by using the following two methods.
1. private void writeObject(ObjectOutputStream os) throws Exception.
=> This method will be executed automatically by jvm at the time of serialization.
=> It is a callback method. Hence at the time of serialization if we want to perform any extra work we have to define that in tl
method only. (prepare encrypted password and write encrypted password seperate to the file)
2. private void readObject(ObjectInputStream is) throws Exception.
=> This method will be executed automatically by JVM at the time of Deserialization.
Hence at the time of Deserialization if we want to perform any extra activity we have to define that in this method only.
(read encrypted password , perform decryption and assign decrypted password to the current object password variable)
eg#1.
import java.io.Serializable;
import java.io.FileOutputStream;
import java.io.ObjectOutputStream;
import java.io.FileInputStream;
Ln 492, Col 1
110%
Windows (CRLF)
UTF-8
67°F
Clear
Search
ENG
IN
22:26
04-01-2023


EditPlus - [D:\IOOperations\Test.java]
File Edit View Search Document Project Tools Browser Format Window Help
Directory Cliptext Functions
D:\
IOOperations
Test.java
Nitin Manjunath
transient String password = "tendulkar";//loss of information
transient int pin=4444;//loss of information
//Write a logic of Serialization
private void writeObject(ObjectOutputStream oos) throws Exception{
System.out.println("writeObject method is called....");
// perform default serialization
oos.defaultWriteObject();
// perform encryption on password
String encypwd = "123" + password;
int encypin = 1111 + pin;
// write the encrytped data as object to serialized file
oos.writeObject(encypwd);
oos.writeInt(encypin);
}
Test.java
Test.java
For Help, press F1
ln 33 col 31 82 00 PC ANSI
67°F Clear
Search
ENG IN
22:35 04-01-2023


EditPlus - [D:\IOOperations\Test.java]
File Edit View Search Document Project Tools Browser Frmmet Window Help
Directory Cliptext Functions
(D:) New Volume
D:\
IOOperations
Test.java
Java (*.java)
Test.java
Test.java
For Help, press F1
Nitin Manjunath
34
35 }
36
37 //Write a logic of Deserialization
38 private void readObject(ObjectInputStream ois) throws Exception{
39 System.out.println("readObject method is called....");
40
41 //perform default deserialization
42 ois.defaultReadObject();
43
44 //read encrypted data from serialized file
45 String encypwd = (String)ois.readObject();
46 int encypin = ois.readInt();
47
48 // perform decryption and attach it to instance variable
49 password = encypwd.substring(3);
50 pin = encypin - 1111;
51 }
52
53 }
54
55 class Test
In 53 col 2 84 00 PC ANSI
67°F Clear
Search
ENG IN
22:37 04-01-2023


EditPlus - [D:\IOOperations\Test.java]
File Edit View Search Document Project Tools Browser Frmmet Window Help
Directory Cliptext Functions
(D:) New Volume
D:\
IOOperations
Test.java
Java (*.java)
Test.java
Test.java
For Help, press F1
Nitin Manjunath
67
68 //To pause the execution till we press some key from keyboard
69 System.in.read();
70
71 System.out.println("De-Serialization started.....");
72 FileInputStream fis = new FileInputStream("abc.ser");
73 ObjectInputStream ois = new ObjectInputStream(fis);
74 Account acc=(Account)ois.readObject();
75
76 System.out.println("Username is :: "+acc.userName);
77 System.out.println("Password is :: "+acc.password);
78 System.out.println("Pin is :: "+acc.pin);
79 System.out.println("De-Serialization ended.....");
80
81 }
82 //JVM shutdown now
83 }
84
85
In 81 col 6 85 00 PC ANSI
67°F Clear
Search
ENG IN
22:37 04-01-2023


Command Prompt
D:\IOOperations>javac Test.java
D:\IOOperations>java Test
Serialization started.....
writeObject method is called....
Serialization ended.....
De-Serialization started.....
readObject method is called....
Username is :: sachin
Password is :: tendulkar
Pin is :: 4444
De-Serialization ended.....
D:\IOOperations>
Nitin Manjunath
67°F
Clear
Search
ENG
IN
22:38
04-01-2023