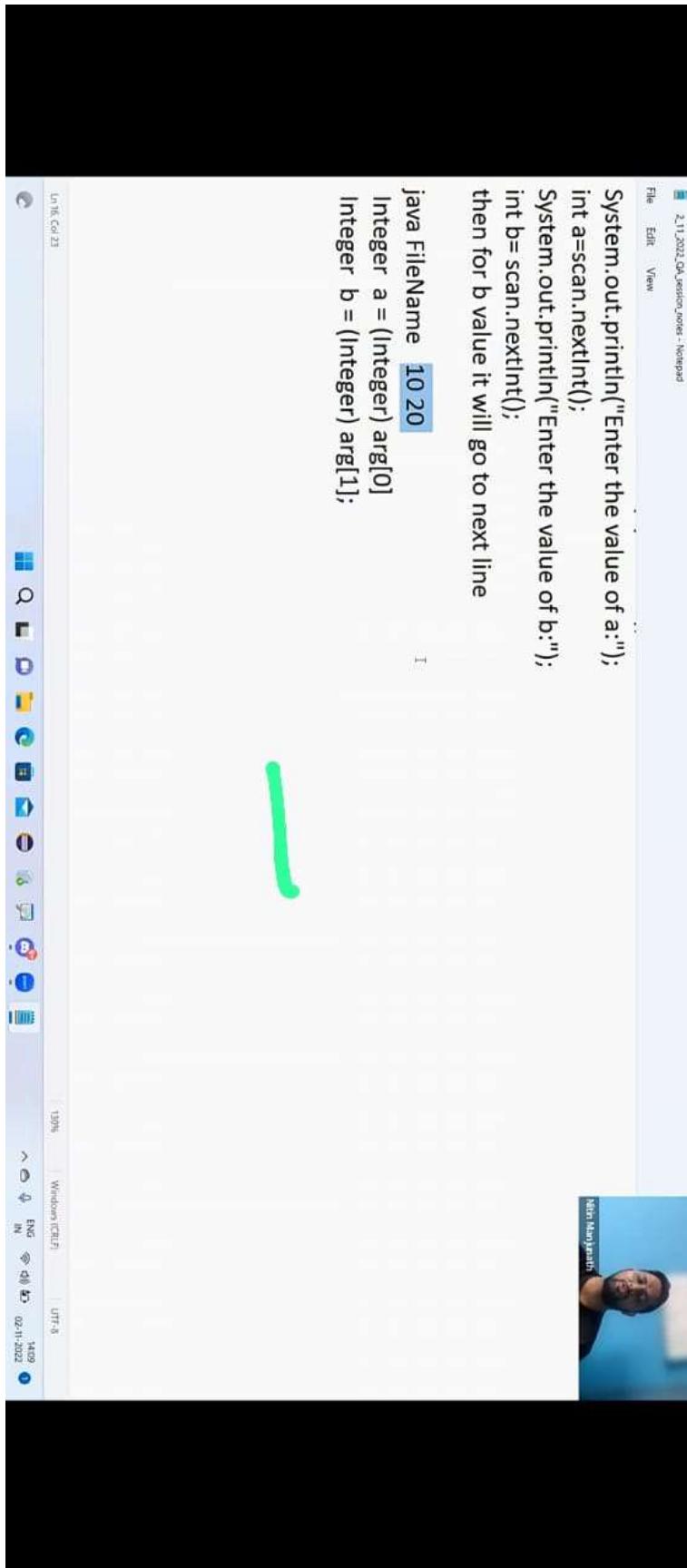


Java String Part2



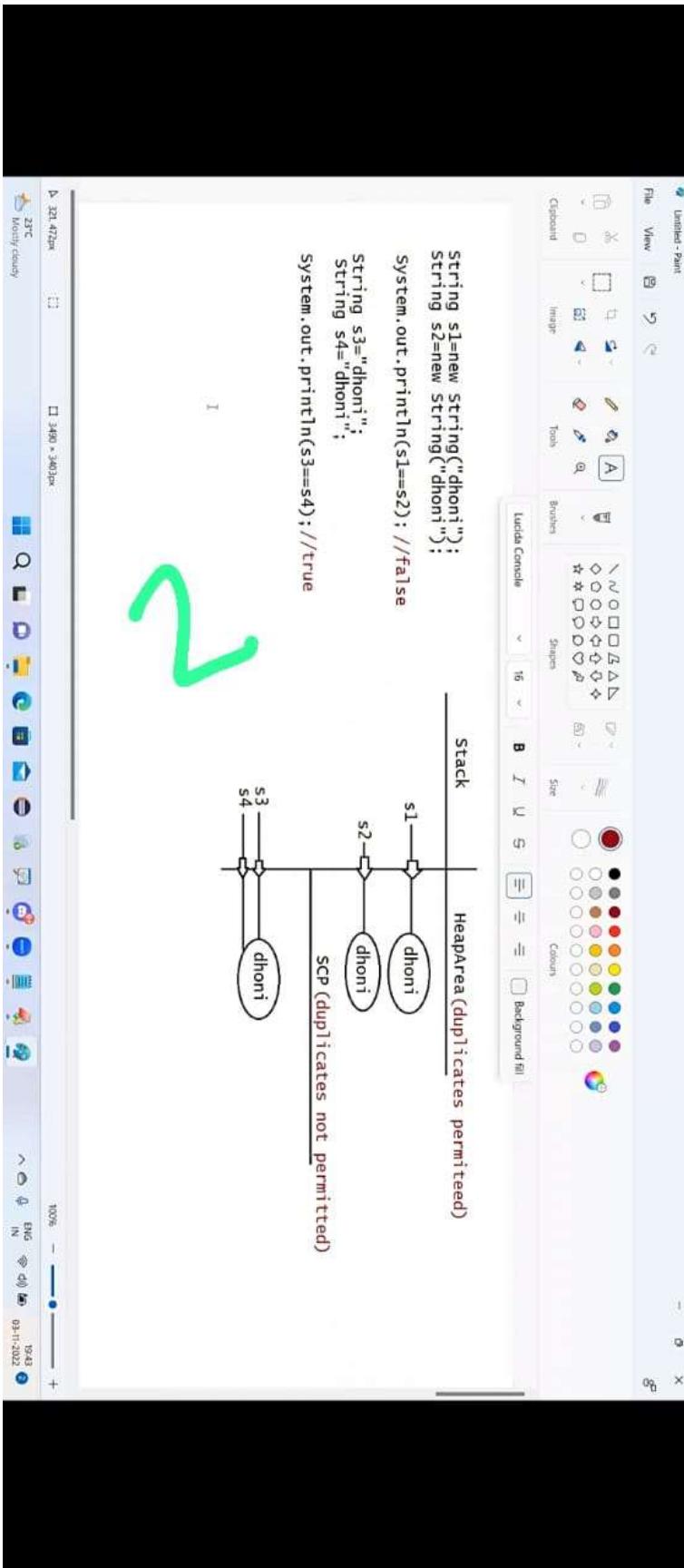
A screenshot of a Windows Notepad window titled "2.11.2022 OA-session_notes - Notepad". The window contains the following Java code:

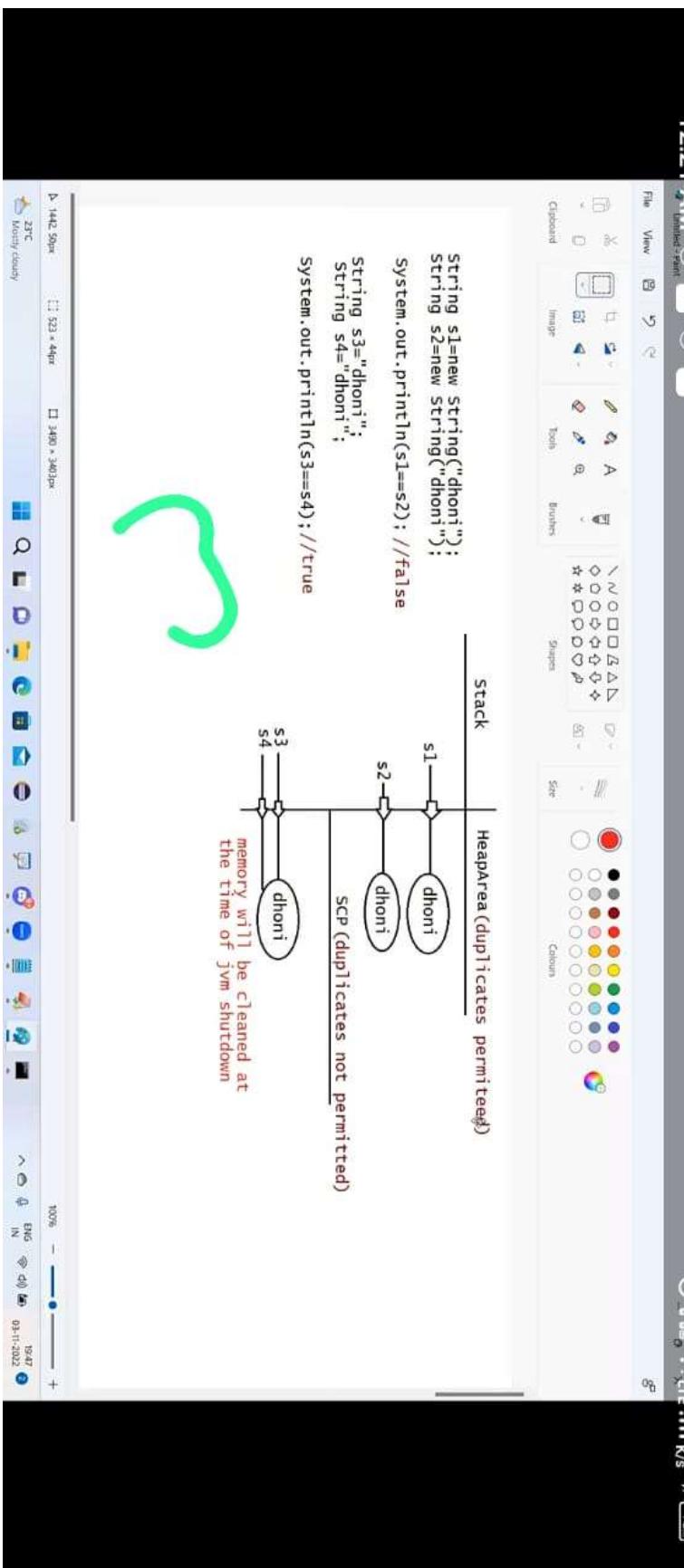
```
java FileName 10 20
Integer a = (Integer) arg[0]
Integer b = (Integer) arg[1];
```

Below the code, a green handwritten mark is present. The Notepad interface shows standard menu options (File, Edit, View), status bar information (Ln 16 Col 23, 15%, Windows-CE(UTF-8), UTF-8, ENG-IN, 14:09, 02-11-2022), and a toolbar with various icons.

System.out.println("Enter the value of a:");
int a=scan.nextInt();
System.out.println("Enter the value of b.");
int b= scan.nextInt();
then for b value it will go to next line

Nitin Mangalath







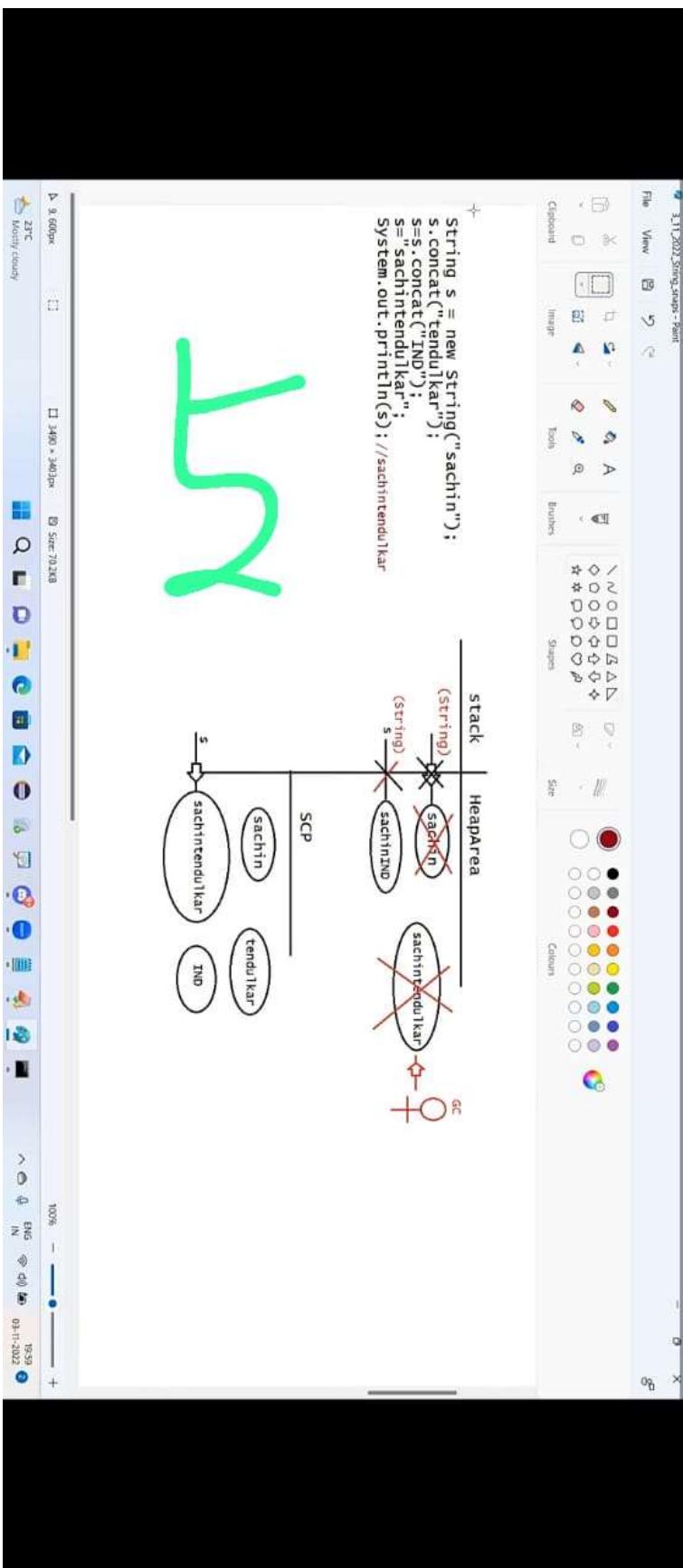
```
3.11.2022.String.println - Notepad  
File Edit View  
ln 15 Col 77 100% Windows (CEP) UTF-8  
23°C ENG IN 03-11-2022 19:48  
Mostly cloudy
```

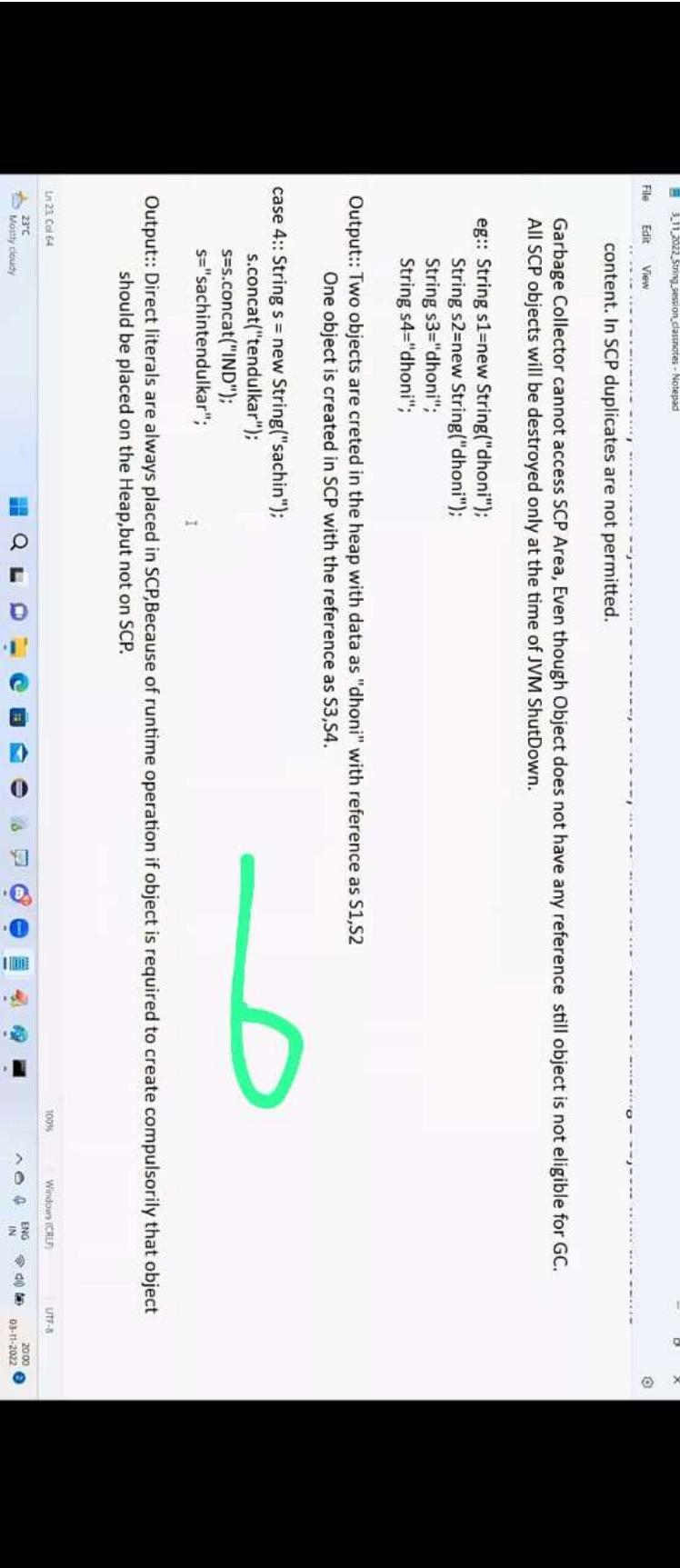
Note:: Object creation in SCP is always optional, 1st jvm will check if any object already created with required content or not.
If it is already available then it will reuse the existing object instead of creating the new Object.
If it is not available only then new object will be created, so we say in SCP there is no chance of existing 2 objects with the same content. In SCP duplicates are not permitted.

Garbage Collector cannot access SCP Area. Even though Object does not have any reference still object is not eligible for GC.
All SCP objects will be destroyed only at the time of JVM ShutDown.

eg:: String s1=new String("dhoni");
String s2=new String("dhoni");
String s3="dhoni";
String s4="dhoni";

Output:: Two objects are created in the heap with data as "dhoni" with reference as S1,S2
One object is created in SCP with the reference as S3,S4.





3.11.2022.String lesson - Notepad

File Edit View

```
content. In SCP duplicates are not permitted.
```

Garbage Collector cannot access SCP Area, Even though Object does not have any reference still object is not eligible for GC.

All SCP objects will be destroyed only at the time of JVM ShutDown.

```
eg:: String s1=new String("dhoni");
String s2=new String("dhoni");
String s3="dhoni";
String s4="dhoni";
```

Output:: Two objects are created in the heap with data as "dhoni" with reference as S1,S2.

One object is created in SCP with the reference as S3,S4.

```
case 4:: String s = new String("sachin");
s.concat("tendulkar");
s=s.concat("IND");
s="sachintendulkar";
```

Output:: Direct literals are always placed in SCP,Because of runtime operation if object is required to create compulsorily that object should be placed on the Heap,but not on SCP.

Ln 21 Col 14



100% Windows (CEP)

UTF-8

ENG IN 03-11-2022 20:00

23°C
Mostly cloudy

3.11.2022 String-strings - Paint

File View

Clipboard Image Tools Brushes Shapes Size Colors

Lucida Console 12 B I U G [=] [=] [=] Background fill

stack

Heapsarea

```

String s1= new String("sachin");
s1.concat("tendulkar");
s1+="IND"; s1 = s1 + "IND";
String s2=s1.concat("MIT");
System.out.println(s1); //sachintendulkar
System.out.println(s2); //sachintendulkarIND

String +
    \n
addition
both operands are number type
addition
if one operand is string then concatenation

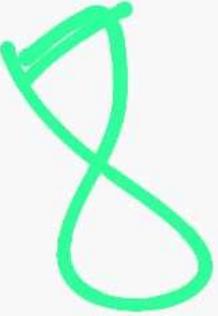
```

100%

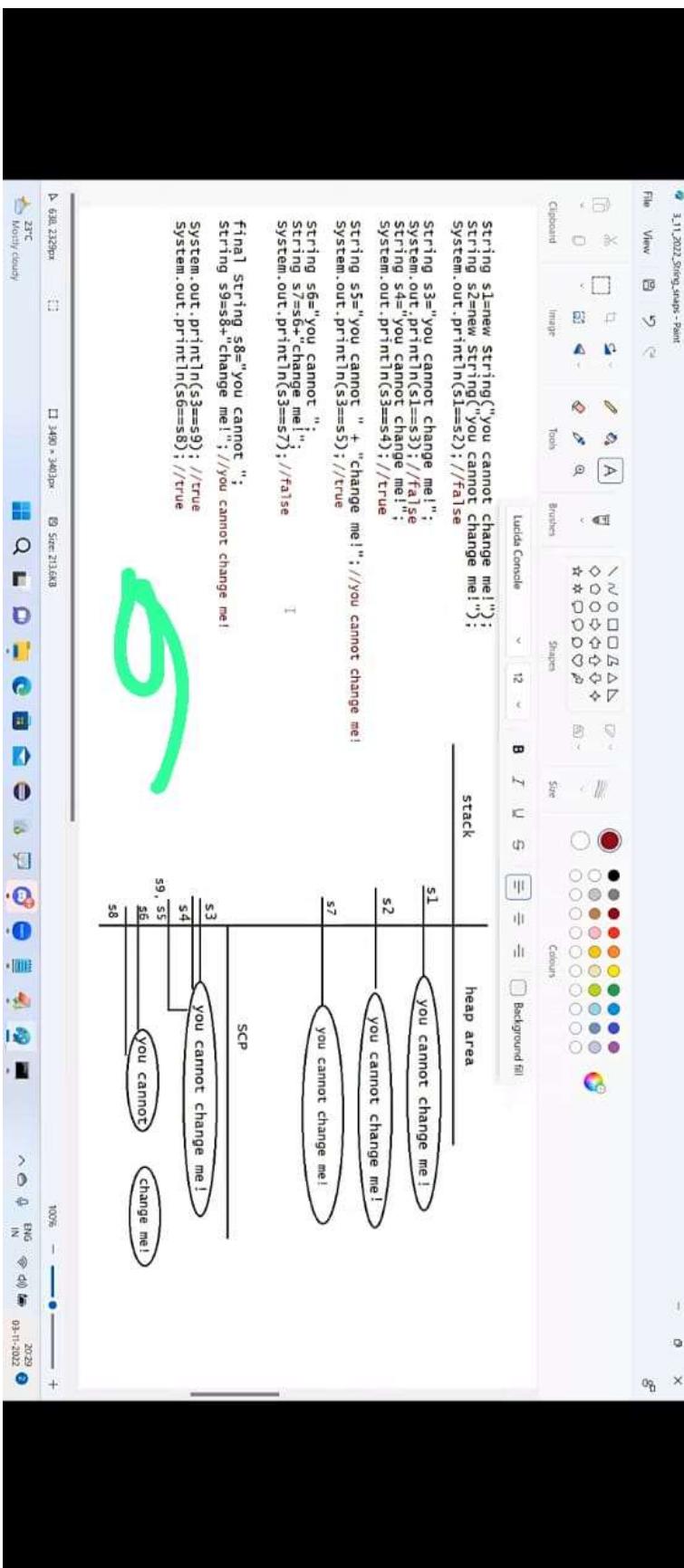
23°C Mostly cloudy

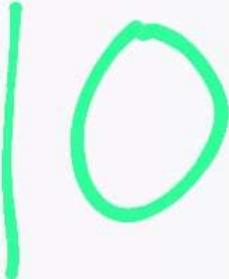
480 x 60px 3490 x 1403px Size:70.25%

Q ENG IN 03-11-2022



```
3.11.2022_String_session0_Glassnotes - Notepad  
File Edit View  
Q>  
String s1=new String("you cannot change me!")  
String s2=new String("you cannot change me!");  
System.out.println(s1==s2);  
  
String s3="you cannot change me!";  
System.out.println(s1==s3);  
String s4="you cannot change me!";  
System.out.println(s3==s4);  
  
String s5="you cannot " + "change me!";  
System.out.println(s3==s5);  
  
String s6="you cannot ";  
String s7=s6+"change me!";  
System.out.println(s3==s7);  
  
final String s8="you cannot ";  
String s9=s8+"change me!";  
System.out.println(s3==s9);  
System.out.println(s6==s8);  
  
L1 57 col 28 22°C Mostly cloudy Q & 20t5 ENG IN 03-11-2022 03:11:2022 Windows (C:\LT) 100% UTF-8
```





```
3.11.2022.StringSession.Characters - Notepad
File Edit View
String s6="You cannot ";
String s7=s6+"change me!";
System.out.println(s3==s7);

final String s8="you cannot ";
String s9=s8+"change me!";
System.out.println(s3==s9);
System.out.println(s6==s8);

output
false
false
true
true
false
true
true
true
```

Ln 66 Col 5 10% Windows-CE(F) UTF-8

23°C Mostly cloudy

03-11-2022 20:33

```
3.11.2022.String.intern().classnotes - Notepad  
File Edit View  
false  
false  
true  
true  
false  
true  
true  
true  
  
case7 :: Interning=> Using Heap object reference, if we want to get Corresponding SCP Object, then we need to use intern() method.  
  
eg1:  
String s1=new String("sachin");// One in heap(s1) and the other one in SCP  
String s2=s1.intern();// using s1 access object in SCP which has no reference  
System.out.println(s1==s2);//false  
String s3="sachin";  
System.out.println(s2==s3);//true
```

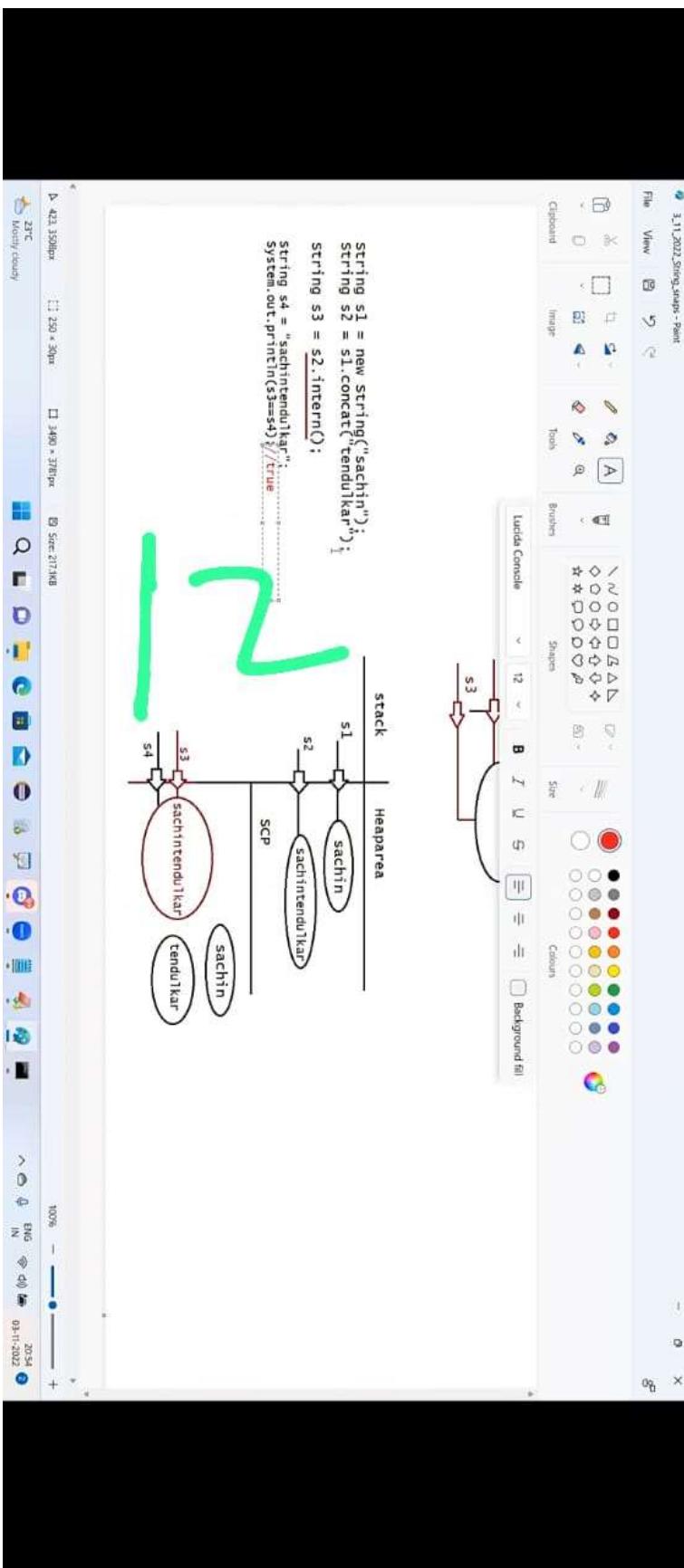
Ln 77 Col 1
23°C
Mostly cloudy

File Edit View
false
false
true
true
false
true
true
true

case7 :: Interning=> Using Heap object reference, if we want to get Corresponding SCP Object, then we need to use intern() method.

eg1:
String s1=new String("sachin");// One in heap(s1) and the other one in SCP
String s2=s1.intern();// using s1 access object in SCP which has no reference
System.out.println(s1==s2);//false
String s3="sachin";
System.out.println(s2==s3);//true

100% Windows (CET)
UTF-8
^ ◁ ← ENG ⇔ QD IN 03-11-2022 20:47





3.11 Java String class notes - Notepad

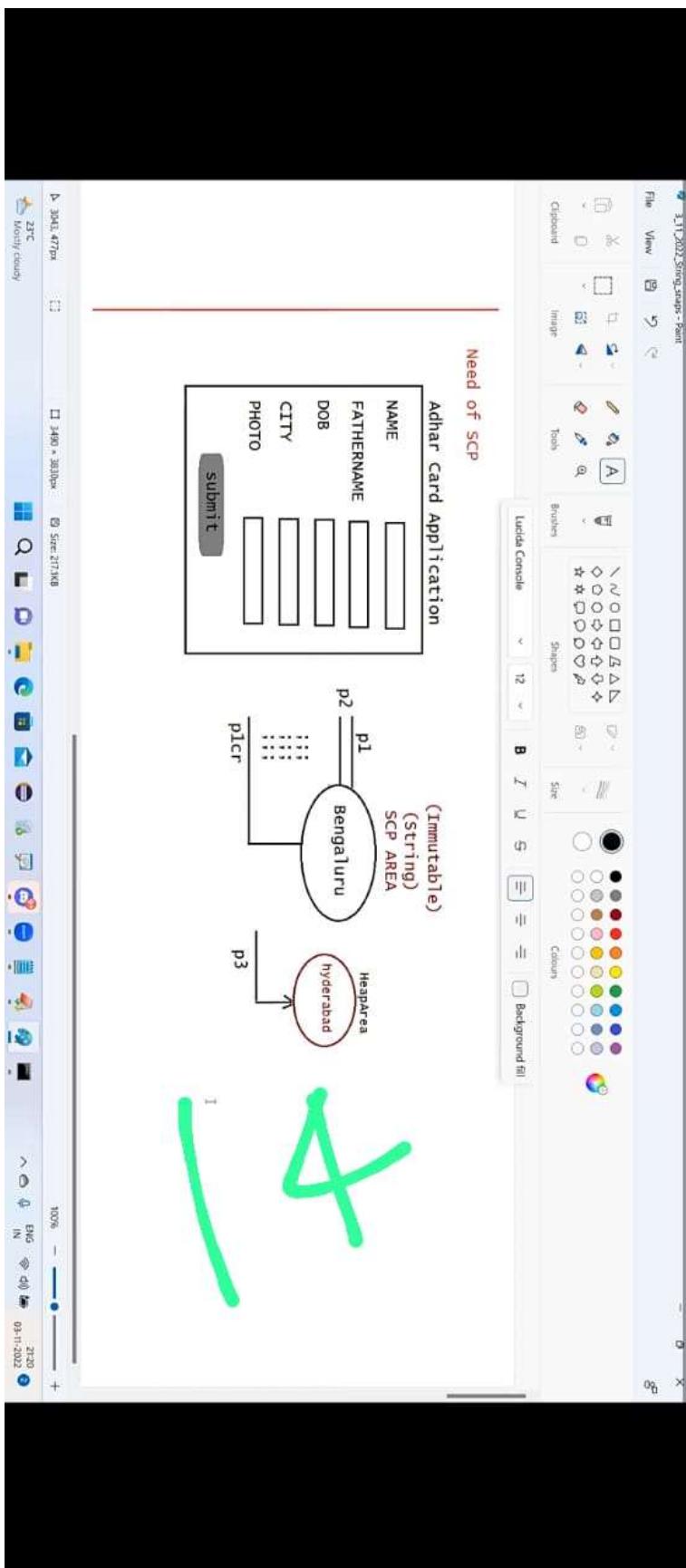
File Edit View

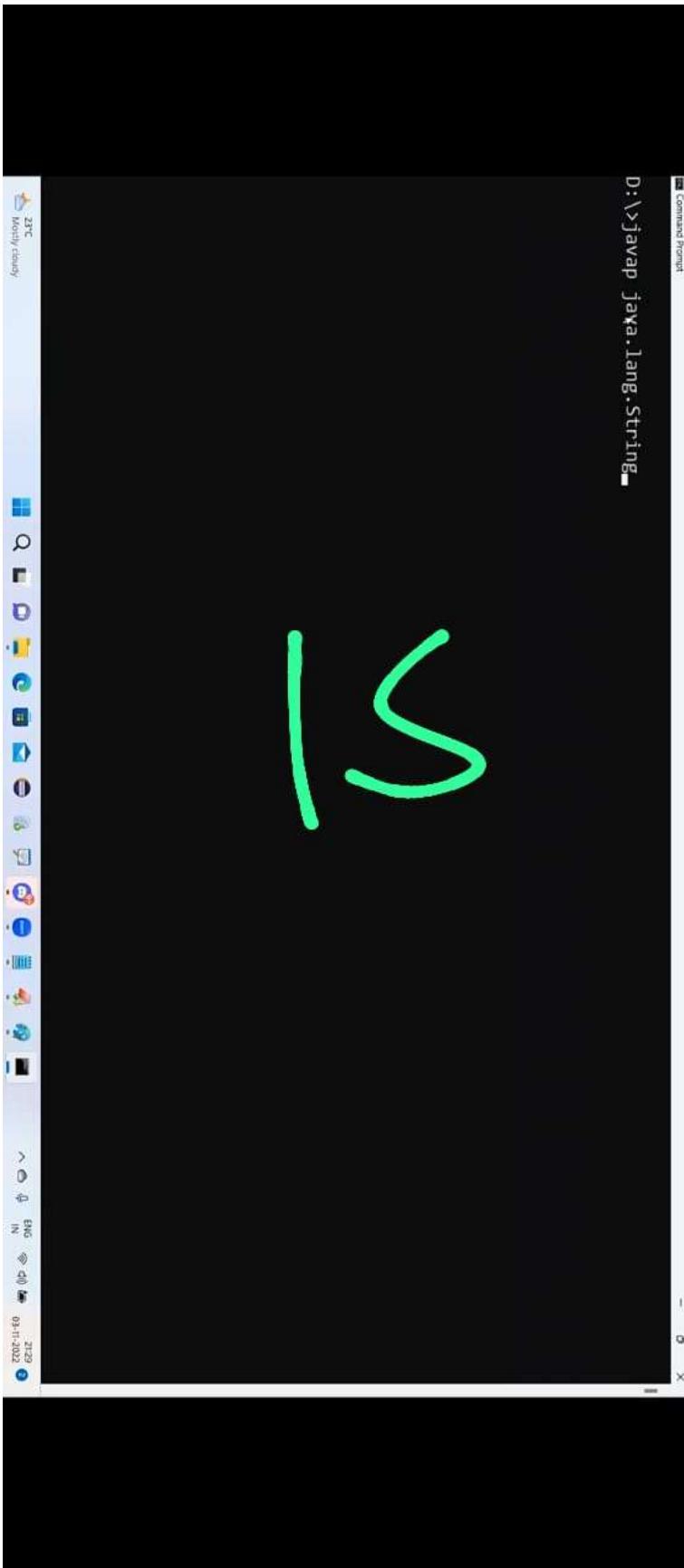
```
String s1=new String("sachin");// One in heap(s1) and the other one in SCP
String s2=s1.concat("IND");// One in SCP(IND) and the other one in heap(s2)
String s3=s2.intern();
String s4="sachinIND";
System.out.println(s3 == s4);//true
```

Importance of SCP

=====

1. In our program if any String object is required to use repeatedly then it is not recommended to create multiple object with same content
it reduces performance of the system and effects memory utilization.
2. We can create only one copy and we can reuse the same object for every requirement. This approach improves performance and memory utilization we can achieve this by using "scp".
3. In SCP several references pointing to same object the main disadvantage in this approach is by using one reference if we are performing any change the remaining references will be impacted. To overcome this problem sun people implemented immutability concept for String objects.
4. According to this once we creates a String object we can't perform any changes in the existing object if we are trying to perform any change a new String object will be created hence immutability is the main disadvantage of scp.





utilization we can achieve this by using "scp".

3. In SCP several references pointing to same object the main disadvantage in this approach is by using one reference if we are performing any change the remaining references will be impacted. To overcome this problem sun people implemented immutability concept for String objects.

4. According to this once we creates a String object we can't perform any changes in the existing object if we are trying to perform any change a new String object will be created hence immutability is the main disadvantage of scp.

String class Constructor

```
=====
String s =new String();           => Creates an Empty String Object
String s =new String(String literals) => Creates an Object with String literals on Heap
String s =new String(StringBuffer sb) => Creates an equivalent String object for StringBuffer
String s =new String(char[] ch)      => Creates an equivalent String object for character array
String s =new String(byte[] b)       => Creates an equivalent String object for byte array
```

16

In 103, Col 25
23°C
Mostly cloudy



100% Windows (C:\EFT) UTF-8

◎

-

×

◎

Editor - (Untitled)

File Edit View Search Document Project Tools Browser Ethernet Window Help

D:\ New Volume

1 D:\

2 SOURCE BIN

3 CodeSnippets

4 Examples

5 Front end video

6 GitHub

7 GitKraken

8 Microsoft

9 Test.java

```
1 public class Test {
2     public static void main(String[] args) {
3         char[] ch = {'j', 'a', 'v', 'a'};
4         String s1 = new String(ch);
5         System.out.println(s1); // java
6         System.out.println();
7         byte[] b = {65,66,67,68};
8         String s2= new String(b);
9         System.out.println(s2); // "ABCD"
10    }
11 }
12
13
14
15
16
17
18
19
20 }
21
22
```

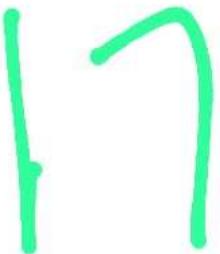
In 14 col 40 9 00 PC ANSI EVALUATION VERSION

Java (*.java)

* Test.java

For Help press F1

23°C
Mostly cloudy



File Edit View
19_11_2022_SimpSession_Glassware - Notepad

Note: If the class name and method name is same in a class then that method is called "Constructor".

String class Constructor

```
String s =new String();                                => Creates an Empty String Object
String s =new String(String literals)                => Creates an Object with String literals on Heap
    eg: String str = new String("sachin");
String s =new String(StringBuffer sb) => Creates an equivalent String object for StringBuffer
String s =new String(char[] ch)                      => Creates an equivalent String object for character array
String s =new String(byte[] b)                       => Creates an equivalent String object for byte array
```

eg:

```
char[] ch = {'j','a','v','a'};
String s1 =new String(ch);
System.out.println(s1); //java
```

```
System.out.println();
```

```
byte[] b = {65,66,67,68};
String s2= new String(b);
System.out.println(s2); //ABCD"
```

18

Ln 116 Col 1
23°C
Mostly cloudy



100% Windows (C:\EF7) UTF-8

EN IN

03-11-2022 21:41

```
19/11/2022 String session Class notes - Notepad  
File Edit View  
  
eg: String str = new String("sachin");  
String s =new String(StringBuffer sb) => Creates an equivalent String object for StringBuffer  
String s =new String(char[] ch) => Creates an equivalent String object for character array  
String s =new String(byte[] b) => Creates an equivalent String object for byte array  
  
eg:  
char[] ch = {'j','a','v','a'};  
String s1 =new String(ch);  
System.out.println(s1); //java  
System.out.println();  
  
byte[] b = {65,66,67,68};  
String s2= new String(b);  
System.out.println(s2); //ABCD  
  
StringBuffer sb =new StringBuffer("sachin");  
System.out.println("StringBuffer data is :: "+sb);  
  
String s1 =new String(sb);  
System.out.println("String data is : "+s1);  
  
19
```

3.11.2022 String session - charAt() - Notepad

File Edit View

Important methods of String

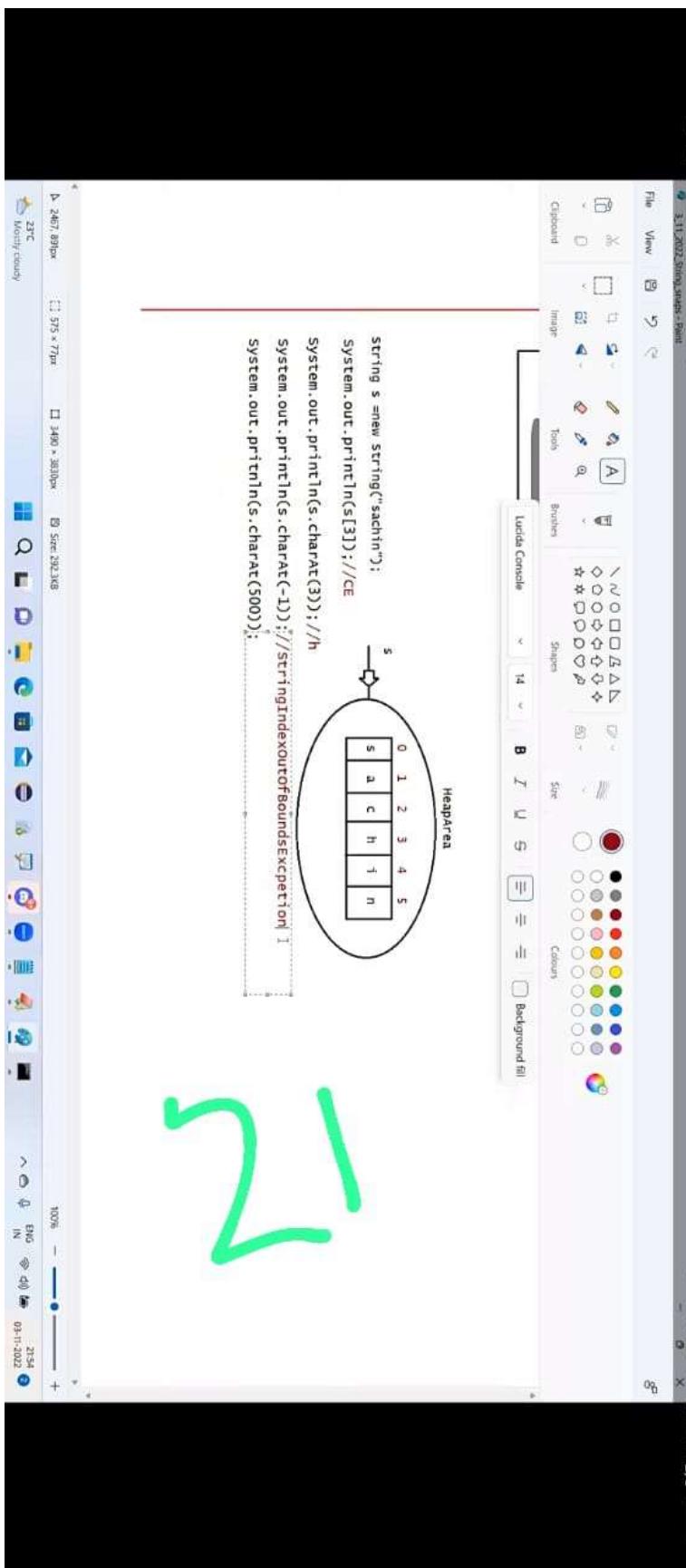
```
=====
1.public char charAt(int index)
2.public String concat(String str)
3.public boolean equals(Object o)
4.public boolean equalsIgnoreCase(String s)
5.public String substring(int begin)
6.public String substring(int begin,int end)
7.public int length()
8.public String replace(char old,char new)
9.public String toLowerCase()
10.public String toUpperCase()
11.public String trim()
12.public int indexOf(char ch)
13.public int lastIndexOf(char ch)
```

20

23°C
Mostly cloudy



194x107 100% Windows (CET) UTF-8 ENG IN 03-11-2022 21:44



File Edit View

Which among the following declaration is valid?

- ```
1. int[] a;
2. int a[];
3. int []a;
4. int[6] a;//can't specify the size
```

Predict the answer.

- A. 1,2,3  
B. 1,2,4  
C. 2,3,4  
D. None of the above

Answer: A

Q>  
1. int[] a;  
a=new int[5];  
2. int[] a =new int[5];

Both the declarations are they same?

B. no

23°C  
Mostly cloudy

A vertical column of small icons representing various applications and system functions, typical of a Windows taskbar or Start menu.

100% Windows (CHM) UTF

3.11.2022 imports, session, closures - Notepad

File Edit View

Q>

1. int[] a;  
a=new int[5];

2. int[] a =new int[5];

Both the declarations are they same?

A. yes  
B. no

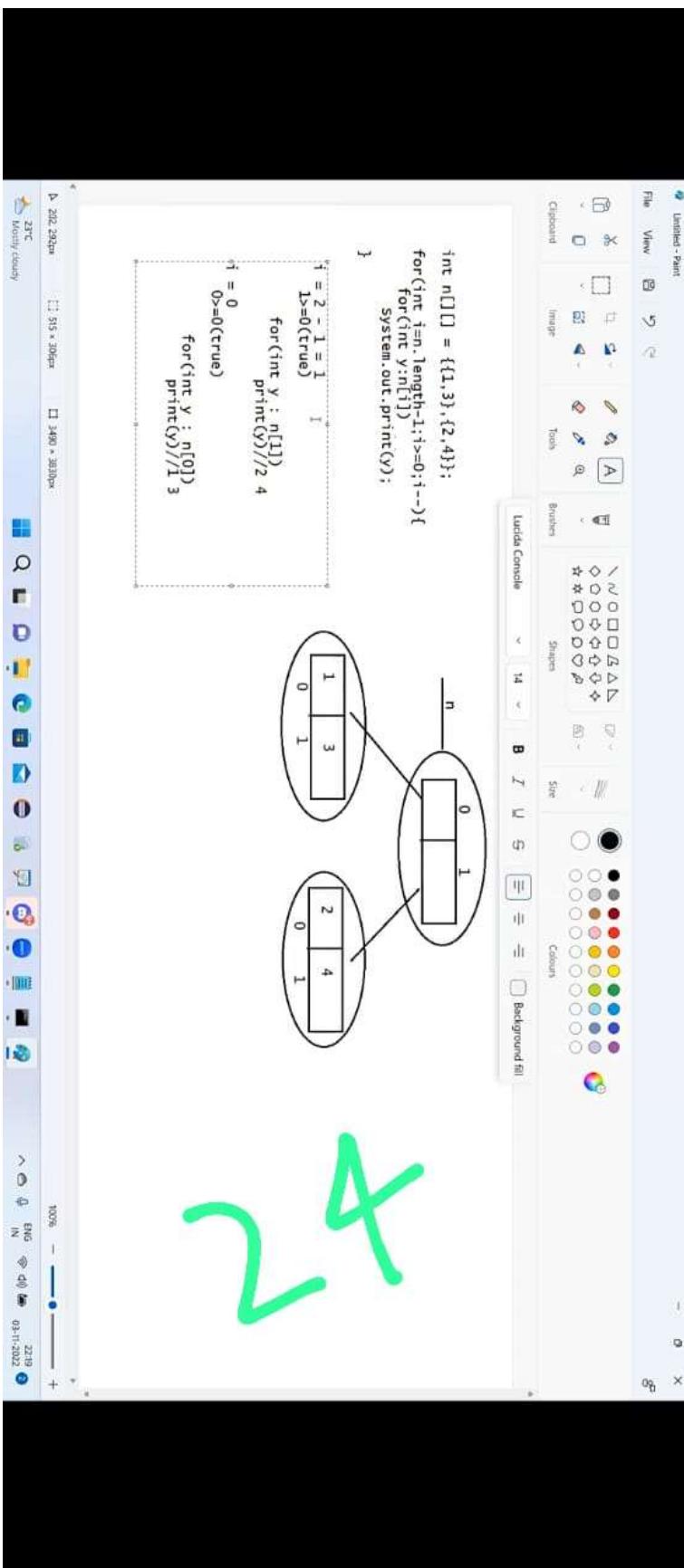
Answer: A

23

Ln 24 Col 10

100% Windows (CP1252) UTF-8

^ ◁ ◁ ENG IN ⇨ ⇧ 22:13 03-11-2022



3.11.2022.Snapshotsession.classnames - Notepad

File Edit View

Answer: A

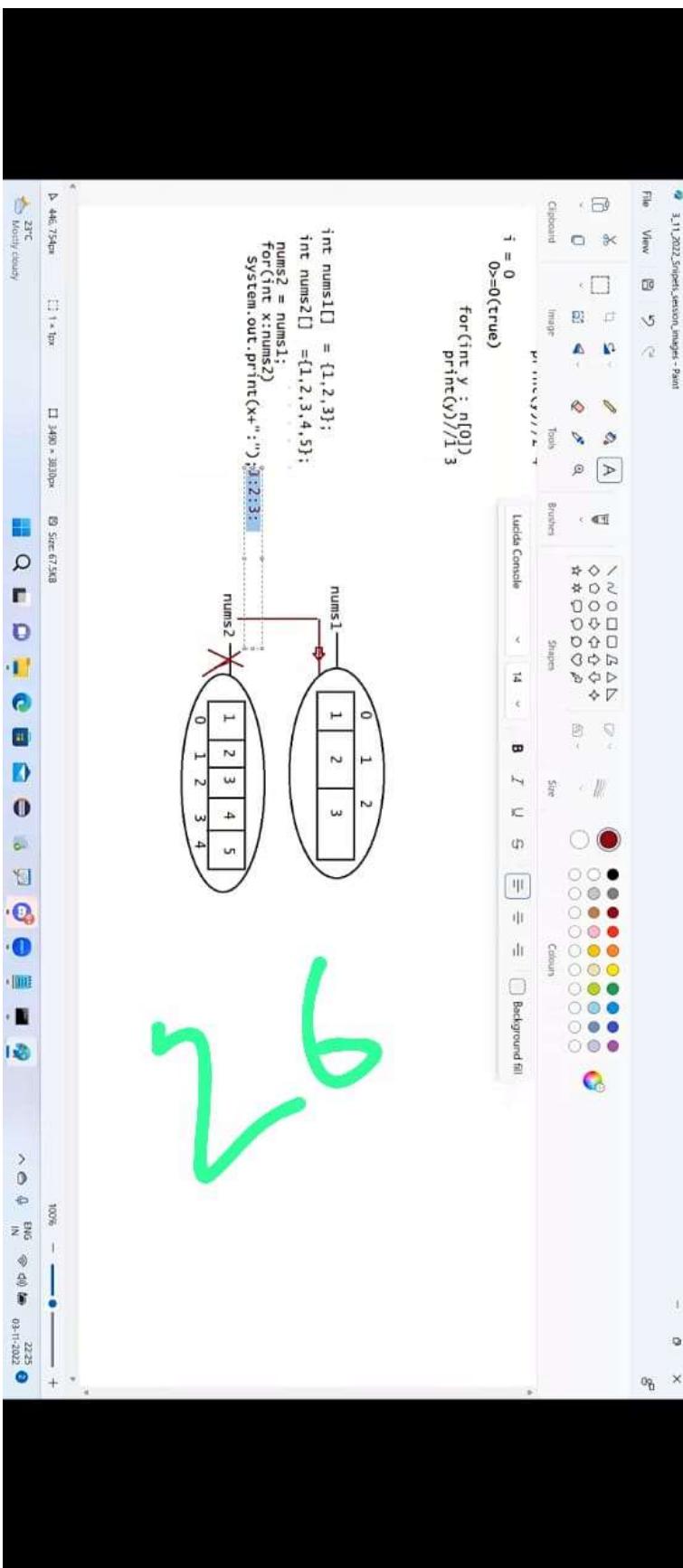
Q>

```
3. int n[] = {{1,3},{2,4}};
for(int i=n.length-1;i>=0;i-){
 for(int y:n[i])
 System.out.print(y);
}
```

A. 1234  
B. 2313  
C. 3142  
D. 4231  
E. 2413  
F. compilation error  
G. Some problem by JVM at the runtime

Answer: E

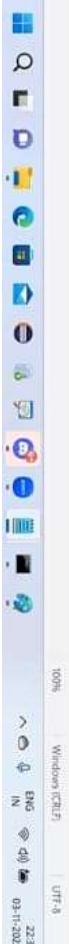
2 4 3 1



```
3.11.2022.Sympy,version,concretes - Notepad
File Edit View
count++;
}
System.out.println(count+" found");
What is the result?
A. Compilation fails
B. 0 found
C. 1 found
D. 3 found
E. 2 found
data[0]= 2010
data[1]= 2013
data[2]= 2014
data[3]= 2015
data[4]= 2014
key = 2014,count= 0
e=2010
Answer: B
```

27

Ln RZ\_Cal 97  
23°C  
Mostly cloudy



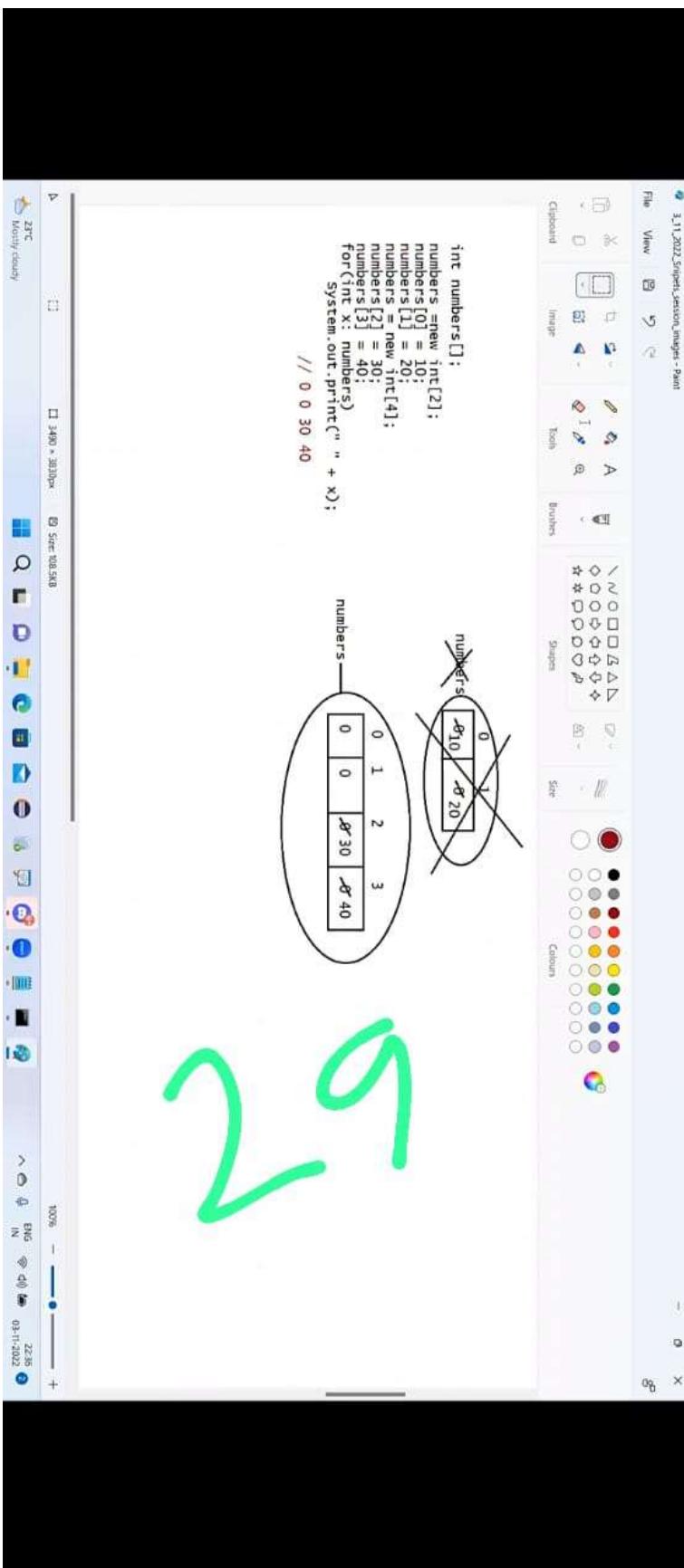
2014

```
key = 2014, count= 0
e=2010
```

Answer: B(at any time count value will not change becoz of continue, or even if condition fails)

3.11.2022 Snippets, version, differences - Notepad  
File Edit View  
E. 2 found

```
data[0]= 2010
data[1]= 2013
data[2]= 2014
data[3]= 2015
data[4]= 2014
```





```
13.11.2022 Snippets session 2 Resources - Notepad
File Edit View
class Test{
public static void main(String[] args){
 int numbers[];
 numbers = new int[2];
 numbers[0] = 10;
 numbers[1] = 20;
 numbers = new int[4];
 numbers[2] = 30;
 numbers[3] = 40;
 for(int x: numbers)
 System.out.print(" " + x);
}
}

What is the result?
A. 10 20 30 40
B. 0 0 30 40
C. Compilation fails
D. An exception is thrown at runtime
```

Answer: B

```
3.11.2022 Snippets, version, differences - Notepad
File Edit View
Q>
int wd = 0;
String days[] = {"sun", "mon", "wed", "sat"};
for(String s:days){
 switch(s){
 case "sat":
 case "sun":
 wd+=1;
 break;
 case "mon":
 wd++;
 case "wed":
 wd+=2;
 }
}

```

What is the result?

- A. 3
- B. 4
- C. -1
- D. compilation fails

JVM

Ln 134 Col 32  
23°C, Cloudy  
Mostly cloudy

3 /

3.11.2022 Snippets, version, classes - Notepad

File Edit View

```
JVM
===
days[0] = "sun", days[1] = "mon", days[2] = "wed", days[3] = "sat"
wd = 0, s = "sun", change wd = wd-1 = 0-1 = -1
wd = -1, s = "mon", wd = -1+1 = 0, wd = 0+2 = 2
wd = 2, s = "wed", wd = 4
wd = 4, s = "sat", wd = 4-1 = 3
```

}

What is the result?

A. 3

B. 4

C. -1

D. compilation fails

32

3.11.2022 Snippets lesson classes - Notepad

File Edit View

=> concat(String)

=> String object by default immutable(changes made will reflect in new memory)

```
String[] str = {"A", "B"};
int idx = 0;
for(String s: str){
 str[idx].concat(" element " + idx);
 idx++;
}
//using for loop
for(idx = 0;idx<str.length;idx++){
 System.out.print(str[idx]);
}
```

What is the result?

A. AB  
B. A elementB element1  
C. A NullPointerException is thrown at runtime  
D. A OB 1  
E. Compilation Error

Answer: A

1

Ln 153 Col 2      100%      Windows (CELT)      UTF-8

23°C      ENG ⇔ qd ENG      03-11-2022 22:49

Morbi cloudy

