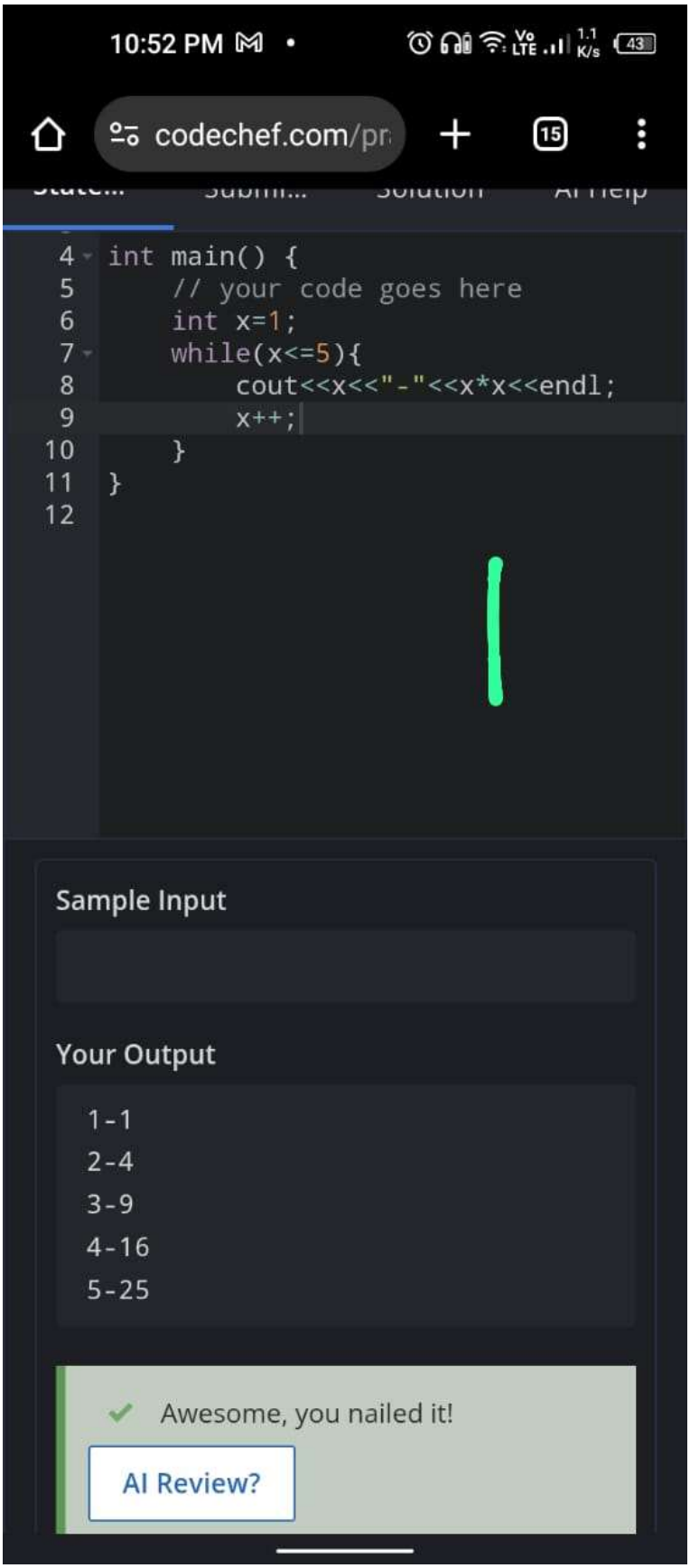
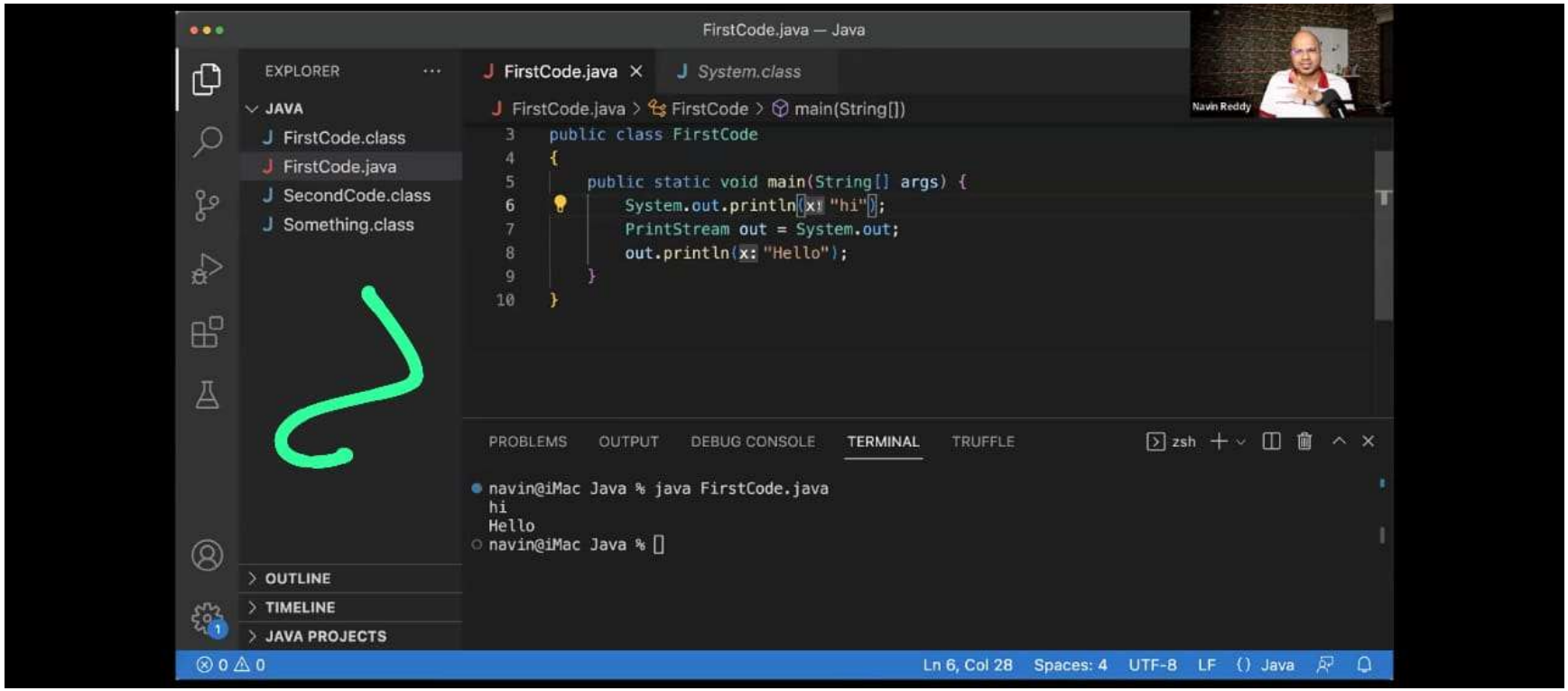
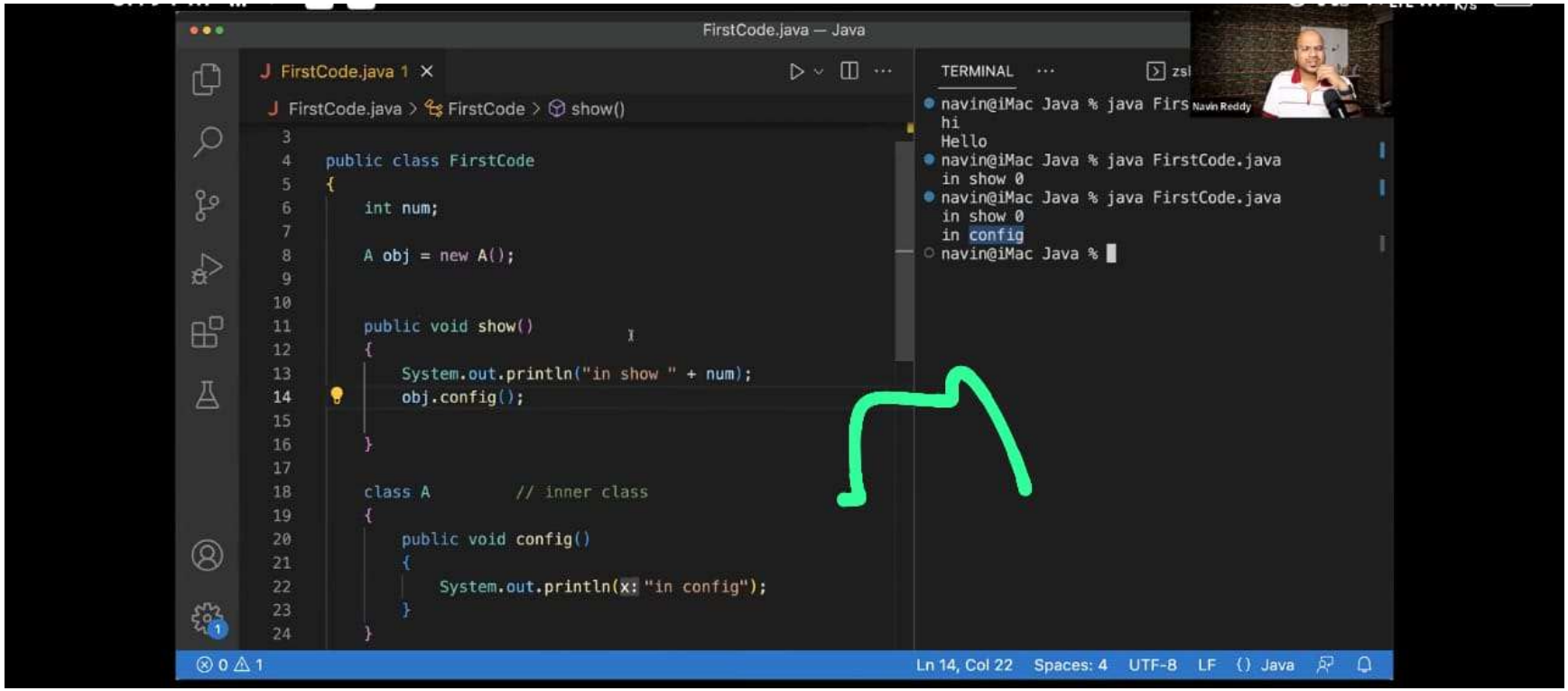
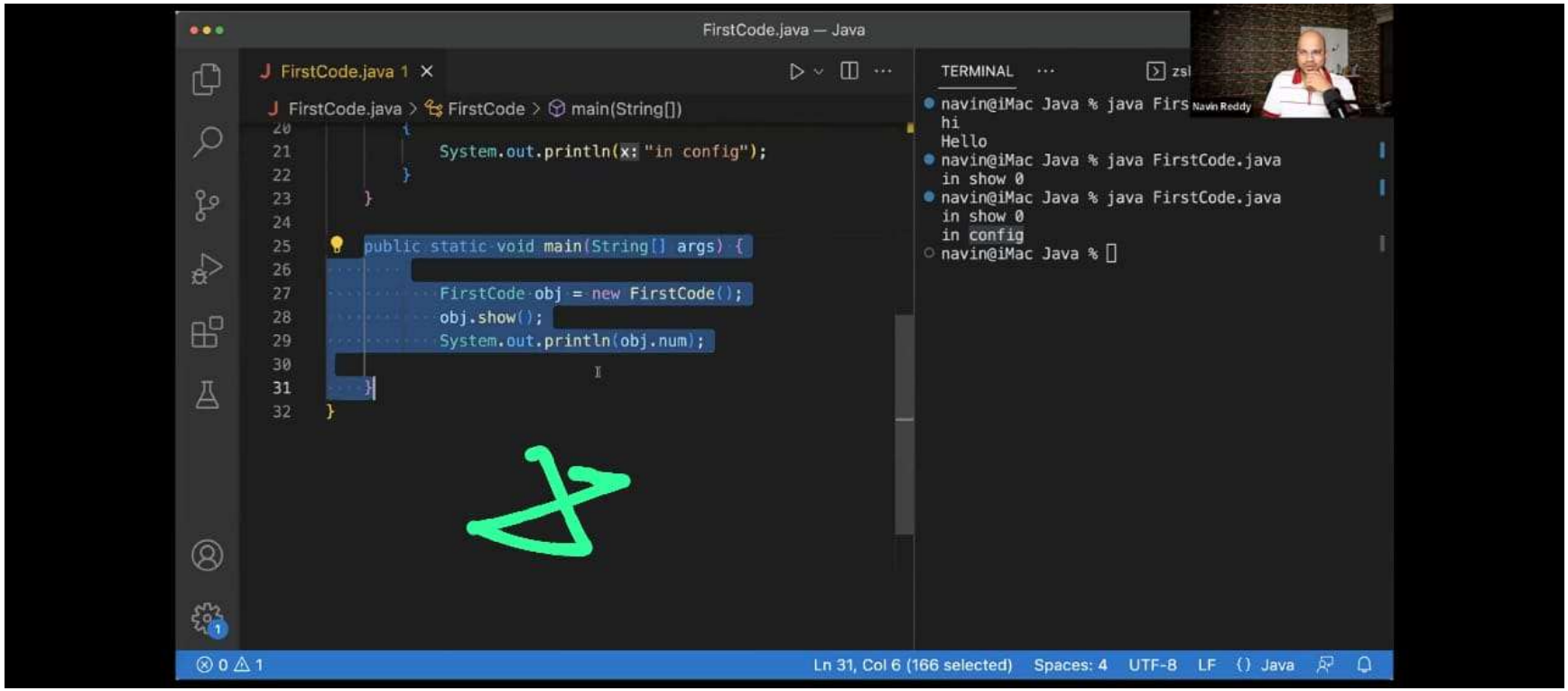
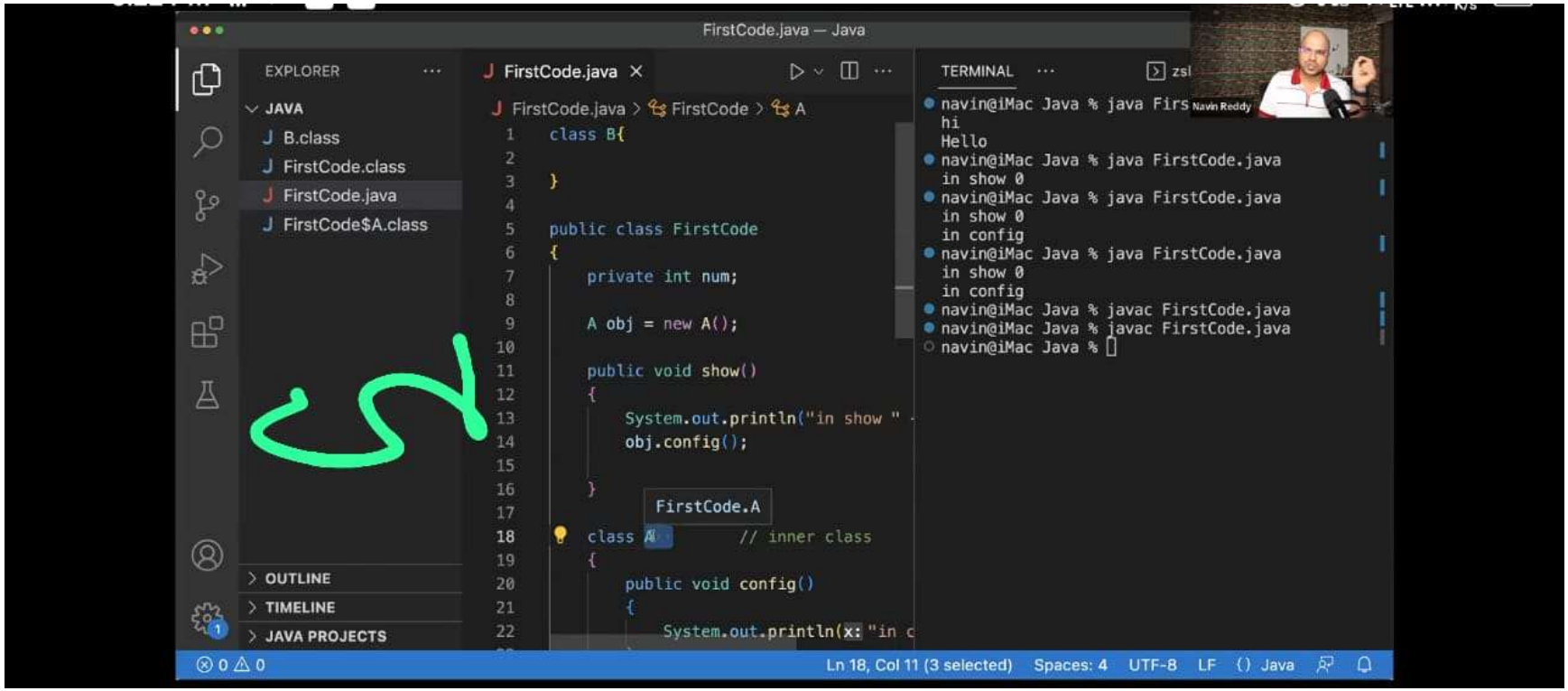
Java Inner Classes And Interfaces

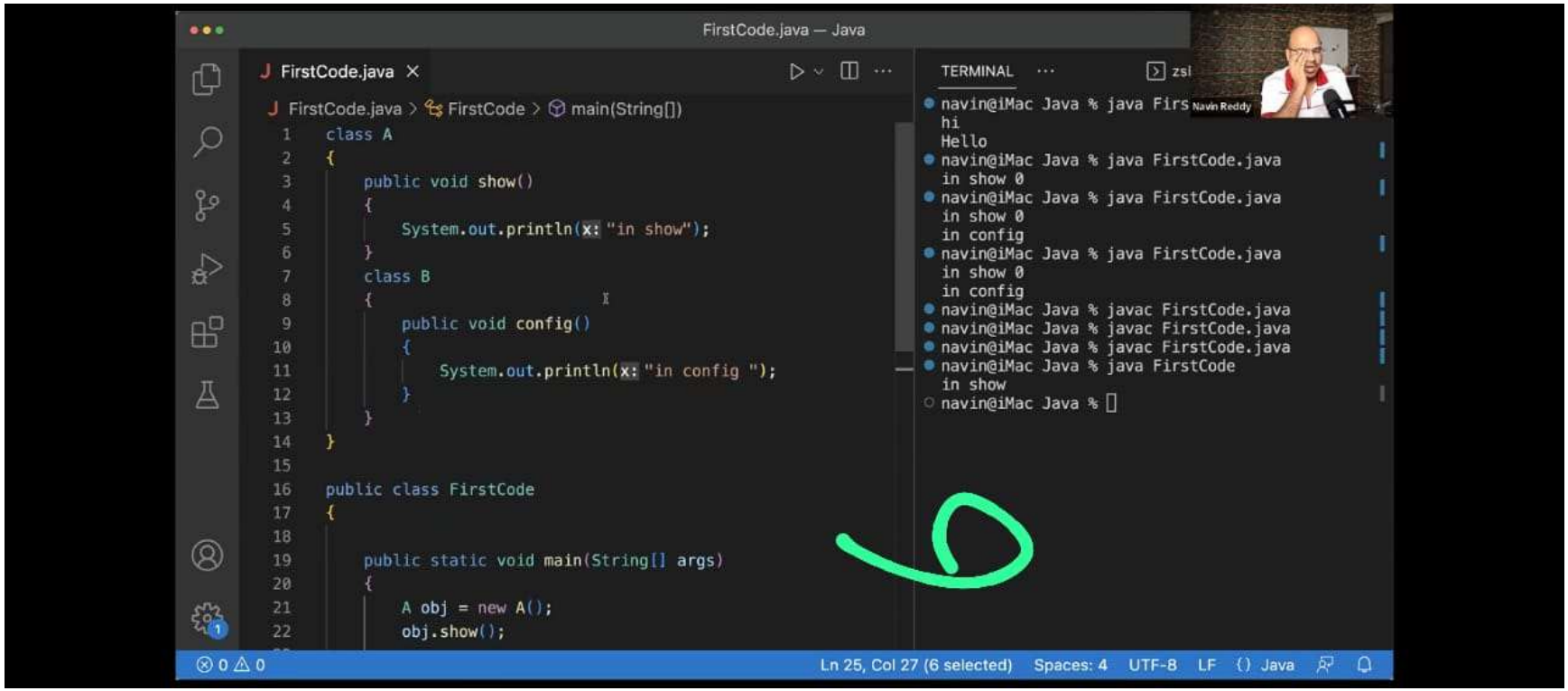


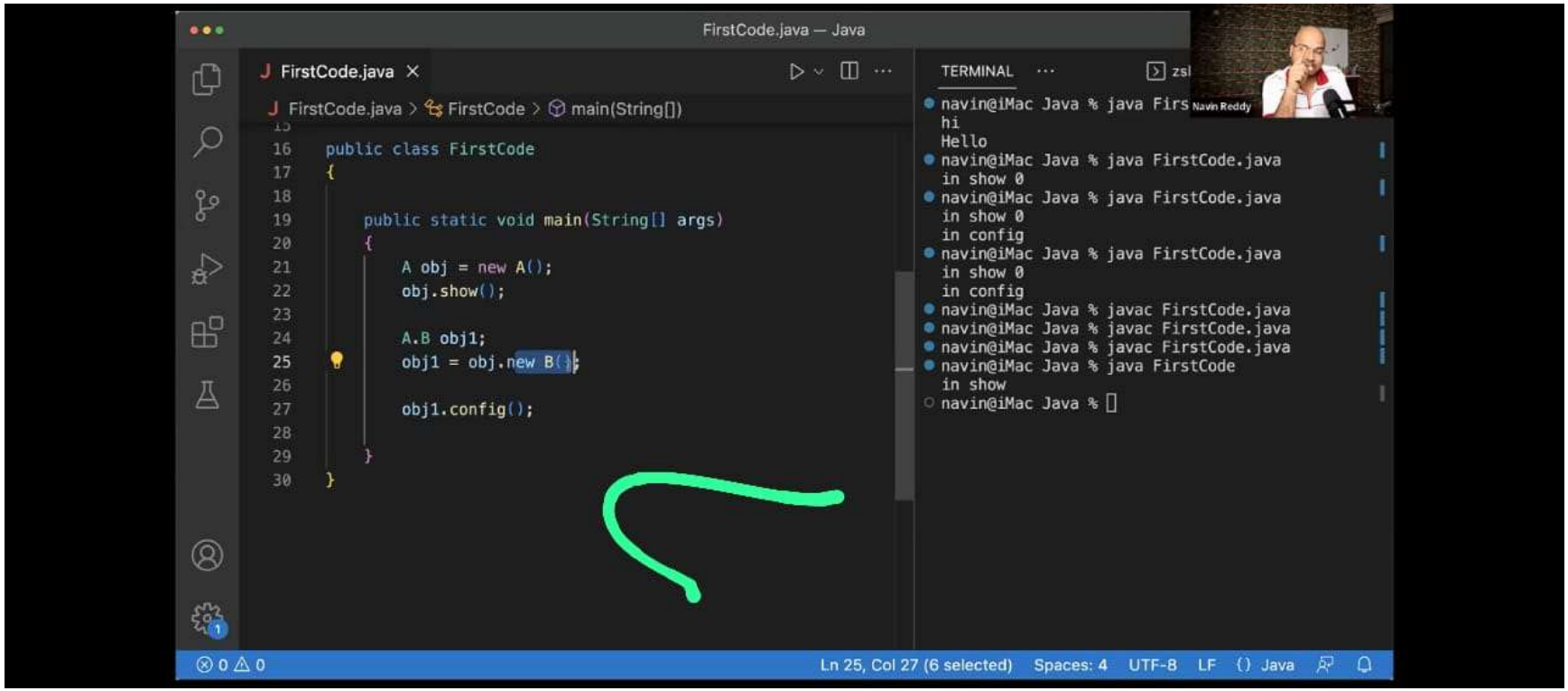

FirstCode.java — Java
EXPLORER
JAVA
FirstCode.class
FirstCode.java
SecondCode.class
Something.class
OUTLINE
TIMELINE
JAVA PROJECTS
FirstCode.java
System.class
FirstCode.java > FirstCode > main(String[])
3 public class FirstCode
4 {
5 public static void main(String[] args) {
6 System.out.println(x: "hi");
7 PrintStream out = System.out;
8 out.println(x: "Hello");
9 }
10 }
Navin Reddy
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
TRUFFLE
zsh
navin@iMac Java % java FirstCode.java
hi
Hello
navin@iMac Java %
0 0
Ln 6, Col 28
Spaces: 4
UTF-8
LF
Java

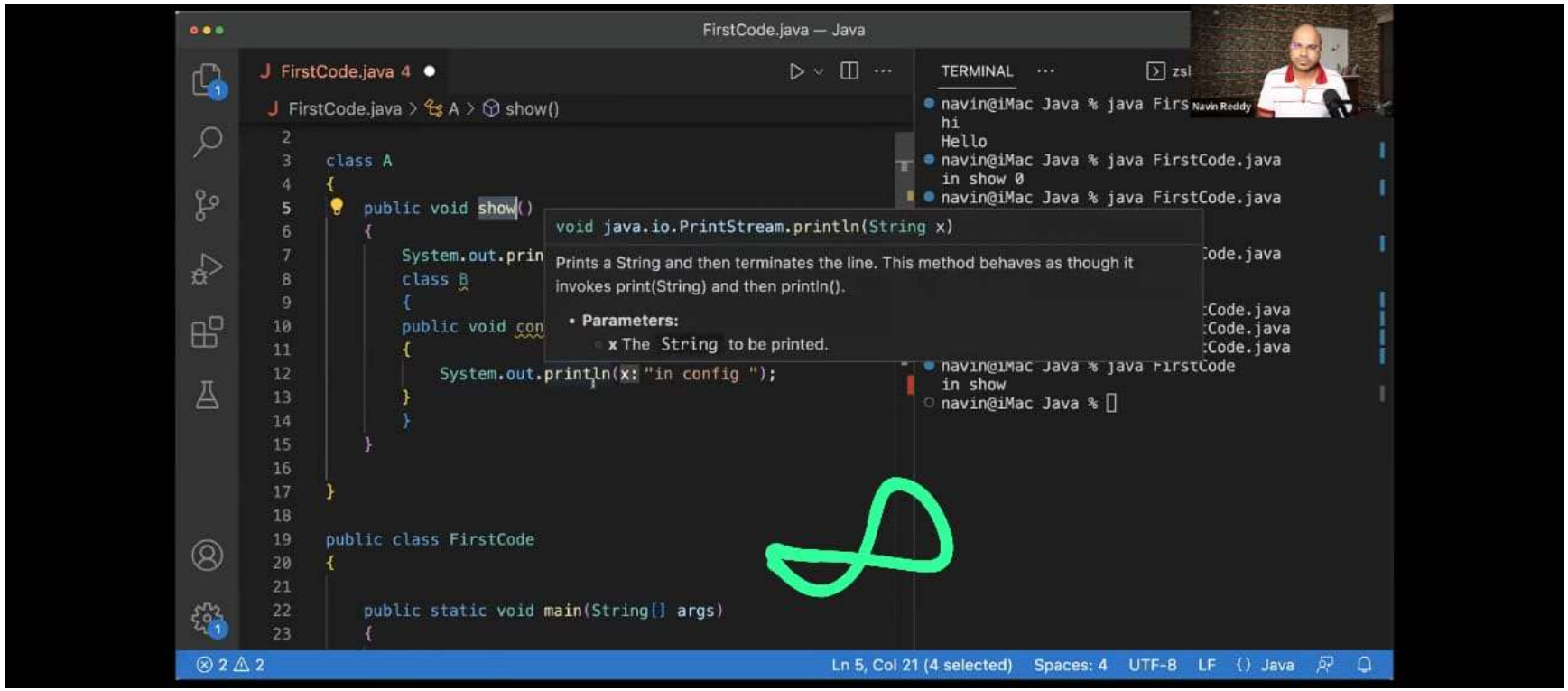

FirstCode.java — Java
FirstCode.java 1
FirstCode.java > FirstCode > show()
3
4 public class FirstCode
5 {
6 int num;
7
8 A obj = new A();
9
10
11 public void show()
12 {
13 System.out.println("in show " + num);
14 obj.config();
15
16 }
17
18 class A // inner class
19 {
20 public void config()
21 {
22 System.out.println(x: "in config");
23 }
24 }
TERMINAL
navin@iMac Java % java Firs
hi
Hello
navin@iMac Java % java FirstCode.java
in show 0
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java %
Navin Reddy
⊗ 0 ⚠ 1
Ln 14, Col 22
Spaces: 4
UTF-8
LF
{} Java

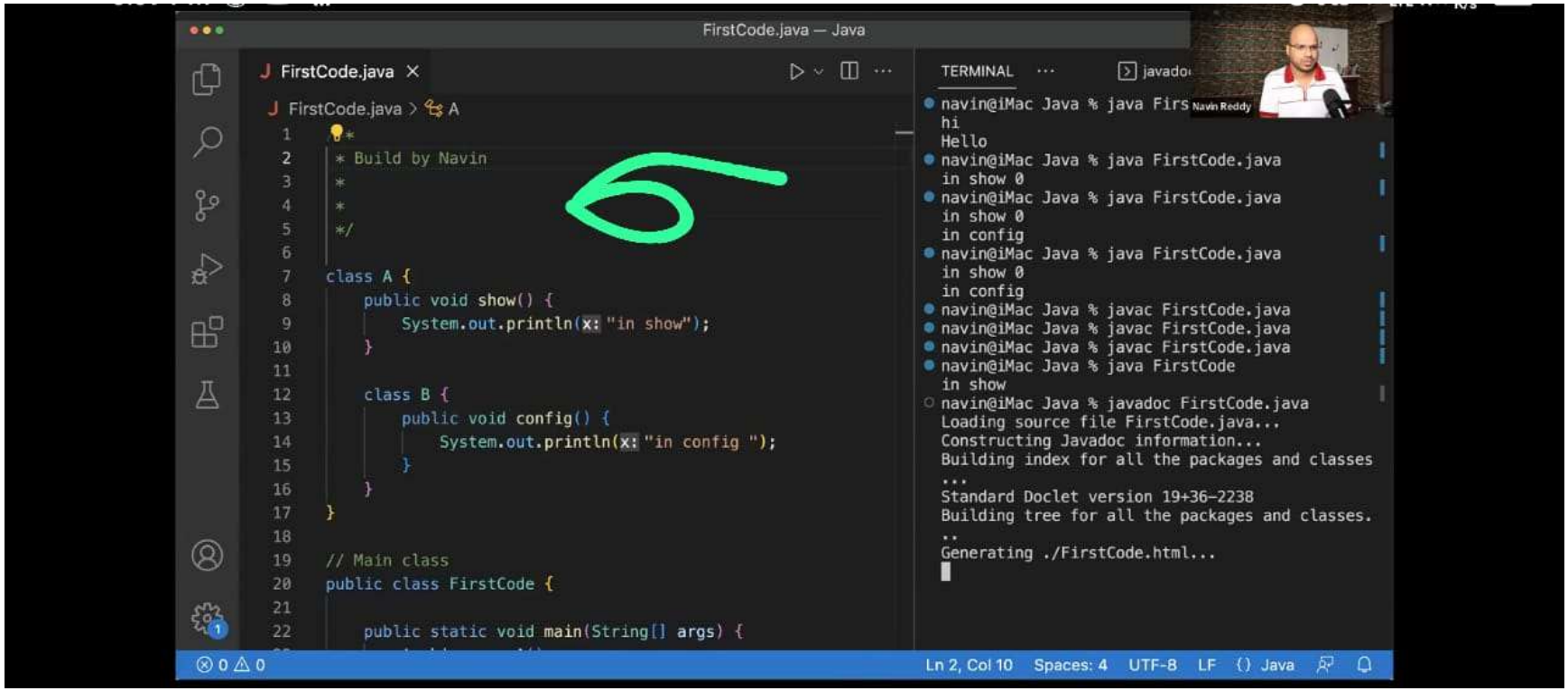

FirstCode.java — Java
FirstCode.java 1
FirstCode.java > FirstCode > main(String[])
System.out.println(x: "in config");
public static void main(String[] args) {
FirstCode obj = new FirstCode();
obj.show();
System.out.println(obj.num);
TERMINAL
navin@iMac Java % java Firs
hi
Hello
navin@iMac Java % java FirstCode.java
in show 0
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java %
Navin Reddy
Ln 31, Col 6 (166 selected)
Spaces: 4
UTF-8
LF
Java

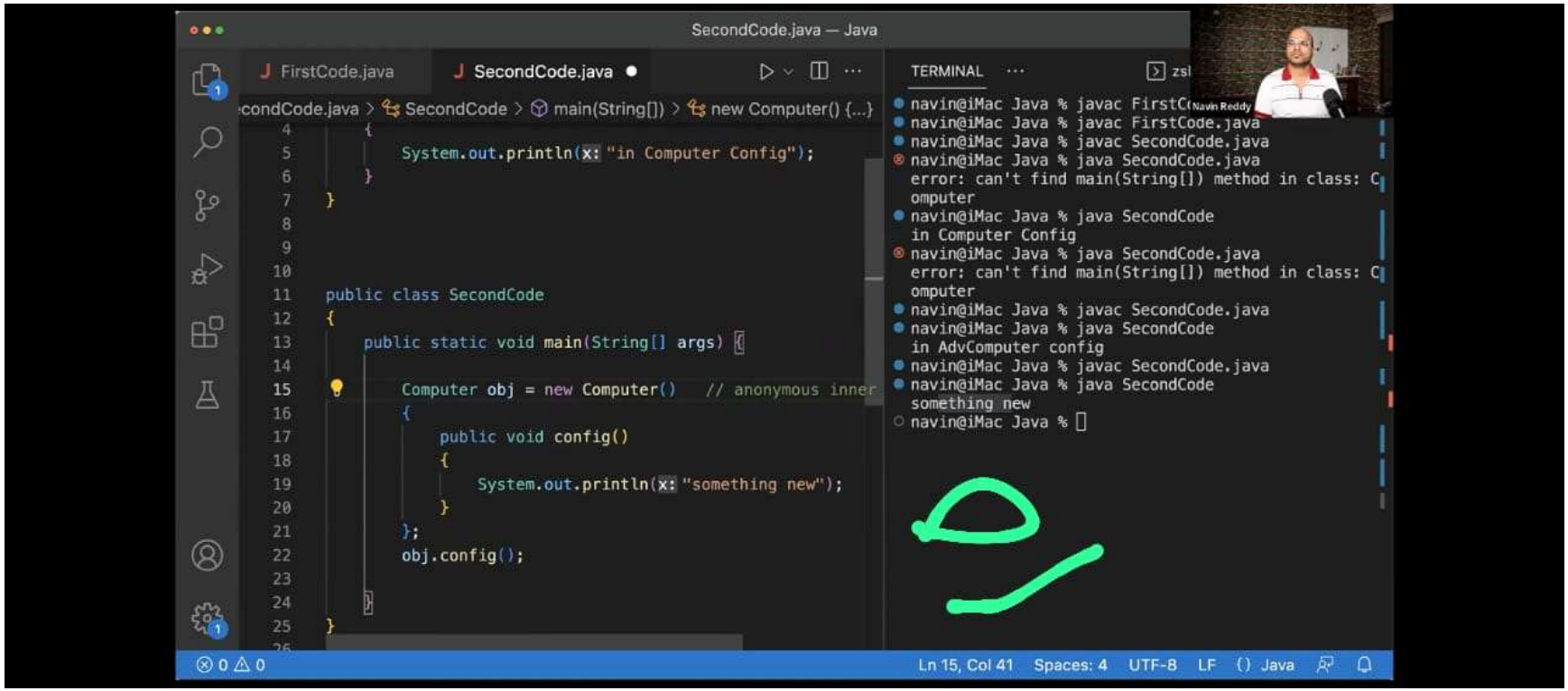

FirstCode.java — Java
EXPLORER
JAVA
B.class
FirstCode.class
FirstCode.java
FirstCode$A.class
OUTLINE
TIMELINE
JAVA PROJECTS
FirstCode.java
FirstCode.java > FirstCode > A
class B{
}
public class FirstCode
{
private int num;
A obj = new A();
public void show()
{
System.out.println("in show "
obj.config();
}
FirstCode.A
class A // inner class
{
public void config()
{
System.out.println(x: "in c
TERMINAL
zsh
navin@iMac Java % java Firs
hi
Hello
navin@iMac Java % java FirstCode.java
in show 0
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java % javac FirstCode.java
navin@iMac Java % javac FirstCode.java
navin@iMac Java %
Navin Reddy
0 0
Ln 18, Col 11 (3 selected)
Spaces: 4
UTF-8
LF
Java

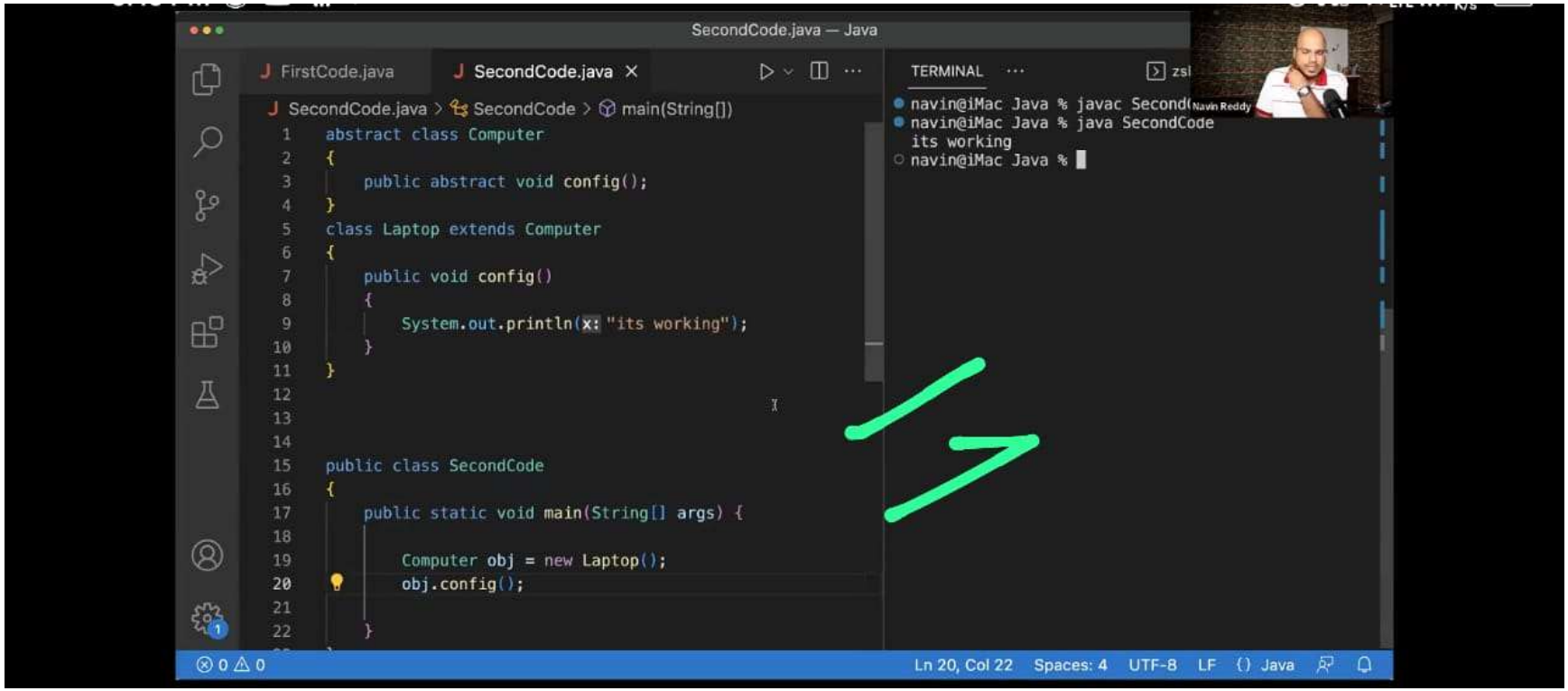

FirstCode.java — Java
FirstCode.java
FirstCode.java > FirstCode > main(String[])
class A
{
public void show()
{
System.out.println(x: "in show");
}
class B
{
public void config()
{
System.out.println(x: "in config ");
}
}
}
public class FirstCode
{
public static void main(String[] args)
{
A obj = new A();
obj.show();
TERMINAL
zsi
navin@iMac Java % java Firs
hi
Hello
navin@iMac Java % java FirstCode.java
in show 0
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java % javac FirstCode.java
navin@iMac Java % javac FirstCode.java
navin@iMac Java % javac FirstCode.java
navin@iMac Java % java FirstCode
in show
navin@iMac Java %
Navin Reddy
0 0
Ln 25, Col 27 (6 selected)
Spaces: 4
UTF-8
LF
Java

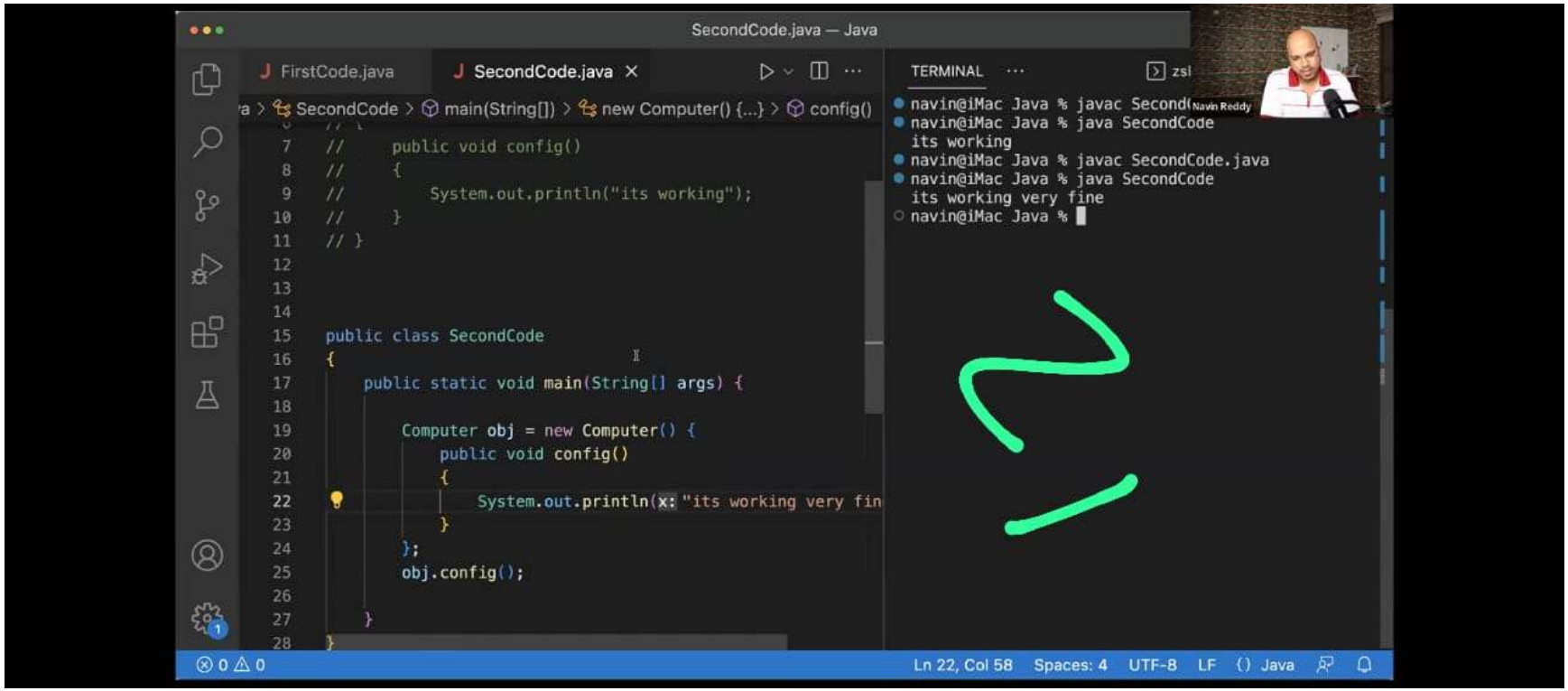

FirstCode.java — Java
FirstCode.java
FirstCode.java > FirstCode > main(String[])
public class FirstCode
{
public static void main(String[] args)
{
A obj = new A();
obj.show();
A.B obj1;
obj1 = obj.new B();
obj1.config();
}
}
TERMINAL
navin@iMac Java % java Firs
hi
Hello
navin@iMac Java % java FirstCode.java
in show 0
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java % javac FirstCode.java
navin@iMac Java % javac FirstCode.java
navin@iMac Java % javac FirstCode.java
navin@iMac Java % java FirstCode
in show
navin@iMac Java %
Navin Reddy
Ln 25, Col 27 (6 selected)
Spaces: 4
UTF-8
LF
Java

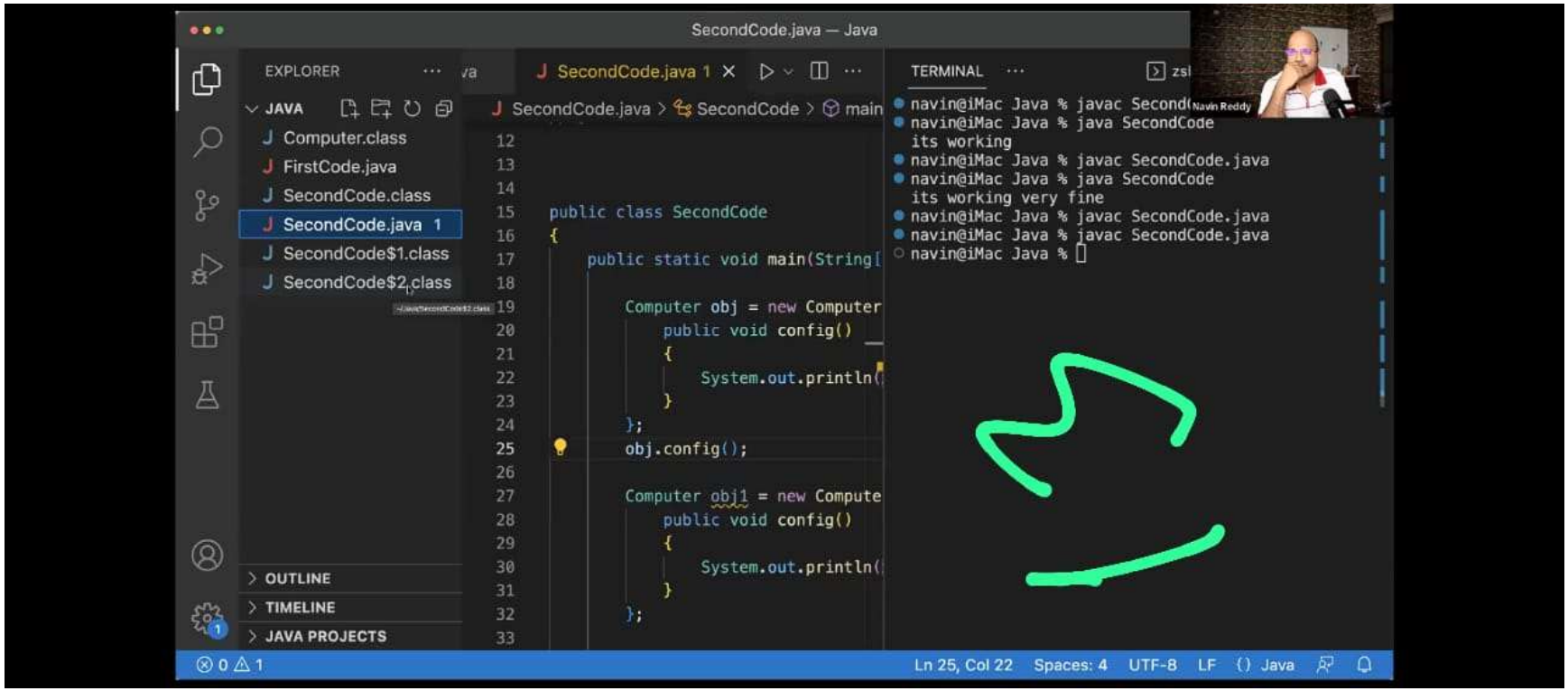

FirstCode.java — Java
FirstCode.java 4
FirstCode.java > A > show()
class A
{
public void show()
{
System.out.prin
class B
{
public void con
{
System.out.println(x: "in config ");
}
}
}
}
public class FirstCode
{
public static void main(String[] args)
{
void java.io.PrintStream.println(String x)
Prints a String and then terminates the line. This method behaves as though it invokes print(String) and then println().
• Parameters:
x The String to be printed.
TERMINAL
zsh
navin@iMac Java % java Firs
hi
Hello
navin@iMac Java % java FirstCode.java
in show 0
navin@iMac Java % java FirstCode.java
navin@iMac Java % java FirstCode
in show
navin@iMac Java %
Navin Reddy
Ln 5, Col 21 (4 selected)
Spaces: 4
UTF-8
LF
Java

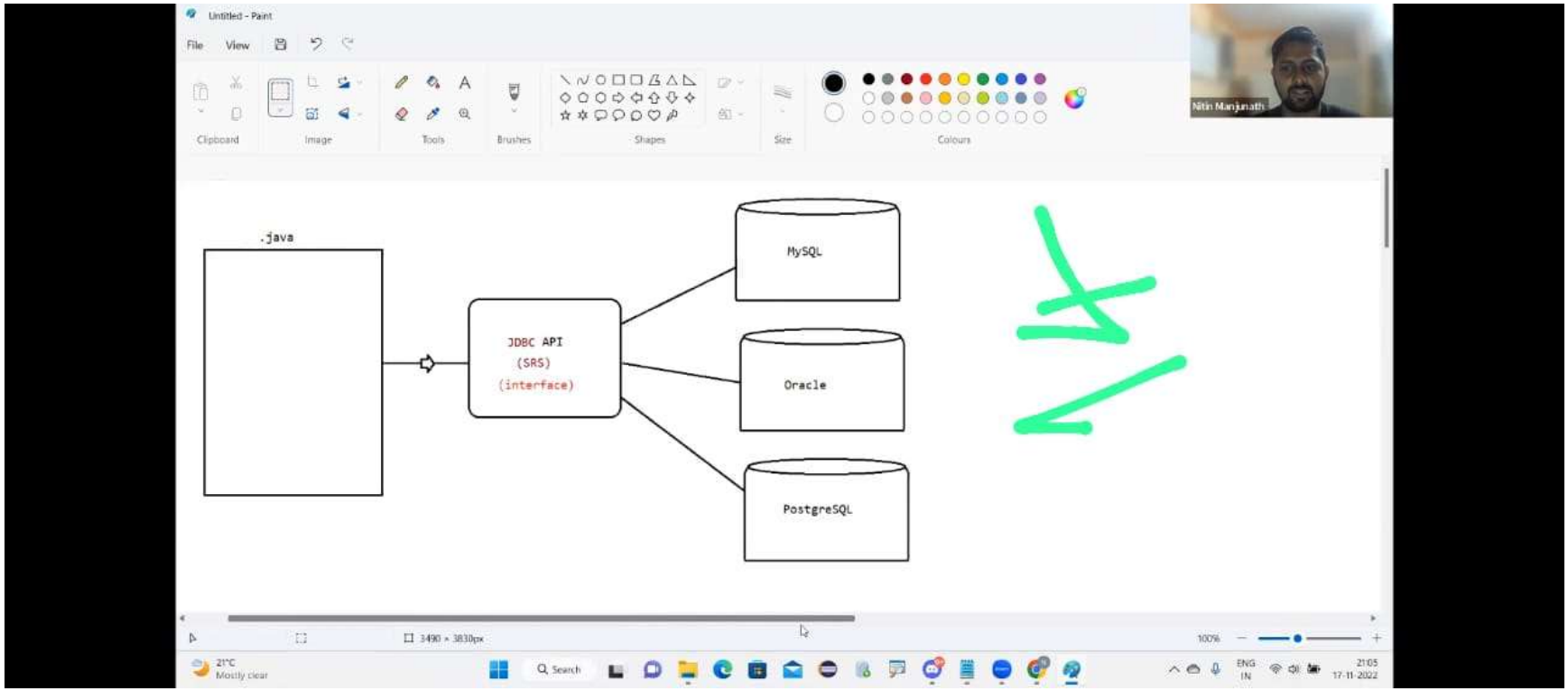

FirstCode.java — Java
FirstCode.java
FirstCode.java > A
1 /*
2 * Build by Navin
3 *
4 *
5 */
6
7 class A {
8 public void show() {
9 System.out.println(x: "in show");
10 }
11
12 class B {
13 public void config() {
14 System.out.println(x: "in config ");
15 }
16 }
17 }
18
19 // Main class
20 public class FirstCode {
21
22 public static void main(String[] args) {
TERMINAL
navin@iMac Java % java Firs
hi
Hello
navin@iMac Java % java FirstCode.java
in show 0
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java % java FirstCode.java
in show 0
in config
navin@iMac Java % javac FirstCode.java
navin@iMac Java % javac FirstCode.java
navin@iMac Java % javac FirstCode.java
navin@iMac Java % java FirstCode
in show
navin@iMac Java % javadoc FirstCode.java
Loading source file FirstCode.java...
Constructing Javadoc information...
Building index for all the packages and classes
...
Standard Doclet version 19+36-2238
Building tree for all the packages and classes.
..
Generating ./FirstCode.html...
Navin Reddy
⊗ 0 ⚠ 0
Ln 2, Col 10
Spaces: 4
UTF-8
LF
{} Java

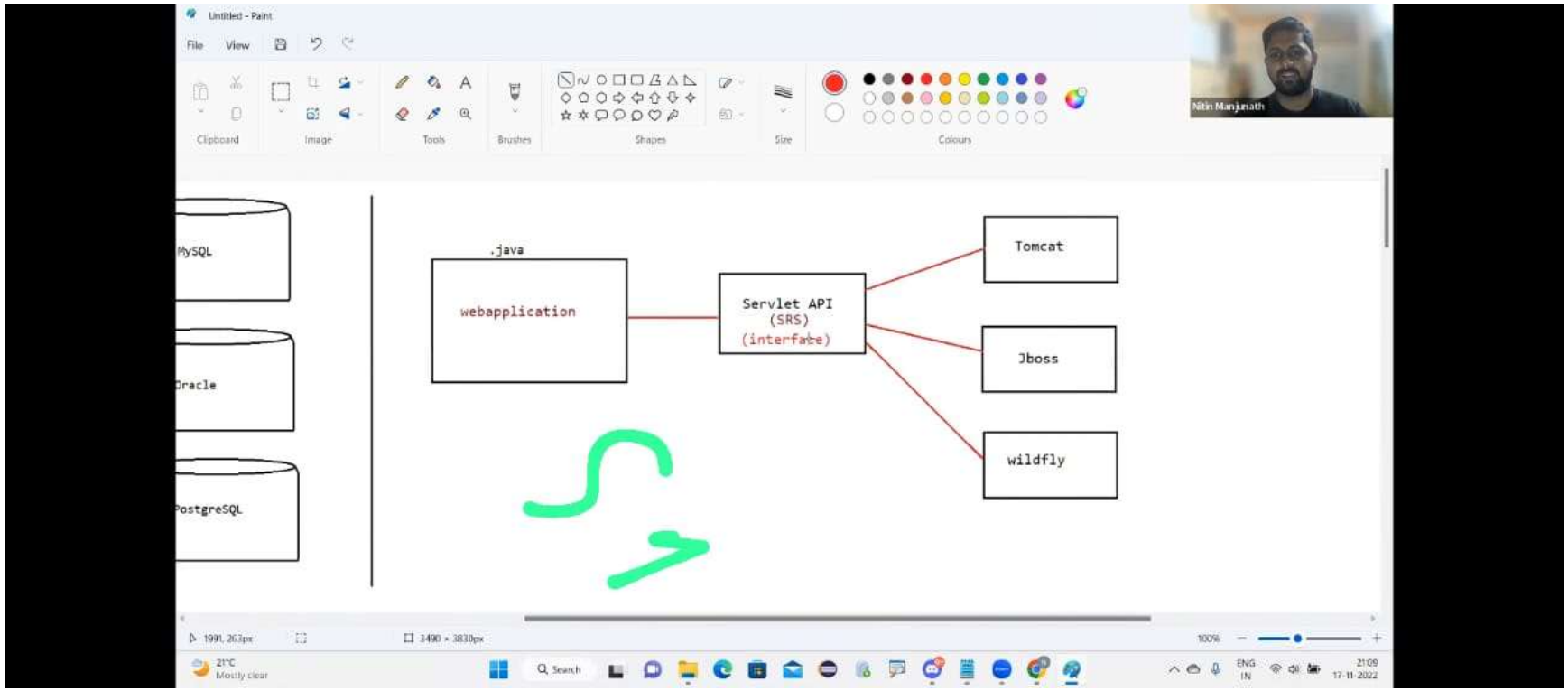

SecondCode.java — Java
FirstCode.java
SecondCode.java
condCode.java > SecondCode > main(String[]) > new Computer() {...}
4 {
5 System.out.println(x: "in Computer Config");
6 }
7 }
8
9
10
11 public class SecondCode
12 {
13 public static void main(String[] args) {
14
15 Computer obj = new Computer() // anonymous inner
16 {
17 public void config()
18 {
19 System.out.println(x: "something new");
20 }
21 };
22 obj.config();
23
24 }
25 }
TERMINAL
zsh
navin@iMac Java % javac FirstC
navin@iMac Java % javac FirstCode.java
navin@iMac Java % javac SecondCode.java
navin@iMac Java % java SecondCode.java
error: can't find main(String[]) method in class: C
omputer
navin@iMac Java % java SecondCode
in Computer Config
navin@iMac Java % java SecondCode.java
error: can't find main(String[]) method in class: C
omputer
navin@iMac Java % javac SecondCode.java
navin@iMac Java % java SecondCode
in AdvComputer config
navin@iMac Java % javac SecondCode.java
navin@iMac Java % java SecondCode
something new
navin@iMac Java %
Navin Reddy
0 0
Ln 15, Col 41
Spaces: 4
UTF-8
LF
Java

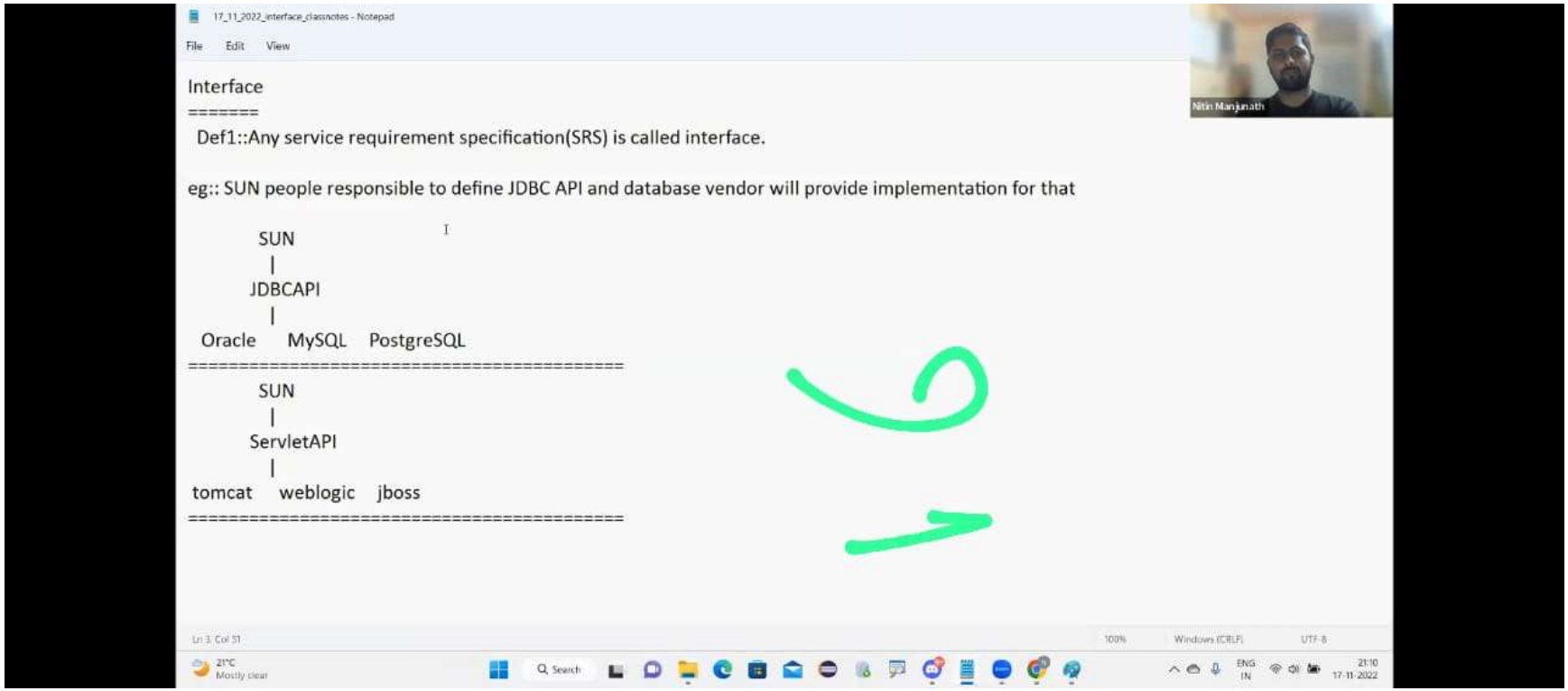

SecondCode.java — Java
FirstCode.java
SecondCode.java
SecondCode.java > SecondCode > main(String[])
1 abstract class Computer
2 {
3 public abstract void config();
4 }
5 class Laptop extends Computer
6 {
7 public void config()
8 {
9 System.out.println(x: "its working");
10 }
11 }
12
13
14
15 public class SecondCode
16 {
17 public static void main(String[] args) {
18
19 Computer obj = new Laptop();
20 obj.config();
21
22 }
TERMINAL
navin@iMac Java % javac Second
navin@iMac Java % java SecondCode
its working
navin@iMac Java %
Navin Reddy
0 0
Ln 20, Col 22
Spaces: 4
UTF-8
LF
Java

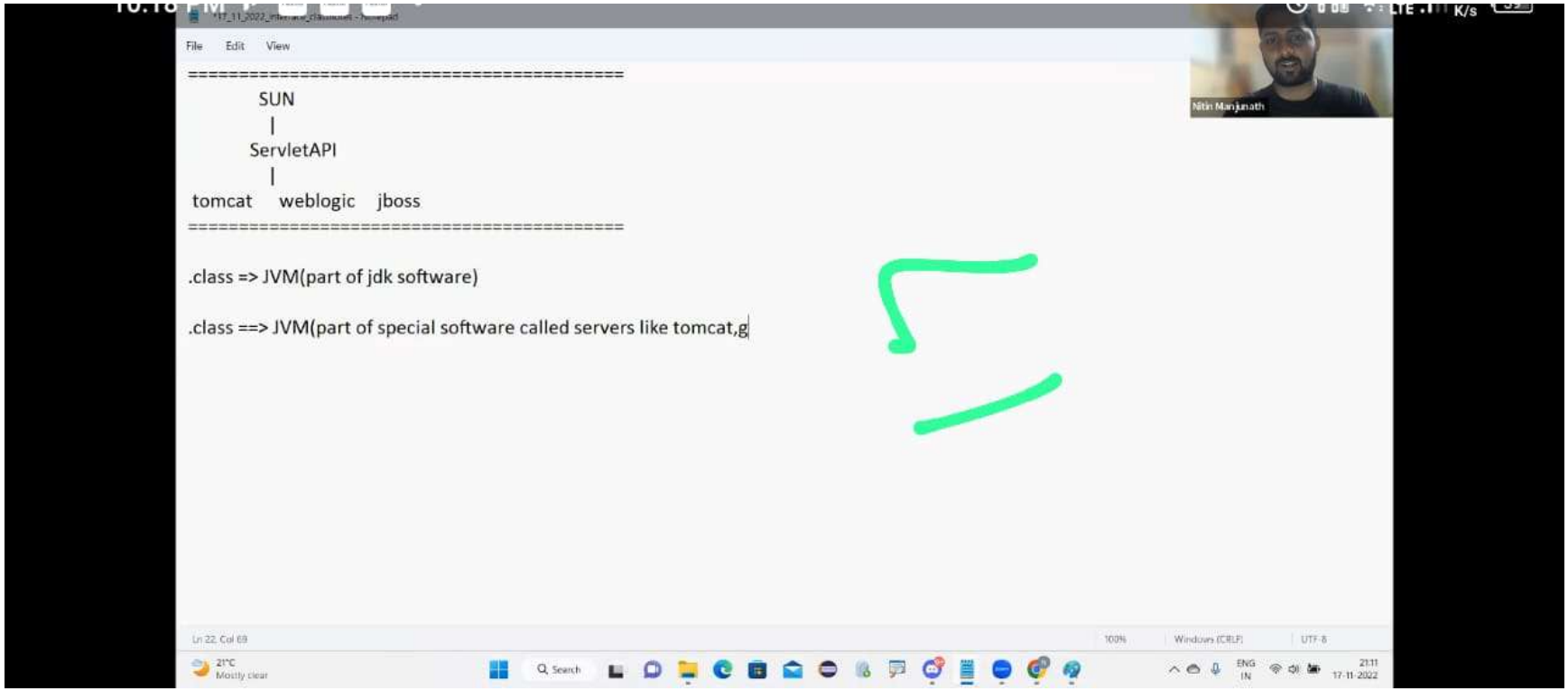

SecondCode.java — Java
FirstCode.java
SecondCode.java
SecondCode > main(String[]) > new Computer() {...} > config()
// public void config()
// {
// System.out.println("its working");
// }
// }
public class SecondCode
{
public static void main(String[] args) {
Computer obj = new Computer() {
public void config()
{
System.out.println(x: "its working very fin
}
};
obj.config();
}
}
TERMINAL
zsh
navin@iMac Java % javac Second
navin@iMac Java % java SecondCode
its working
navin@iMac Java % javac SecondCode.java
navin@iMac Java % java SecondCode
its working very fine
navin@iMac Java %
Navin Reddy
0 0
Ln 22, Col 58
Spaces: 4
UTF-8
LF
Java

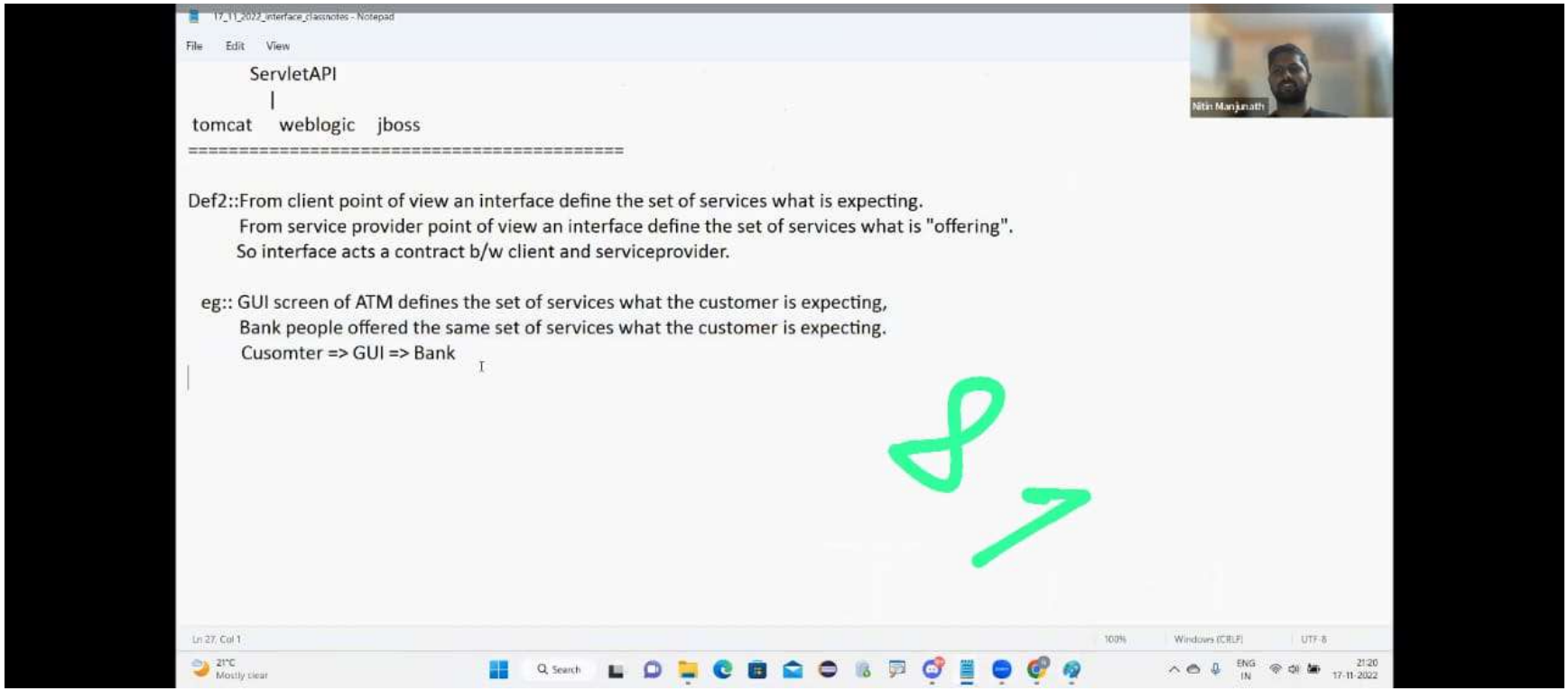

SecondCode.java — Java
EXPLORER
JAVA
Computer.class
FirstCode.java
SecondCode.class
SecondCode.java 1
SecondCode$1.class
SecondCode$2.class
OUTLINE
TIMELINE
JAVA PROJECTS
SecondCode.java 1
SecondCode.java > SecondCode > main
public class SecondCode
{
public static void main(String[
Computer obj = new Computer
public void config()
System.out.println(
};
obj.config();
Computer obj1 = new Compute
public void config()
System.out.println(
};
TERMINAL
navin@iMac Java % javac Second
navin@iMac Java % java SecondCode
its working
navin@iMac Java % javac SecondCode.java
navin@iMac Java % java SecondCode
its working very fine
navin@iMac Java % javac SecondCode.java
navin@iMac Java % javac SecondCode.java
navin@iMac Java %
Navin Reddy
Ln 25, Col 22
Spaces: 4
UTF-8
LF
Java

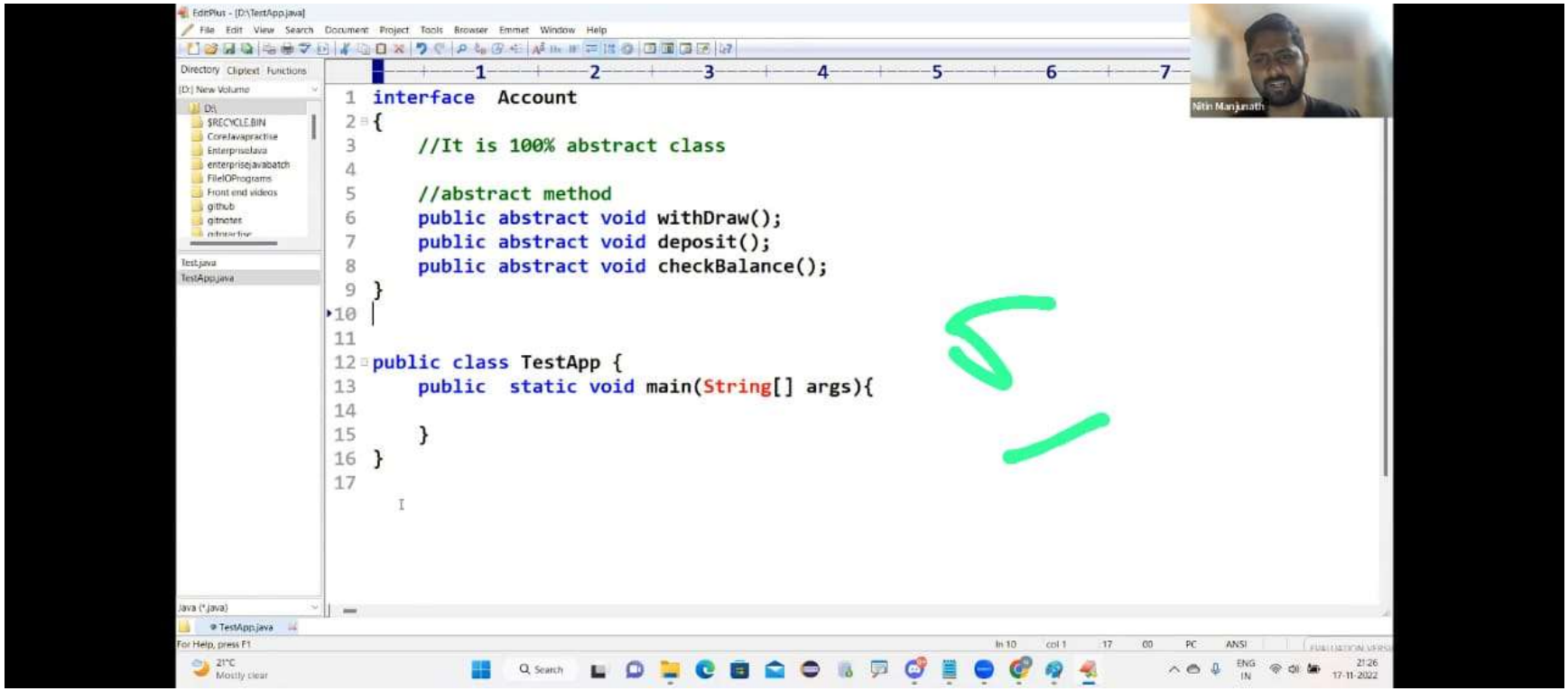

Untitled - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
.java
JDBC API
(SRS)
(interface)
MySQL
Oracle
PostgreSQL
3490 × 3830px
100%
21°C
Mostly clear
Search
ENG
IN
21:05
17-11-2022

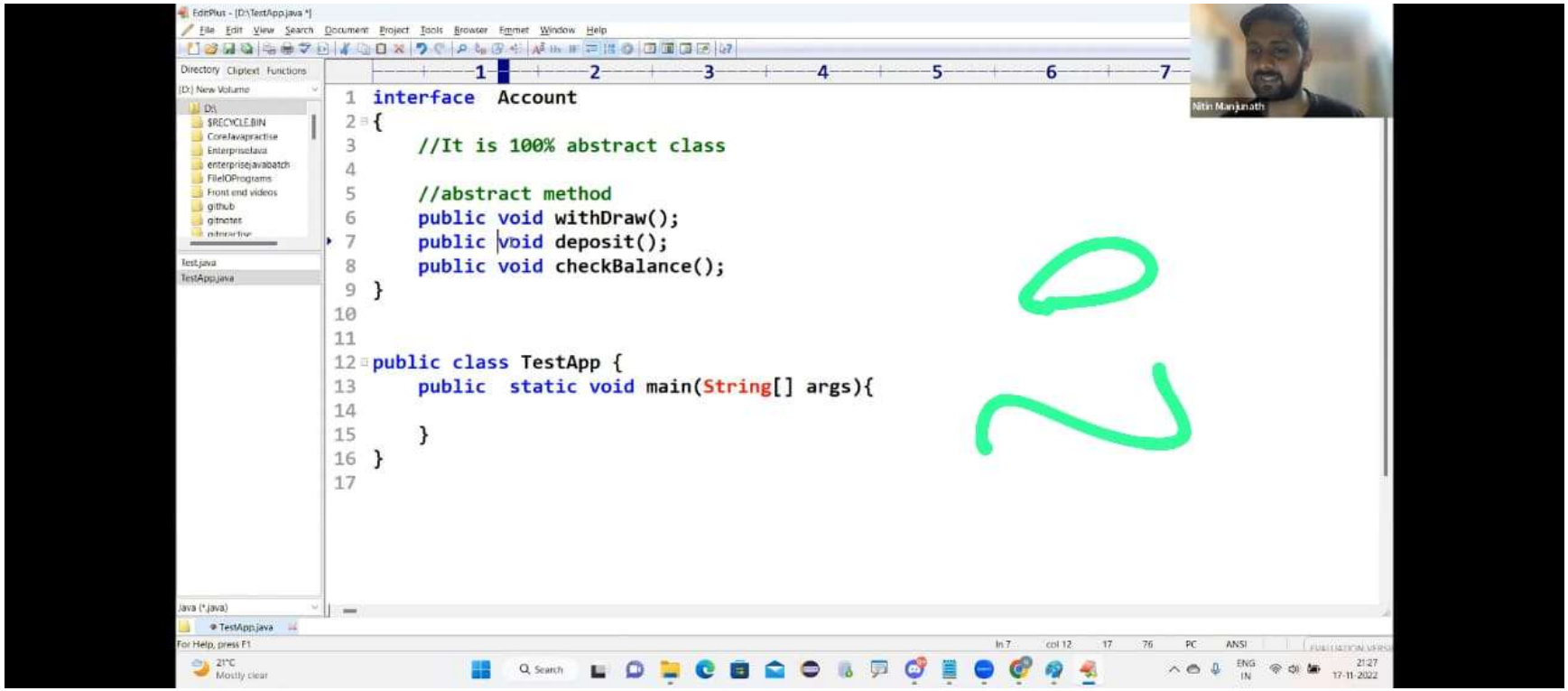

Untitled - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
MySQL
Oracle
PostgreSQL
.java
webapplication
Servlet API
(SRS)
(interface)
Tomcat
Jboss
wildfly
1991, 263px
3490 × 3830px
100%
21°C
Mostly clear
Search
ENG
IN
21:09
17-11-2022

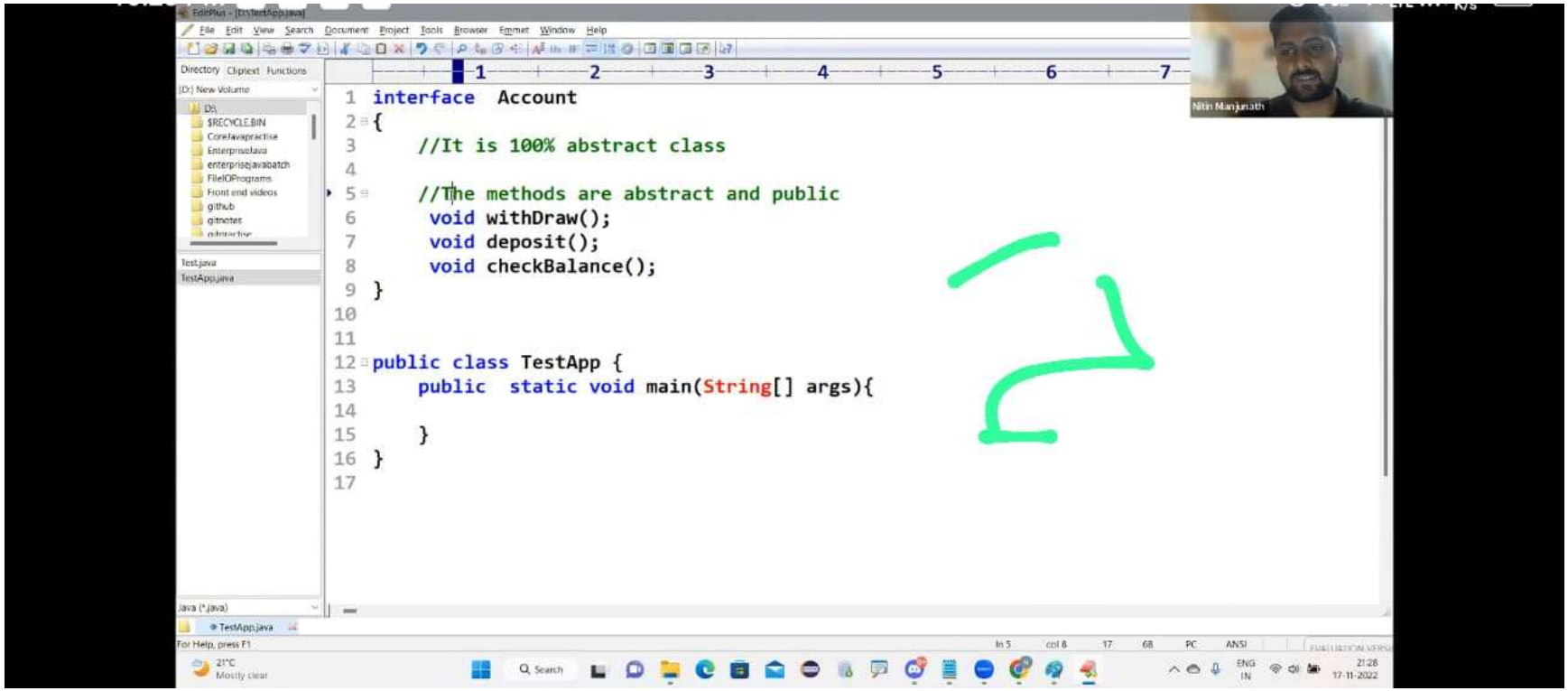

17_11_2022_interface_classnotes - Notepad
File Edit View
Interface
=======
Def1::Any service requirement specification(SRS) is called interface.
eg:: SUN people responsible to define JDBC API and database vendor will provide implementation for that
SUN
|
JDBCAPI
|
Oracle MySQL PostgreSQL
==========
SUN
|
ServletAPI
|
tomcat weblogic jboss
==========
Nitin Manjunath
Ln 3, Col 51
100%
Windows (CRLF)
UTF-8
21°C
Mostly clear
Search
ENG
IN
21:10
17-11-2022


File Edit View
==========
SUN
|
ServletAPI
|
tomcat weblogic jboss
==========
.class => JVM(part of jdk software)
.class ==> JVM(part of special software called servers like tomcat,g
Nitin Manjunath
Ln 22, Col 69
100%
Windows (CRLF)
UTF-8
21°C
Mostly clear
Search
ENG
IN
21:11
17-11-2022


17_11_2022_interface_classnotes - Notepad
File Edit View
ServletAPI
tomcat weblogic jboss
==========
Def2::From client point of view an interface define the set of services what is expecting.
From service provider point of view an interface define the set of services what is "offering".
So interface acts a contract b/w client and serviceprovider.
eg:: GUI screen of ATM defines the set of services what the customer is expecting,
Bank people offered the same set of services what the customer is expecting.
Cusomter => GUI => Bank
Nitin Manjunath
Ln 27, Col 1
100%
Windows (CRLF)
UTF-8
21°C
Mostly clear
Search
ENG
IN
21:20
17-11-2022


EditPlus - [D:\TestApp.java]
File Edit View Search Document Project Tools Browser Emmet Window Help
Directory Cliptext Functions
Nitin Manjunath
1 interface Account
2 {
3 //It is 100% abstract class
4
5 //abstract method
6 public abstract void withDraw();
7 public abstract void deposit();
8 public abstract void checkBalance();
9 }
10
11
12 public class TestApp {
13 public static void main(String[] args){
14
15 }
16 }
17
TestApp.java
For Help, press F1
Search
21:26
17-11-2022


EditPlus - [D:\TestApp.java *]
File Edit View Search Document Project Tools Browser Format Window Help
Directory Cliptext Functions
(D:) New Volume
D:\
$RECYCLE.BIN
Corejavapractise
Enterprisejava
enterprisejavabatch
FileIOPrograms
Front end videos
github
gitnotes
test.java
TestApp.java
Java (*.java)
TestApp.java
For Help, press F1
Nitin Manjunath
1 interface Account
2 {
3 //It is 100% abstract class
4
5 //abstract method
6 public void withDraw();
7 public void deposit();
8 public void checkBalance();
9 }
10
11
12 public class TestApp {
13 public static void main(String[] args){
14
15 }
16 }
17
20
21°C
Mostly clear
Search
ENG
IN
21:27
17-11-2022


Nitin Manjunath
interface Account
{
//It is 100% abstract class
//The methods are abstract and public
void withDraw();
void deposit();
void checkBalance();
}
public class TestApp {
public static void main(String[] args){
}
}

17_11_2022_interface_classnotes - Notepad

File Edit View

So interface acts a contract b/w client and serviceprovider.

eg:: GUI screen of ATM defines the set of services what the customer is expecting,
Bank people offered the same set of services what the customer is expecting.
Cusomter => GUI => Bank

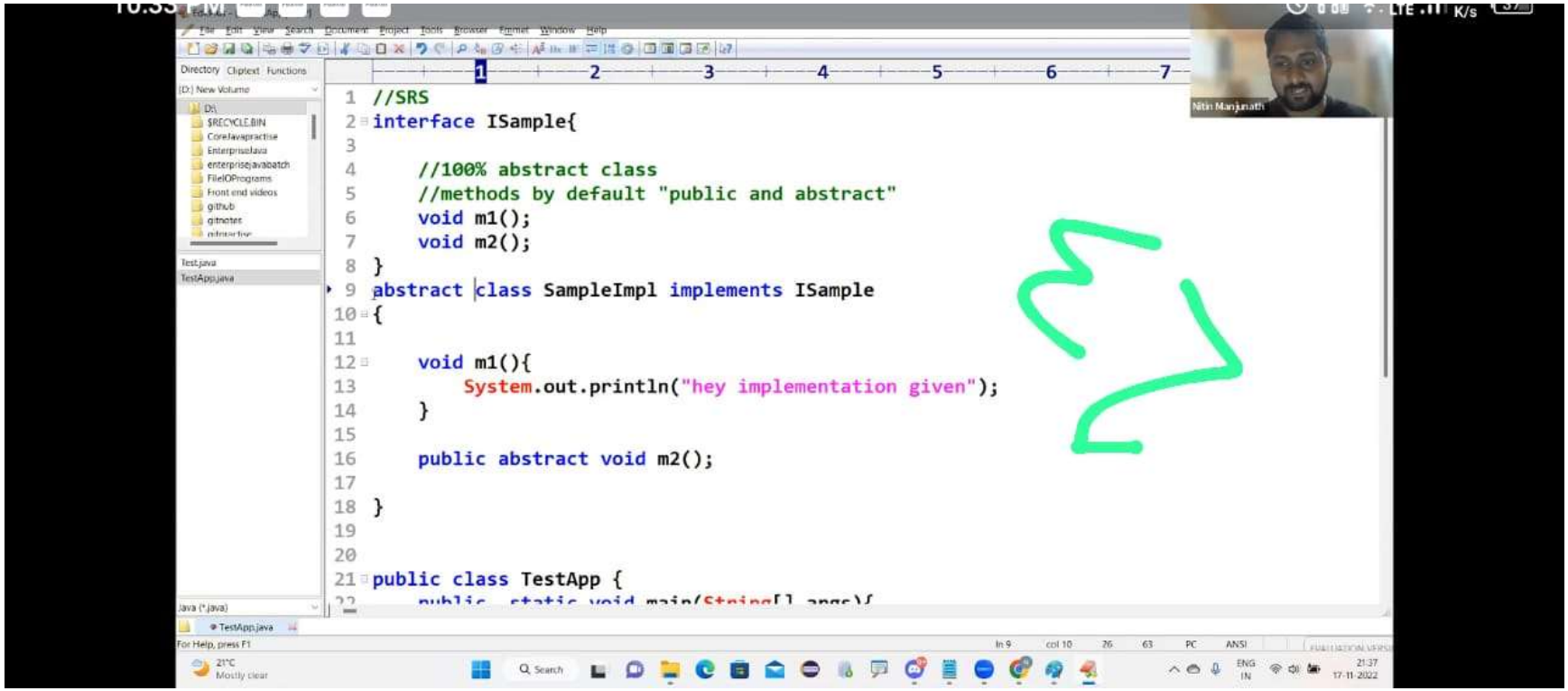
Def3:: Inside interface every method is always abstract whether we are delcaring or not hence interface is considered as 100% pure abstract class.

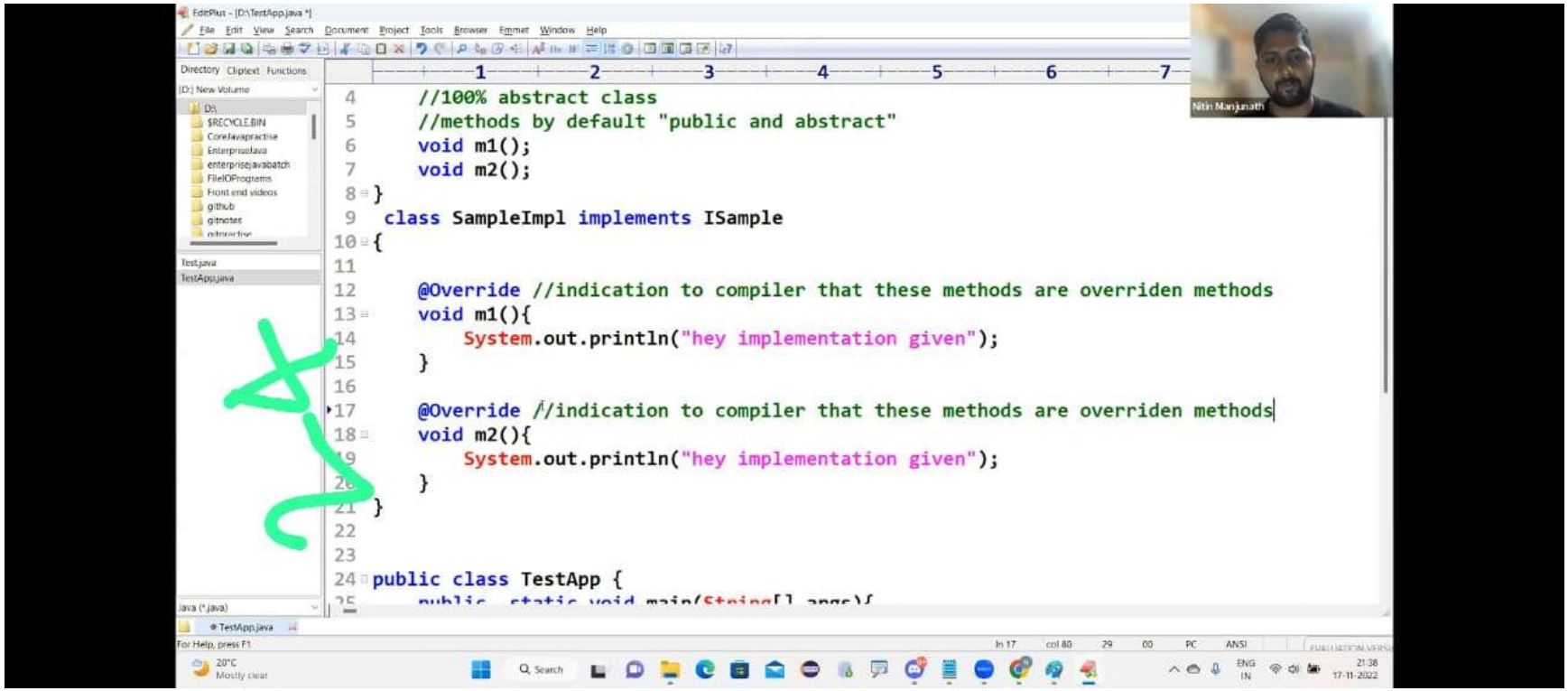
eg:

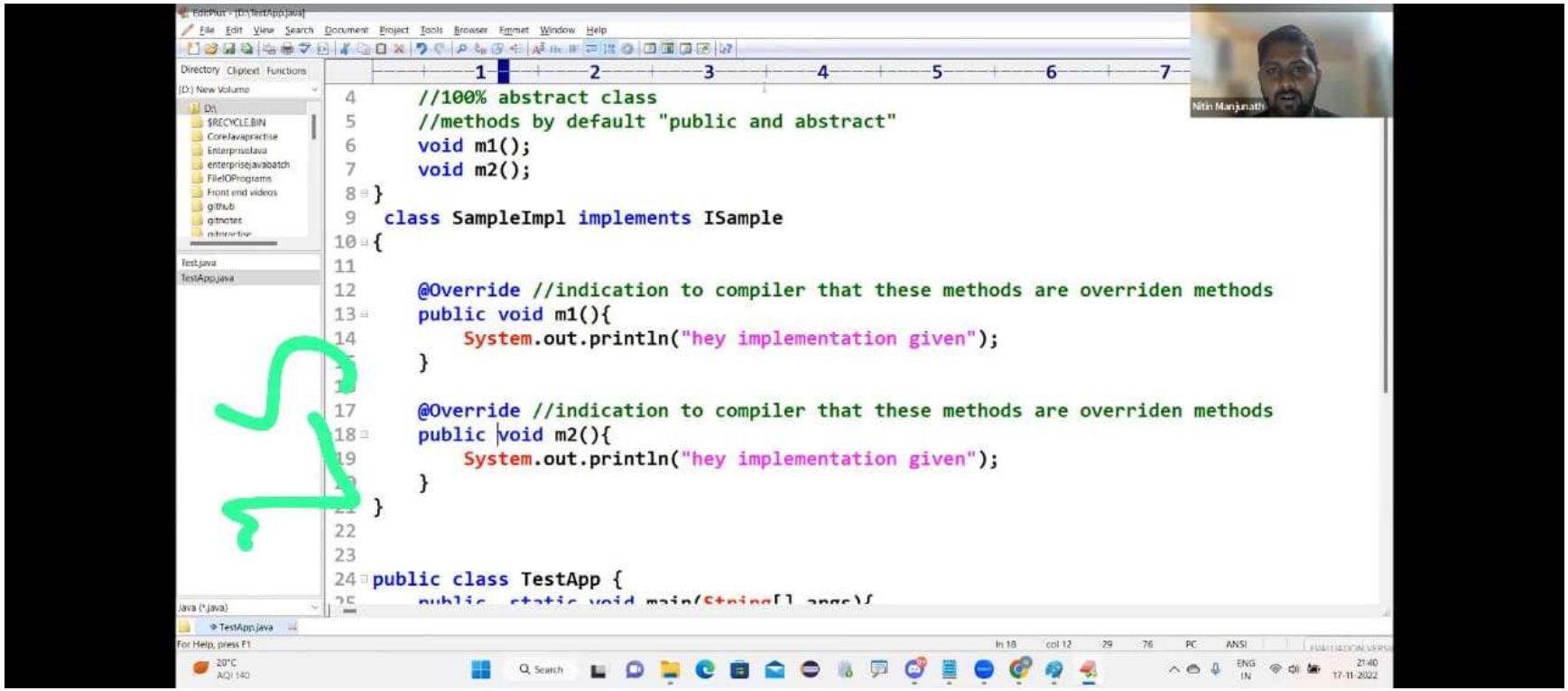
```
interface  Account
{
     //It is 100% abstract class

     //The methods are by default "abstract and public"
      void withDraw();
      void deposit();
      void checkBalance();
}
```

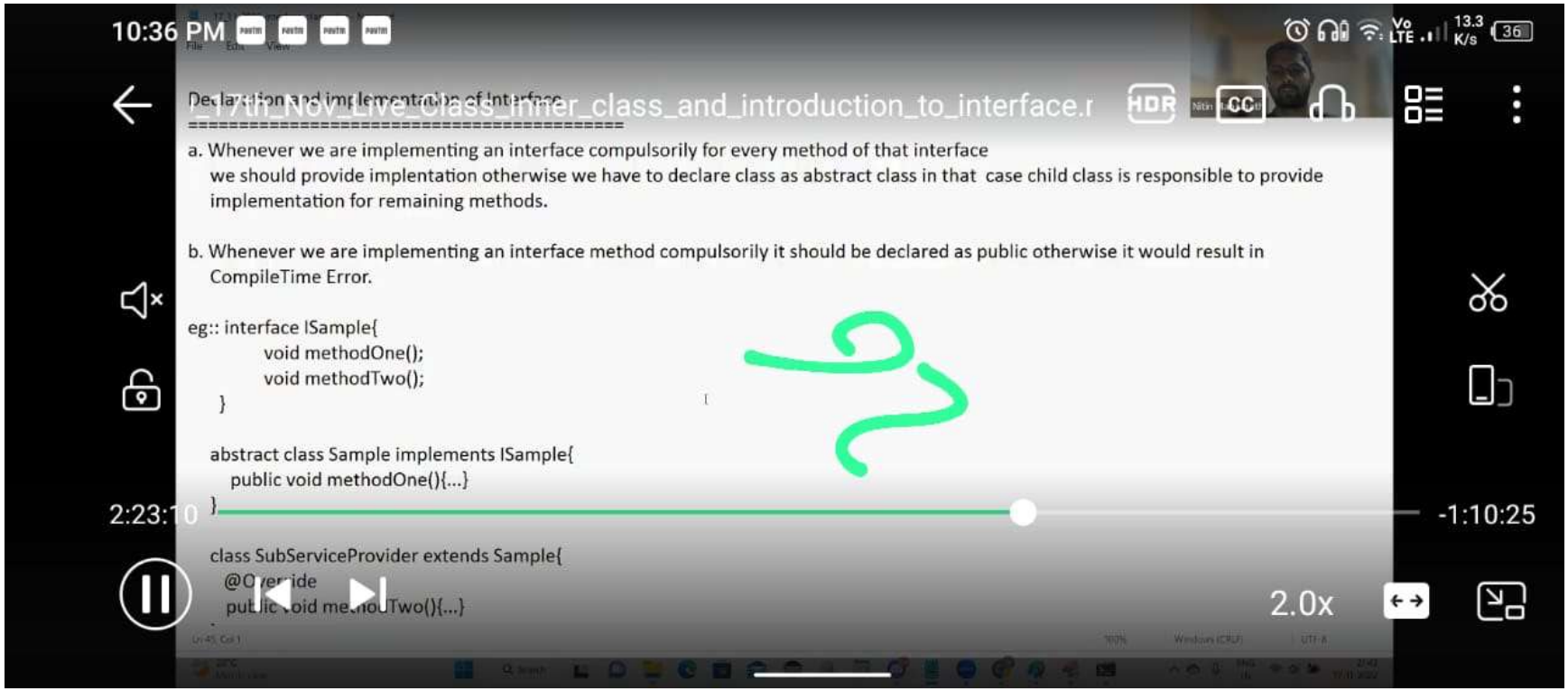
Ln 36, Col 52 100% Windows (CRLF) UTF-8

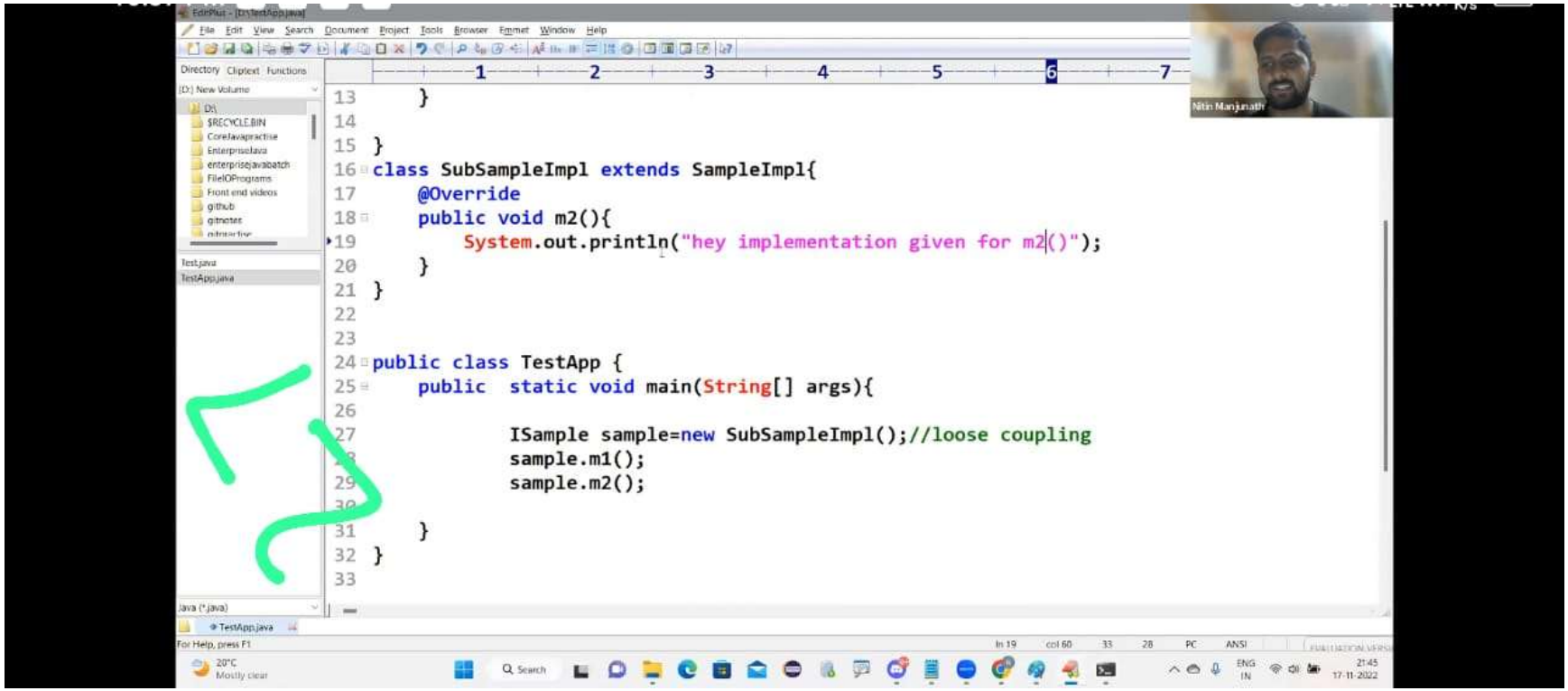
21°C Mostly clear Search ENG IN 21:29 17-11-2022

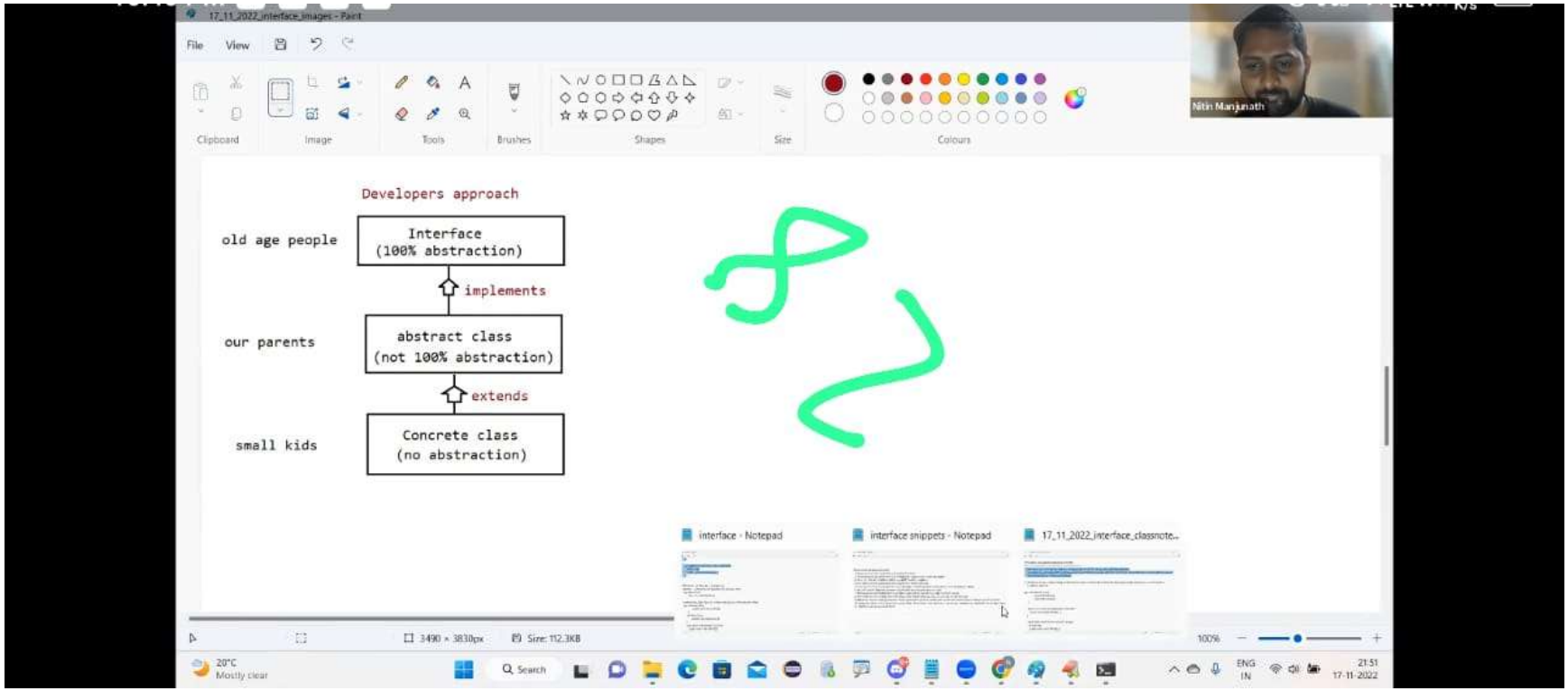

Nitin Manjunath
//SRS
interface ISample{
//100% abstract class
//methods by default "public and abstract"
void m1();
void m2();
}
abstract class SampleImpl implements ISample
{
void m1(){
System.out.println("hey implementation given");
}
public abstract void m2();
}
public class TestApp {

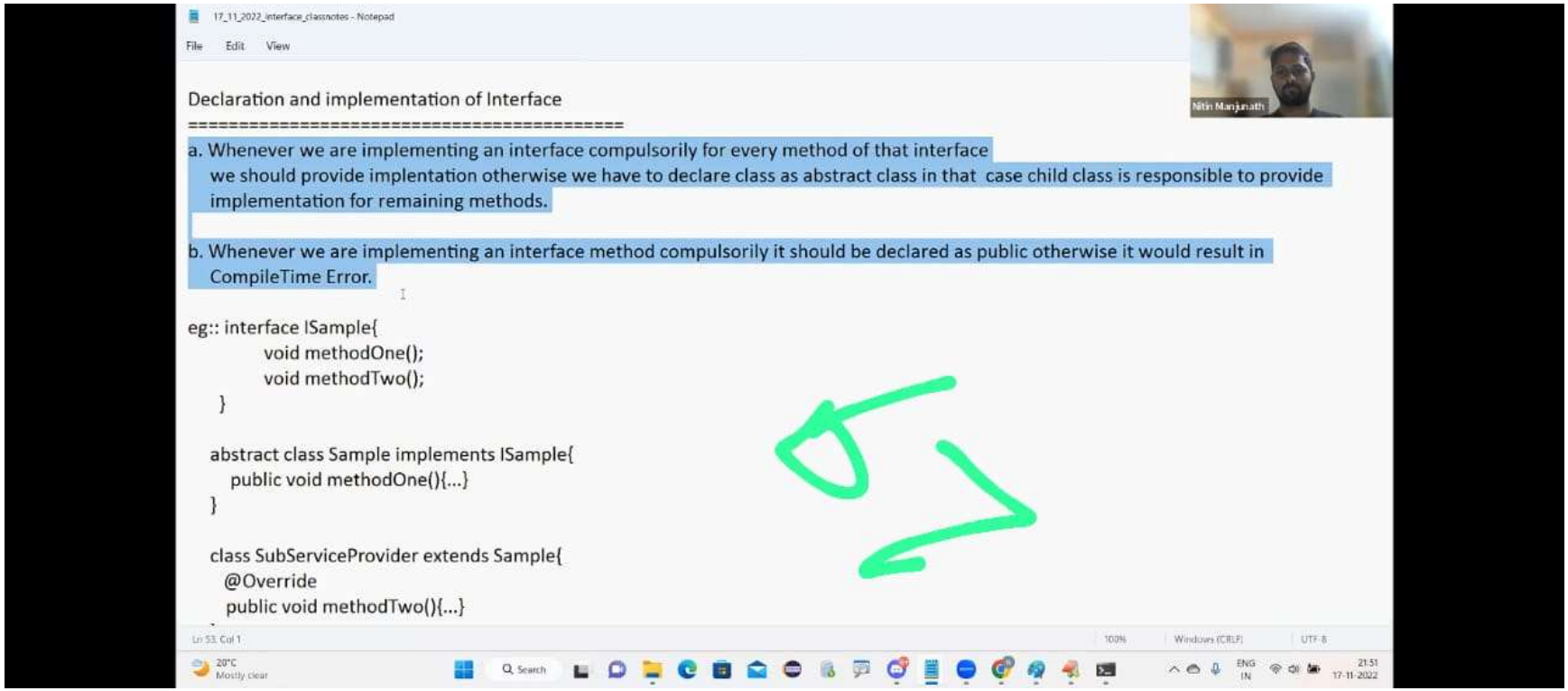

EditPlus - [D:\TestApp.java *]
File Edit View Search Document Project Tools Browser Format Window Help
Directory Cliptext Functions
(D:) New Volume
D:\
$RECYCLE.BIN
CoreJavapractise
Enterprisejava
enterprisejavabatch
FileIOPrograms
Front end videos
github
gitnotes
Test.java
TestApp.java
Java (*.java)
TestApp.java
For Help, press F1
Nitin Manjunath
4 //100% abstract class
5 //methods by default "public and abstract"
6 void m1();
7 void m2();
8 }
9 class SampleImpl implements ISample
10 {
11
12 @Override //indication to compiler that these methods are overriden methods
13 void m1(){
14 System.out.println("hey implementation given");
15 }
16
17 @Override //indication to compiler that these methods are overriden methods
18 void m2(){
19 System.out.println("hey implementation given");
20 }
21 }
22
23
24 public class TestApp {
25 public static void main(String[] args){
In 17 col 80 29 00 PC ANSI
20°C
Mostly clear
Search
ENG
IN
21:38
17-11-2022

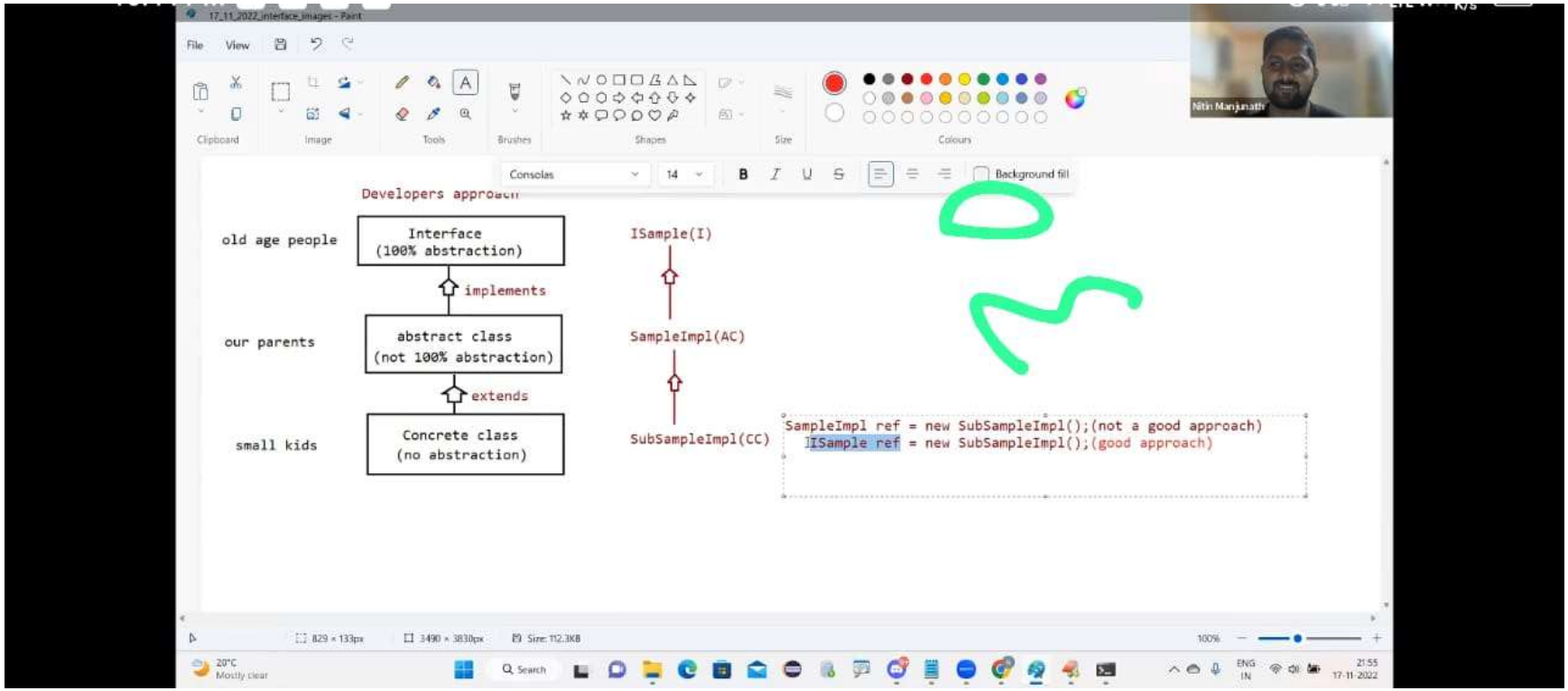

Nitin Manjunath
//100% abstract class
//methods by default "public and abstract"
void m1();
void m2();
}
class SampleImpl implements ISample
{
@Override //indication to compiler that these methods are overriden methods
public void m1(){
System.out.println("hey implementation given");
}
@Override //indication to compiler that these methods are overriden methods
public void m2(){
System.out.println("hey implementation given");
}
}
public class TestApp {

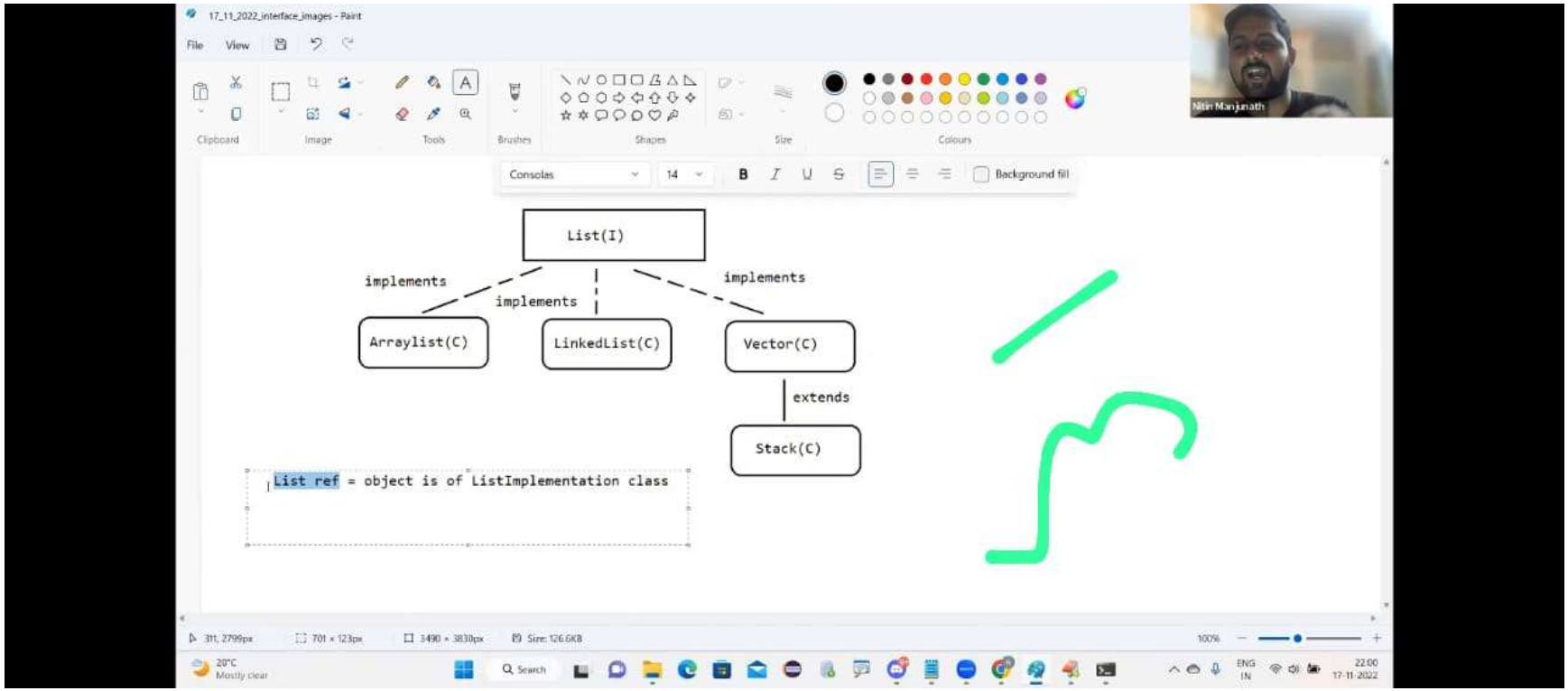

10:36 PM
Declaration and implementation of Interface
a. Whenever we are implementing an interface compulsorily for every method of that interface
we should provide implentation otherwise we have to declare class as abstract class in that case child class is responsible to provide
implementation for remaining methods.
b. Whenever we are implementing an interface method compulsorily it should be declared as public otherwise it would result in
CompileTime Error.
eg:: interface ISample{
void methodOne();
void methodTwo();
}
abstract class Sample implements ISample{
public void methodOne(){...}
}
class SubServiceProvider extends Sample{
@Override
public void methodTwo(){...}
2:23:10
-1:10:25
2.0x

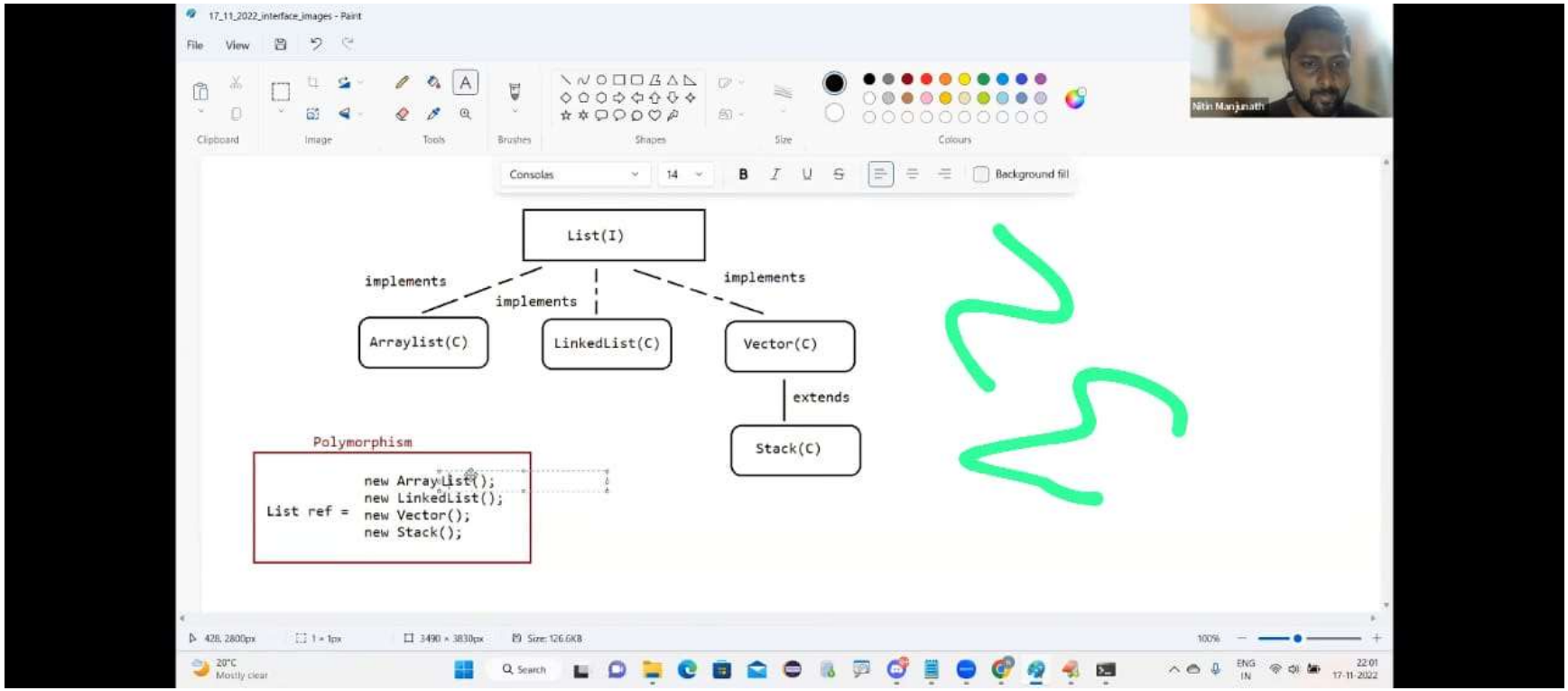

Nitin Manjunath
}
}
class SubSampleImpl extends SampleImpl{
@Override
public void m2(){
System.out.println("hey implementation given for m2()");
}
}
public class TestApp {
public static void main(String[] args){
ISample sample=new SubSampleImpl();//loose coupling
sample.m1();
sample.m2();
}
}

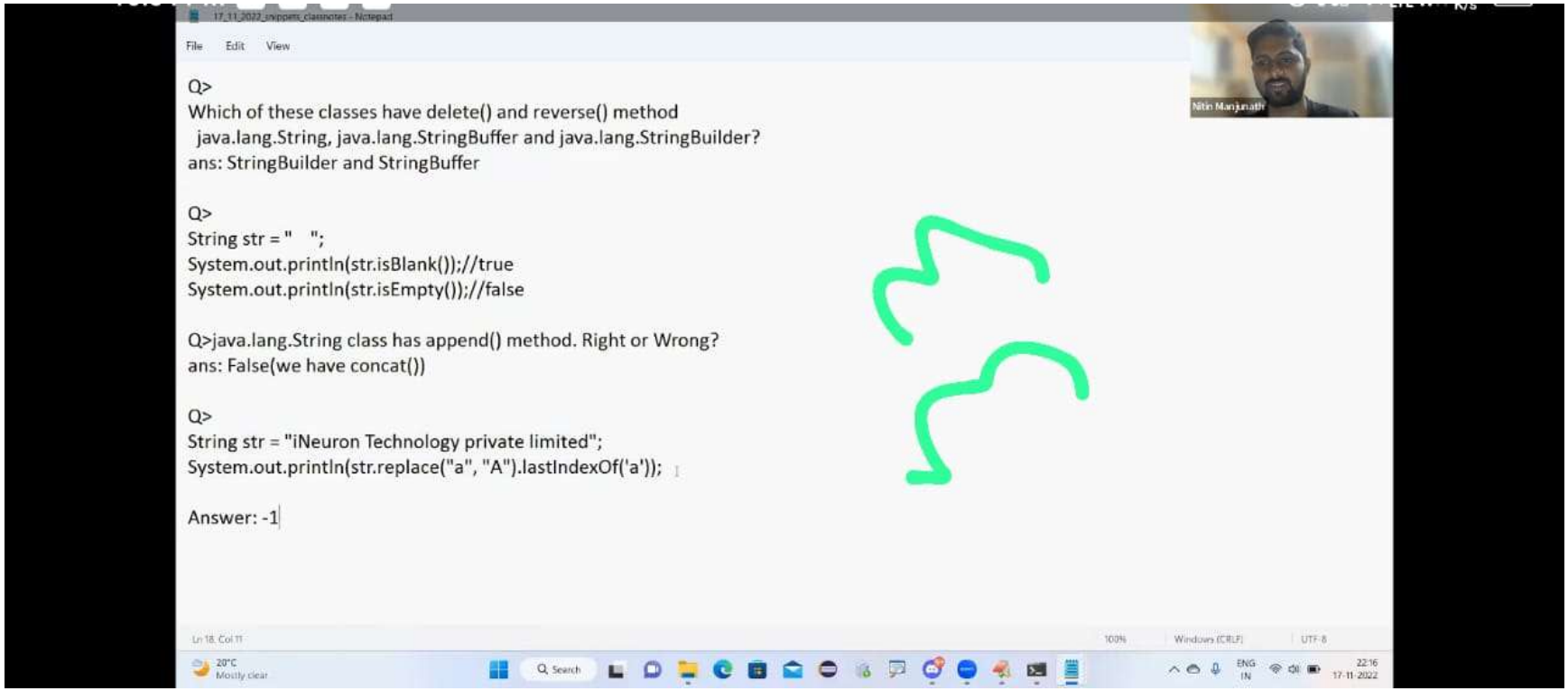

17_11_2022_interface_images - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
Developers approach
old age people
Interface
(100% abstraction)
implements
our parents
abstract class
(not 100% abstraction)
extends
small kids
Concrete class
(no abstraction)
interface - Notepad
interface snippets - Notepad
17_11_2022_interface_classnote...
3490 × 3830px
Size: 112.3KB
100%
20°C
Mostly clear
Search
ENG
IN
21:51
17-11-2022

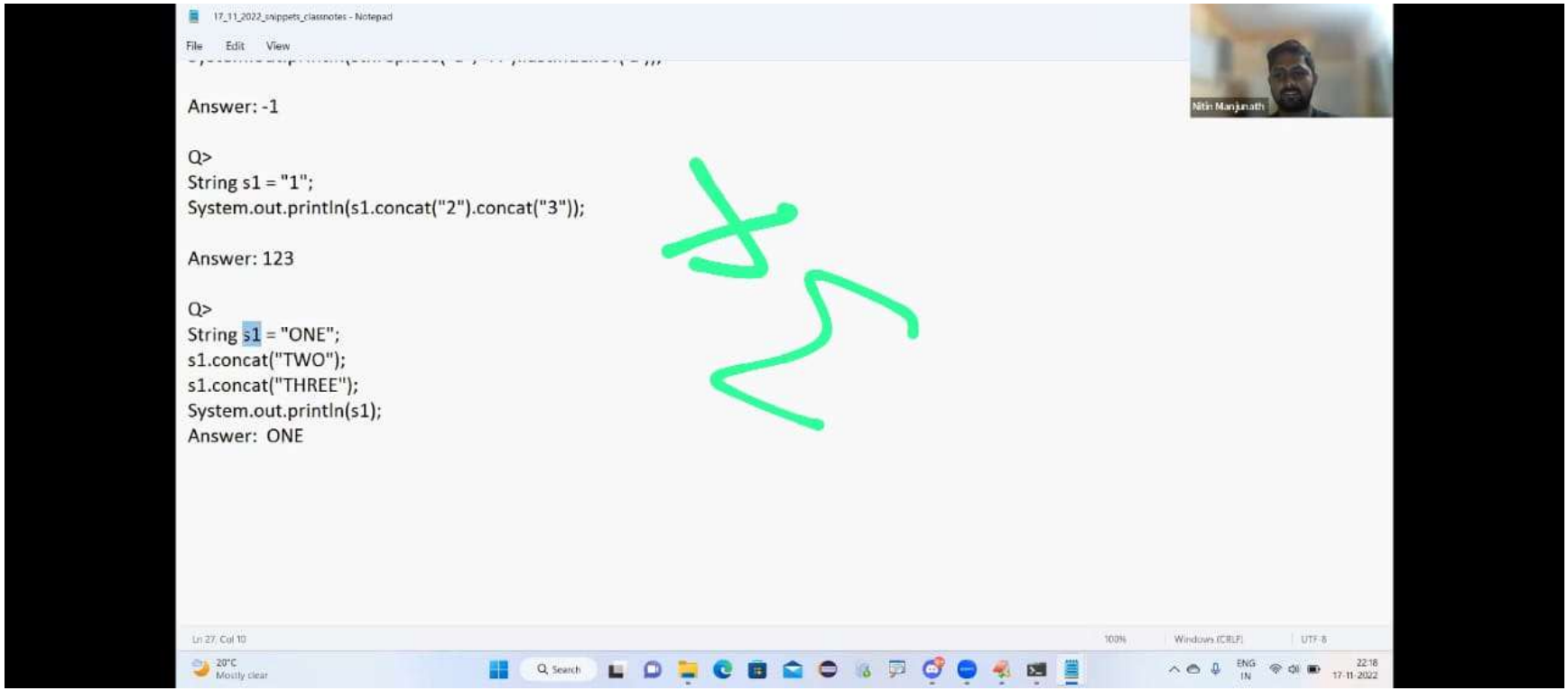

17_11_2022_interface_classnotes - Notepad
File Edit View
Nitin Manjunath
Declaration and implementation of Interface
a. Whenever we are implementing an interface compulsorily for every method of that interface we should provide implentation otherwise we have to declare class as abstract class in that case child class is responsible to provide implementation for remaining methods.
b. Whenever we are implementing an interface method compulsorily it should be declared as public otherwise it would result in CompileTime Error.
eg:: interface ISample{
void methodOne();
void methodTwo();
}
abstract class Sample implements ISample{
public void methodOne(){...}
}
class SubServiceProvider extends Sample{
@Override
public void methodTwo(){...}
Ln 53, Col 1
100%
Windows (CRLF)
UTF-8
20°C
Mostly clear
Search
ENG
IN
21:51
17-11-2022

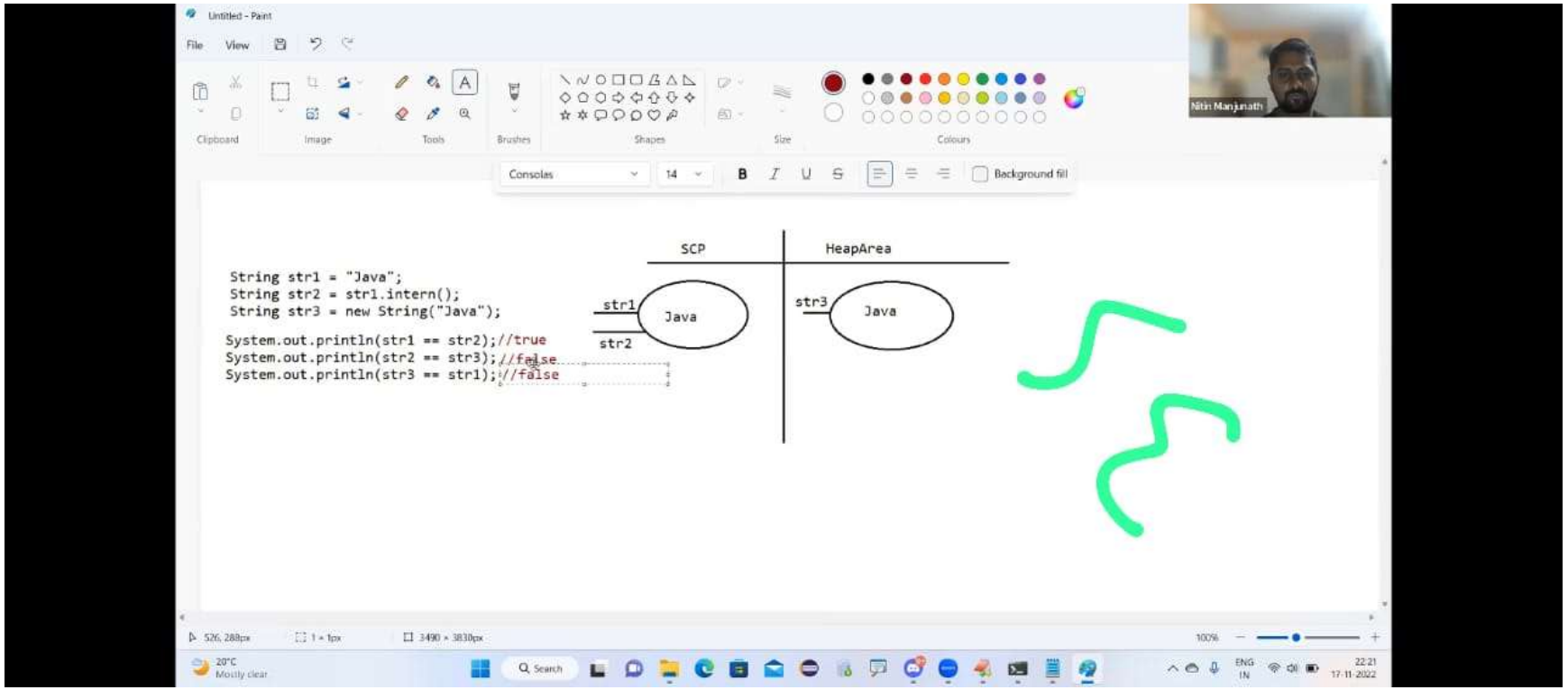

17_11_2022_interface_images - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
Consolas
14
Background fill
Developers approach
old age people
Interface
(100% abstraction)
implements
our parents
abstract class
(not 100% abstraction)
extends
small kids
Concrete class
(no abstraction)
ISample(I)
SampleImpl(AC)
SubSampleImpl(CC)
SampleImpl ref = new SubSampleImpl();(not a good approach)
ISample ref = new SubSampleImpl();(good approach)
829 × 133px
3490 × 3830px
Size: 112.3KB
100%
20°C
Mostly clear
Search
ENG
IN
21:55
17-11-2022

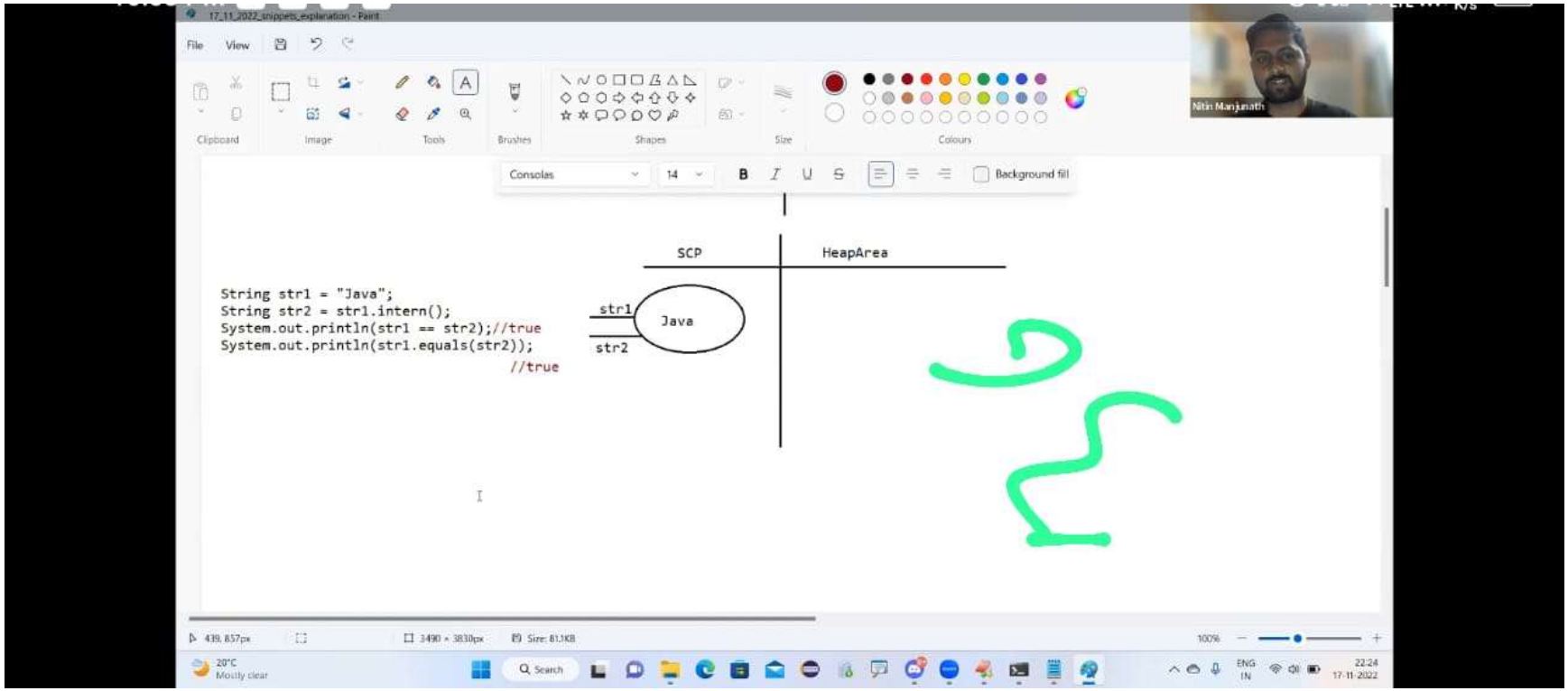

17_11_2022_interface_images - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
Consolas
14
Background fill
List(I)
implements
implements
implements
Arraylist(C)
LinkedList(C)
Vector(C)
extends
Stack(C)
List ref = object is of ListImplementation class
100%
20°C
Mostly clear
Search
ENG
IN
22:00
17-11-2022

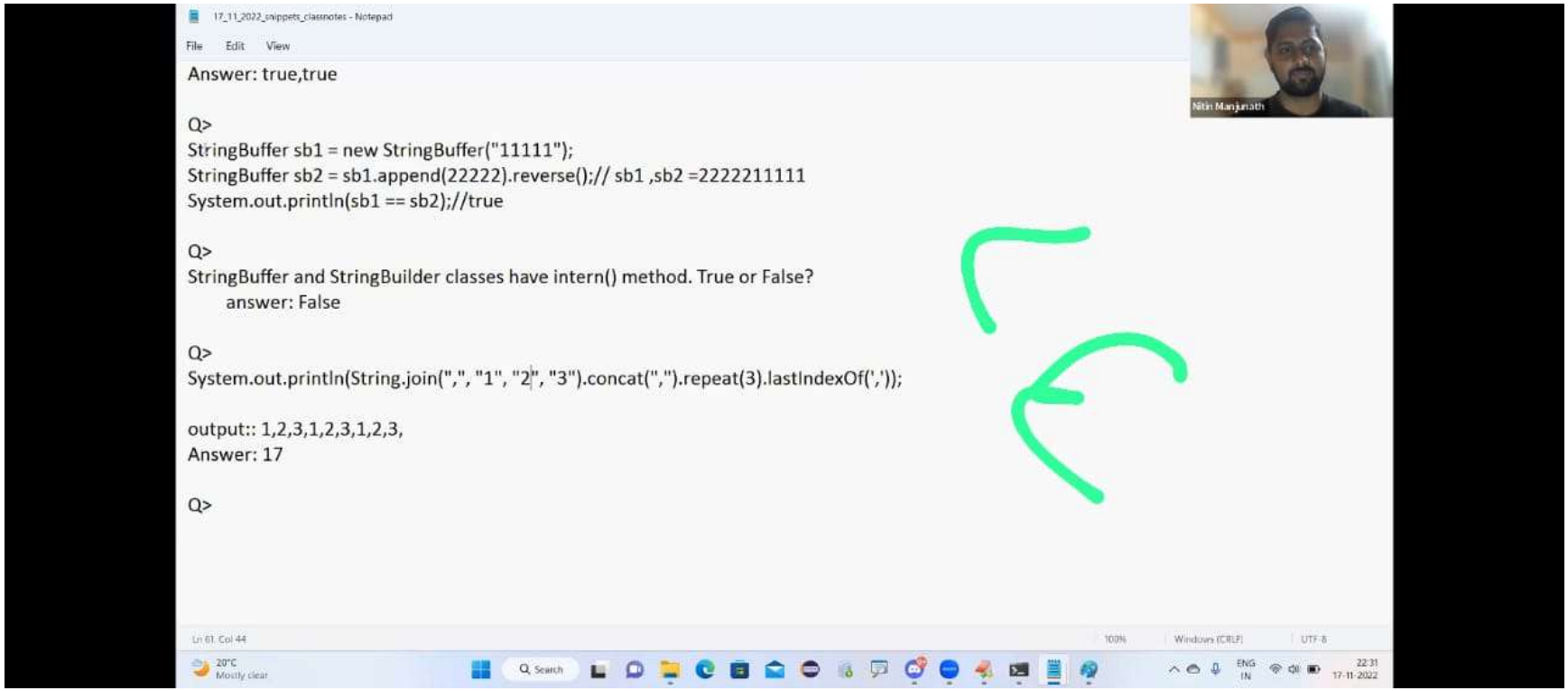

17_11_2022_interface_images - Paint
File
View
Nitin Manjunath
Consolas
14
Background fill
List(I)
implements
implements
implements
Arraylist(C)
LinkedList(C)
Vector(C)
extends
Stack(C)
Polymorphism
List ref =
new ArrayList();
new LinkedList();
new Vector();
new Stack();

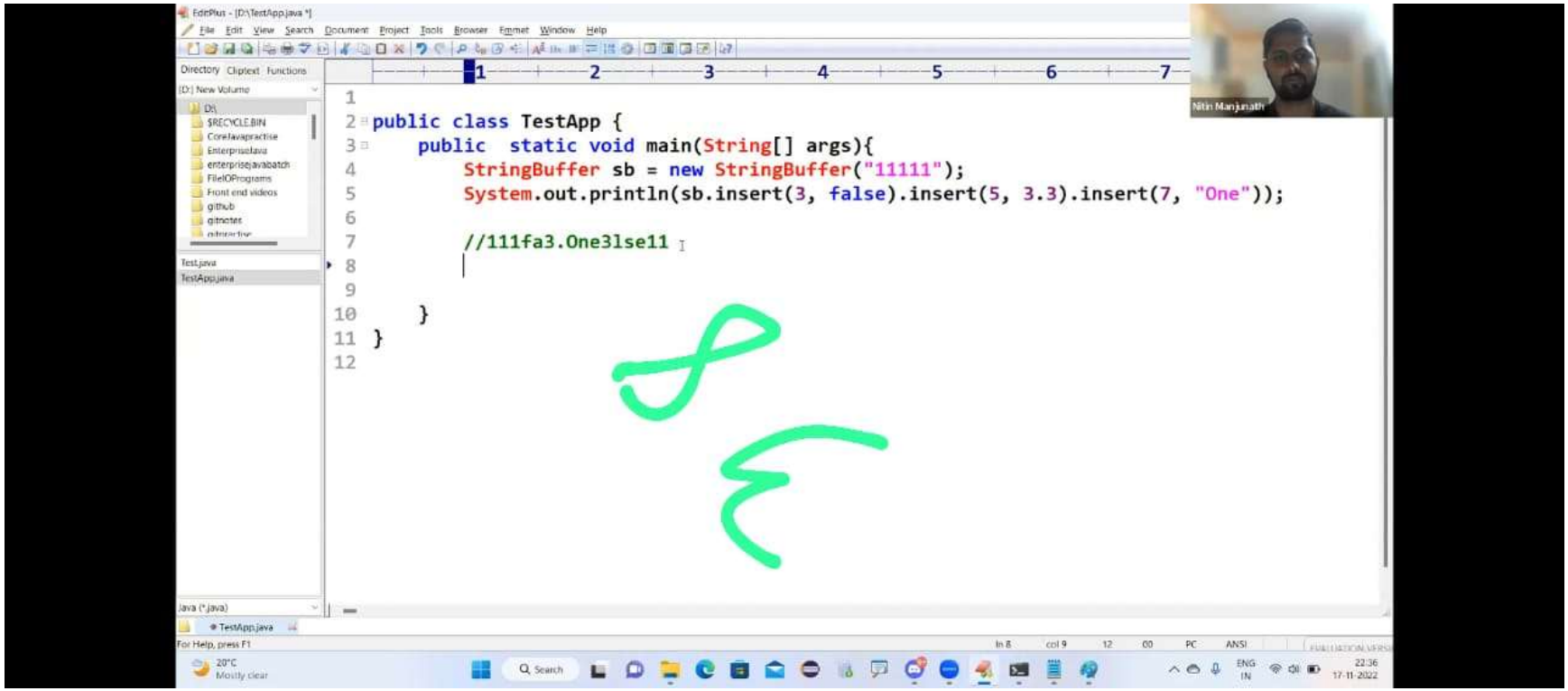

17_11_2022_snippets_classnotes - Notepad
File Edit View
Nitin Manjunath
Q>
Which of these classes have delete() and reverse() method
java.lang.String, java.lang.StringBuffer and java.lang.StringBuilder?
ans: StringBuilder and StringBuffer
Q>
String str = " ";
System.out.println(str.isBlank());//true
System.out.println(str.isEmpty());//false
Q>java.lang.String class has append() method. Right or Wrong?
ans: False(we have concat())
Q>
String str = "iNeuron Technology private limited";
System.out.println(str.replace("a", "A").lastIndexOf('a'));
Answer: -1
Ln 18, Col 11
100%
Windows (CRLF)
UTF-8
20°C
Mostly clear
Search
ENG
IN
22:16
17-11-2022

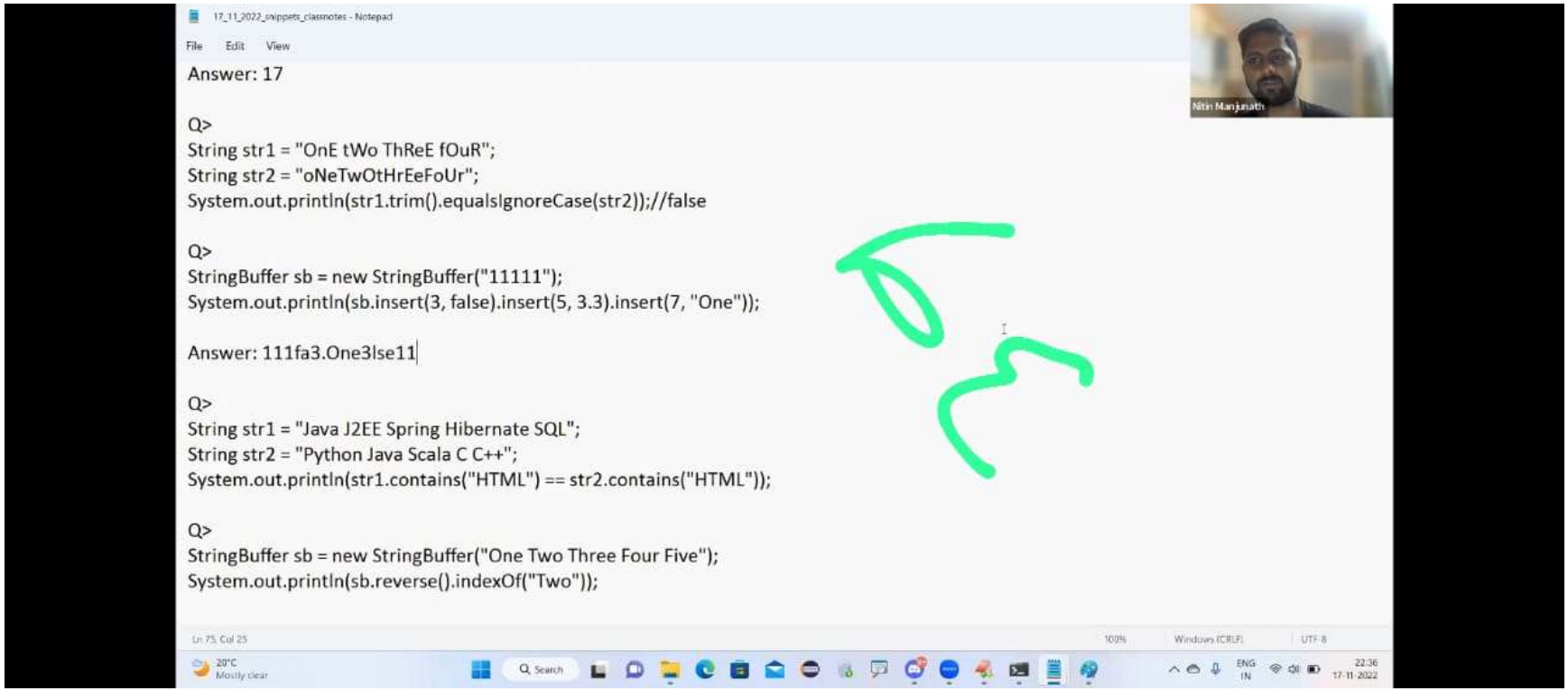

17_11_2022_snippets_classnotes - Notepad
File Edit View
Answer: -1
Q>
String s1 = "1";
System.out.println(s1.concat("2").concat("3"));
Answer: 123
Q>
String s1 = "ONE";
s1.concat("TWO");
s1.concat("THREE");
System.out.println(s1);
Answer: ONE
Nitin Manjunath
Ln 27, Col 10
100%
Windows (CRLF)
UTF-8
20°C
Mostly clear
Search
ENG
IN
22:18
17-11-2022

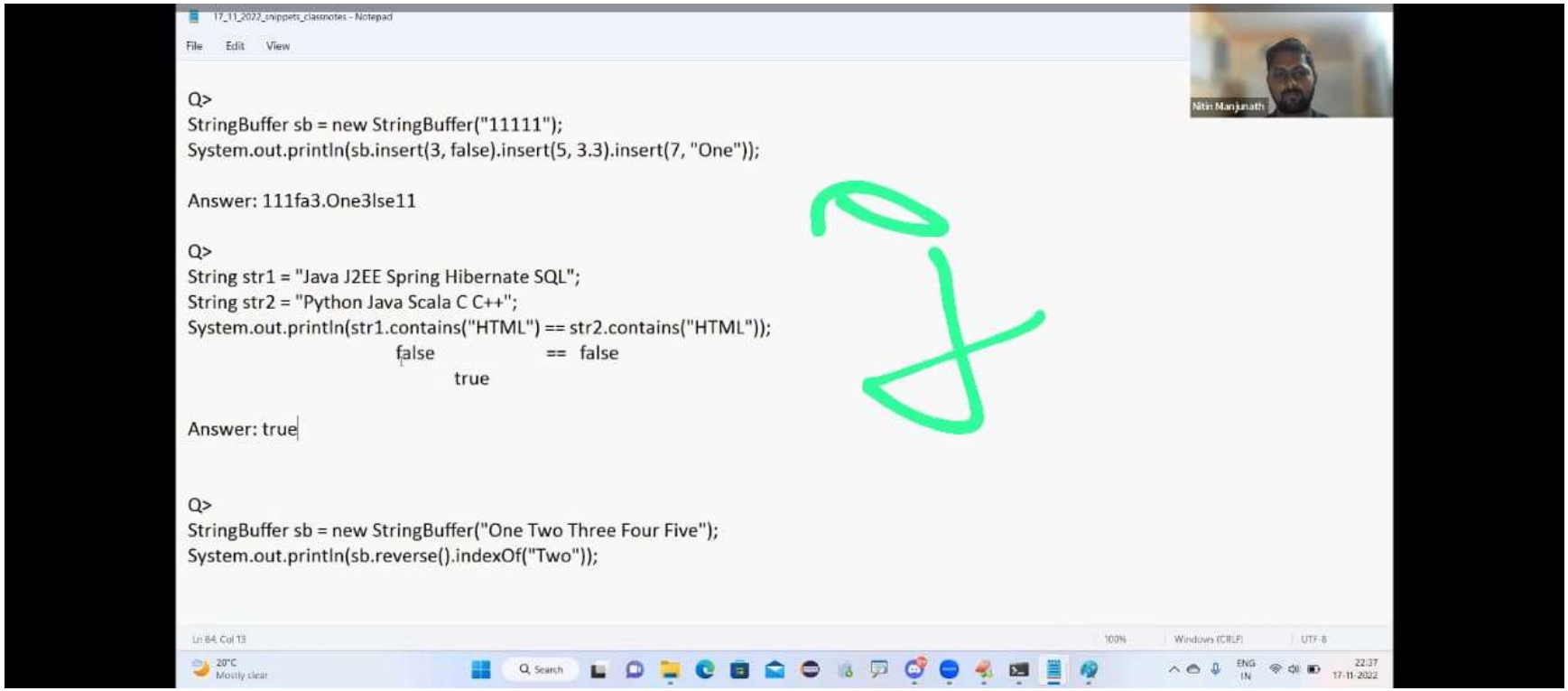

Untitled - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
Consolas
14
Background fill
String str1 = "Java";
String str2 = str1.intern();
String str3 = new String("Java");
System.out.println(str1 == str2);//true
System.out.println(str2 == str3);//false
System.out.println(str3 == str1);//false
SCP
HeapArea
str1
str2
Java
str3
Java
526, 288px
1 × 1px
3490 × 3830px
100%
20°C
Mostly clear
Search
ENG
IN
22:21
17-11-2022

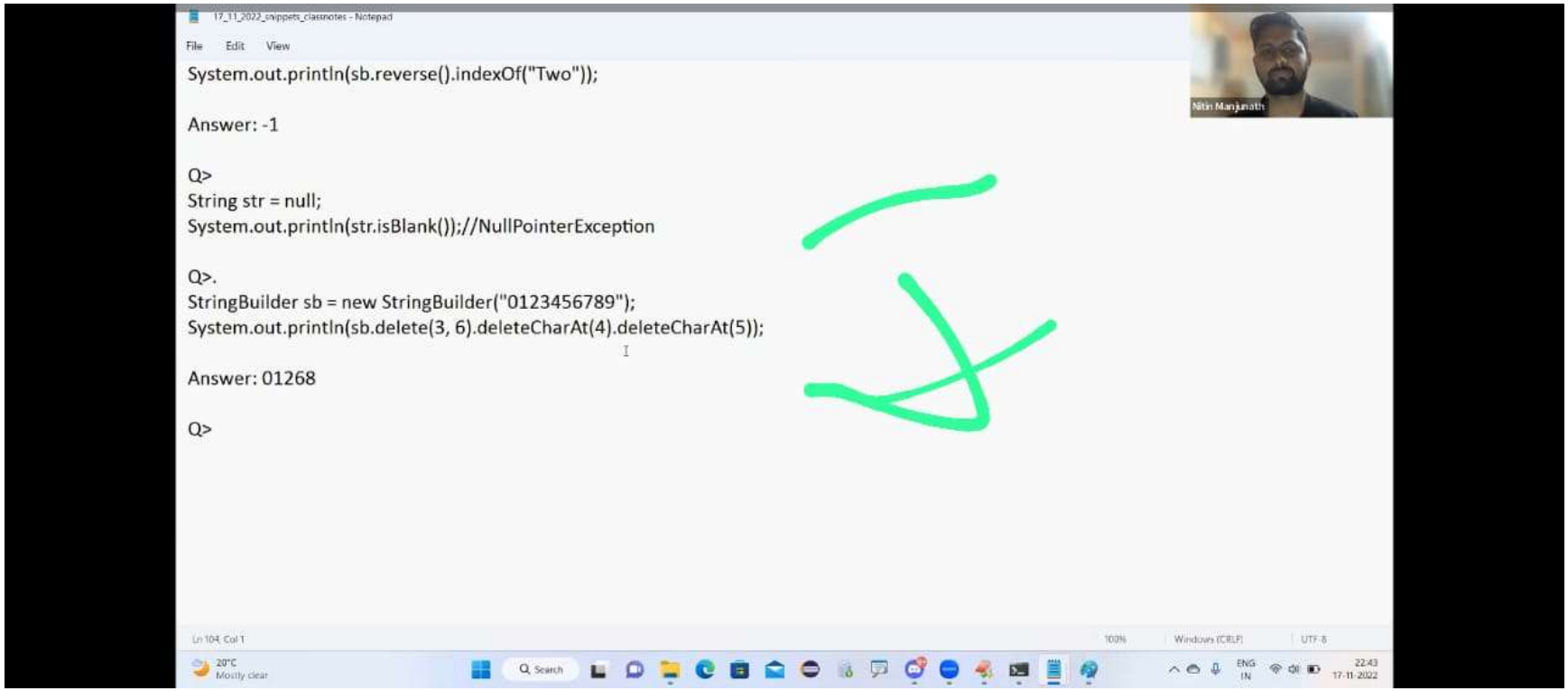

17_11_2022_snippets_explanation - Paint
File
View
Clipboard
Image
Tools
Brushes
Shapes
Size
Colours
Nitin Manjunath
Consolas
14
Background fill
SCP
HeapArea
String str1 = "Java";
String str2 = str1.intern();
System.out.println(str1 == str2);//true
System.out.println(str1.equals(str2));
//true
str1
str2
Java
439, 857px
3490 × 3830px
Size: 81.1KB
100%
20°C
Mostly clear
Search
ENG
IN
22:24
17-11-2022

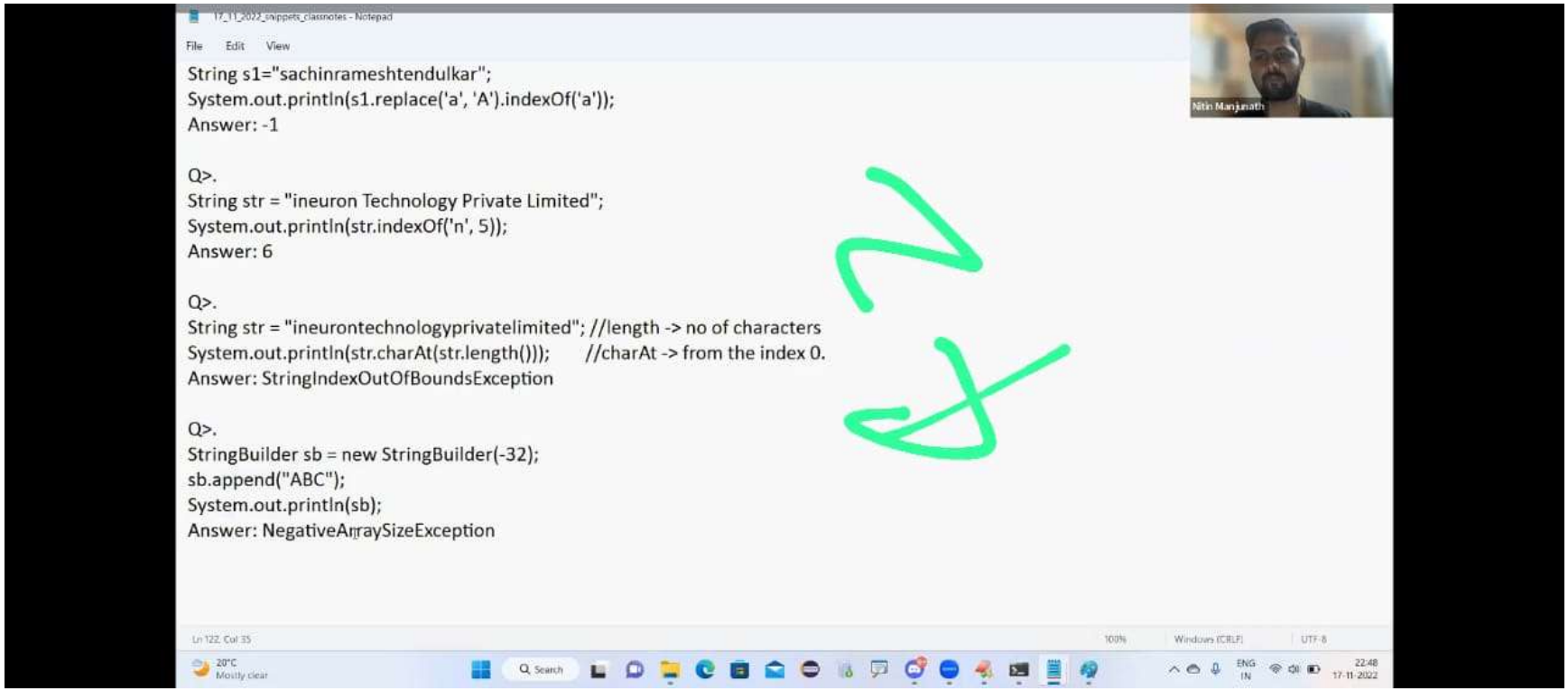

17_11_2022_snippets_classnotes - Notepad
File Edit View
Answer: true,true
Q>
StringBuffer sb1 = new StringBuffer("11111");
StringBuffer sb2 = sb1.append(22222).reverse();// sb1 ,sb2 =2222211111
System.out.println(sb1 == sb2);//true
Q>
StringBuffer and StringBuilder classes have intern() method. True or False?
answer: False
Q>
System.out.println(String.join(",", "1", "2", "3").concat(",").repeat(3).lastIndexOf(','));
output:: 1,2,3,1,2,3,1,2,3,
Answer: 17
Q>
Nitin Manjunath


EditPlus - [D:\TestApp.java *]
File Edit View Search Document Project Tools Browser Format Window Help
Directory Cliptext Functions
[D:] New Volume
D:\
$RECYCLE.BIN
Corejavapractise
Enterprisejava
enterprisejavabatch
FileIOPrograms
Front end videos
github
gitnotes
Test.java
TestApp.java
Java (*.java)
TestApp.java
For Help, press F1
Nitin Manjunath
1 2 3 4 5 6 7
1
2 public class TestApp {
3 public static void main(String[] args){
4 StringBuffer sb = new StringBuffer("11111");
5 System.out.println(sb.insert(3, false).insert(5, 3.3).insert(7, "One"));
6
7 //111fa3.One3lse11
8
9
10 }
11 }
12
ln 8 col 9 12 00 PC ANSI
20°C
Mostly clear
Search
ENG
IN
22:36
17-11-2022


17_11_2022_snippets_classnotes - Notepad
File Edit View
Answer: 17
Q>
String str1 = "OnE tWo ThReE fOuR";
String str2 = "oNeTwOtHrEeFoUr";
System.out.println(str1.trim().equalsIgnoreCase(str2));//false
Q>
StringBuffer sb = new StringBuffer("11111");
System.out.println(sb.insert(3, false).insert(5, 3.3).insert(7, "One"));
Answer: 111fa3.One3lse11
Q>
String str1 = "Java J2EE Spring Hibernate SQL";
String str2 = "Python Java Scala C C++";
System.out.println(str1.contains("HTML") == str2.contains("HTML"));
Q>
StringBuffer sb = new StringBuffer("One Two Three Four Five");
System.out.println(sb.reverse().indexOf("Two"));
Ln 75, Col 25
100%
Windows (CRLF)
UTF-8
Nitin Manjunath
20°C
Mostly clear
Search
ENG
IN
22:36
17-11-2022


17_11_2022_snippets_classnotes - Notepad
File Edit View
Nitin Manjunath
Q>
StringBuffer sb = new StringBuffer("11111");
System.out.println(sb.insert(3, false).insert(5, 3.3).insert(7, "One"));
Answer: 111fa3.One3lse11
Q>
String str1 = "Java J2EE Spring Hibernate SQL";
String str2 = "Python Java Scala C C++";
System.out.println(str1.contains("HTML") == str2.contains("HTML"));
false == false
true
Answer: true
Q>
StringBuffer sb = new StringBuffer("One Two Three Four Five");
System.out.println(sb.reverse().indexOf("Two"));
Ln 84, Col 13
100%
Windows (CRLF)
UTF-8
20°C
Mostly clear
Search
ENG
IN
22:37
17-11-2022


17_11_2022_snippets_classnotes - Notepad
File Edit View
System.out.println(sb.reverse().indexOf("Two"));
Answer: -1
Q>
String str = null;
System.out.println(str.isBlank());//NullPointerException
Q>.
StringBuilder sb = new StringBuilder("0123456789");
System.out.println(sb.delete(3, 6).deleteCharAt(4).deleteCharAt(5));
Answer: 01268
Q>
Nitin Manjunath
Ln 104, Col 1
100%
Windows (CRLF)
UTF-8
20°C
Mostly clear
Search
ENG
IN
22:43
17-11-2022


17_11_2022_snippets_classnotes - Notepad
File Edit View
String s1="sachinrameshtendulkar";
System.out.println(s1.replace('a', 'A').indexOf('a'));
Answer: -1
Q>.
String str = "ineuron Technology Private Limited";
System.out.println(str.indexOf('n', 5));
Answer: 6
Q>.
String str = "ineurontechnologyprivatelimited"; //length -> no of characters
System.out.println(str.charAt(str.length())); //charAt -> from the index 0.
Answer: StringIndexOutOfBoundsException
Q>.
StringBuilder sb = new StringBuilder(-32);
sb.append("ABC");
System.out.println(sb);
Answer: NegativeArraySizeException
Nitin Manjunath
Ln 122, Col 35
100%
Windows (CRLF)
UTF-8
20°C
Mostly clear
Search
ENG
IN
22:48
17-11-2022