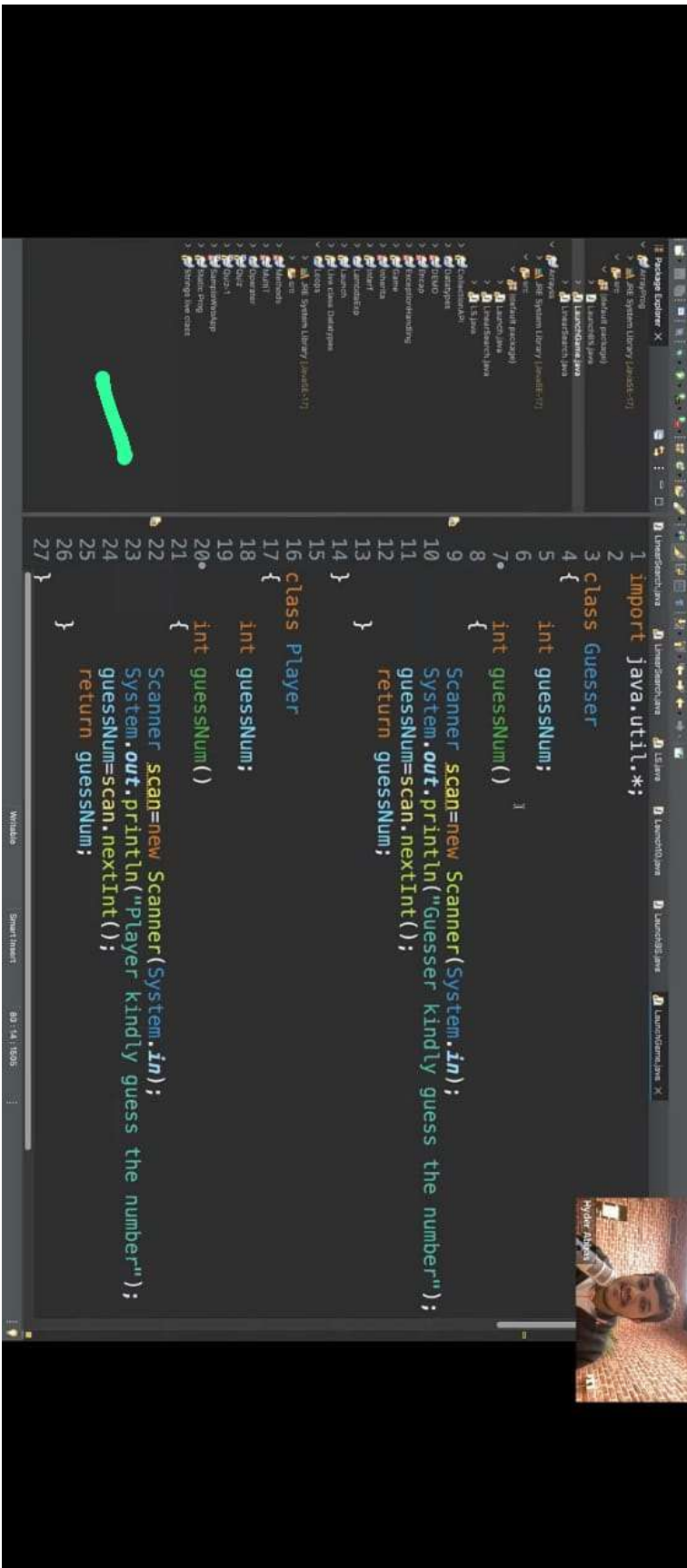
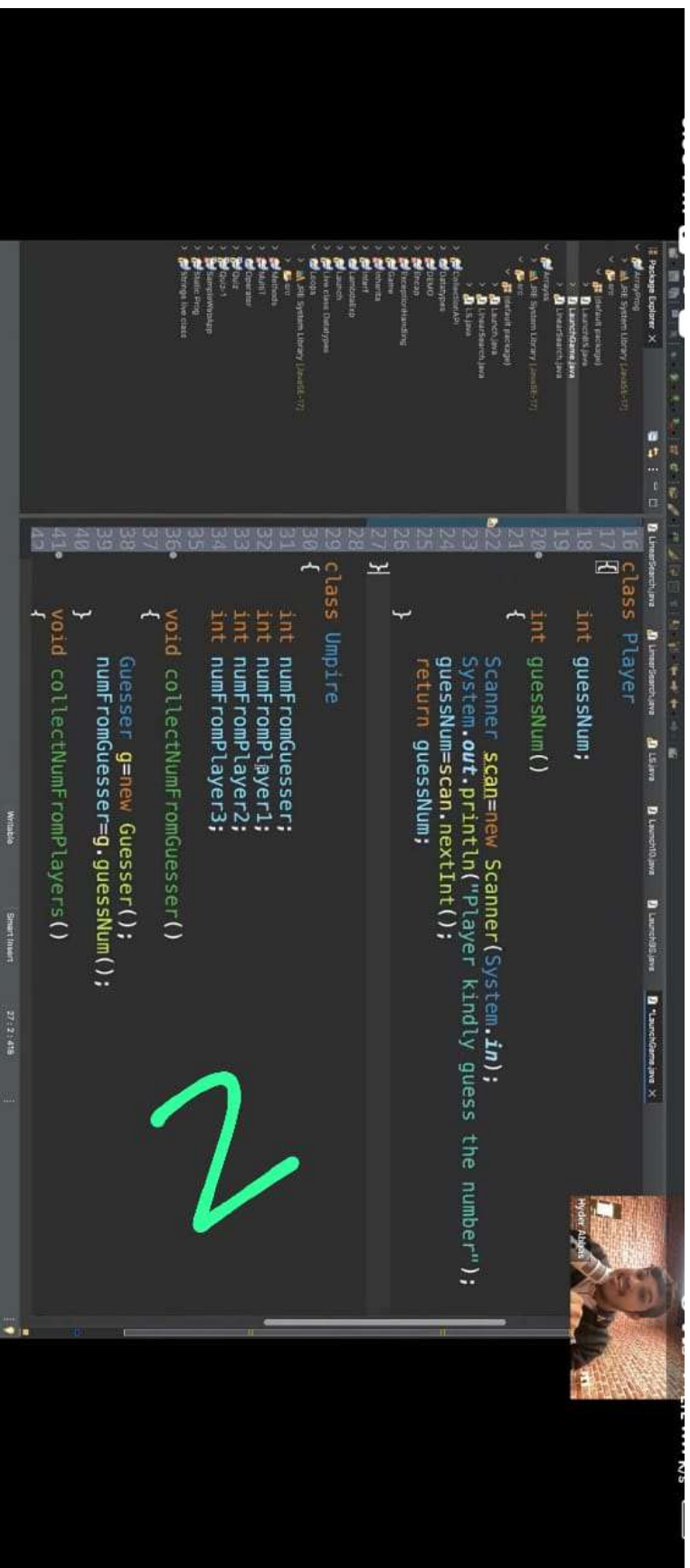
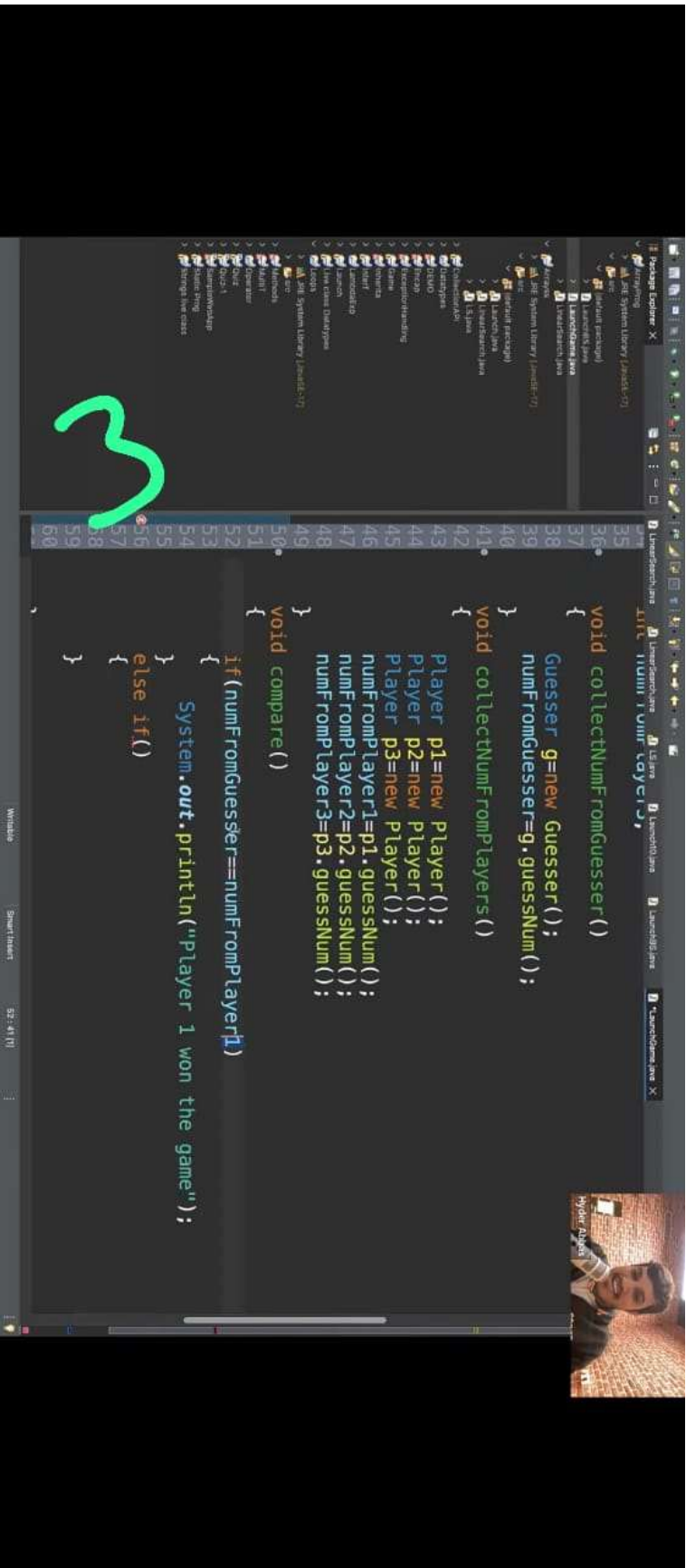
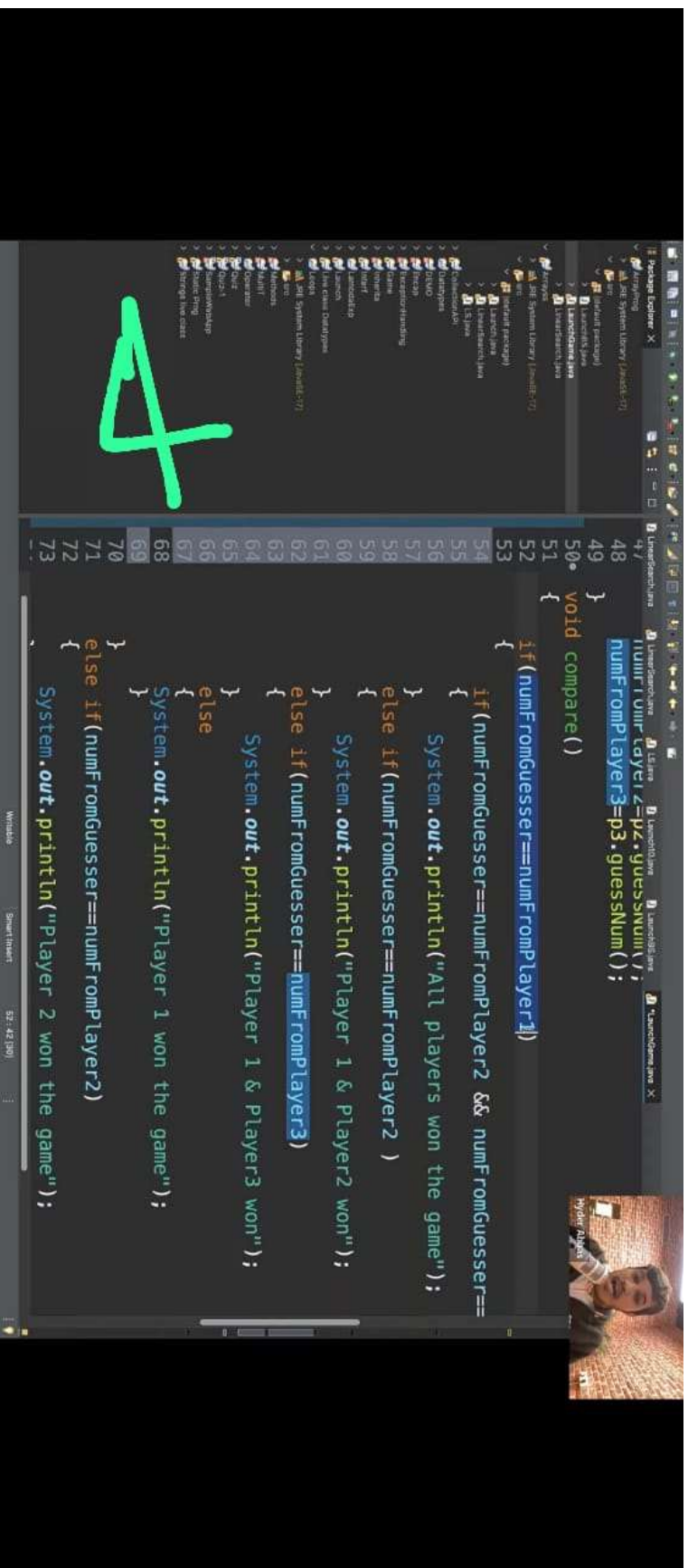


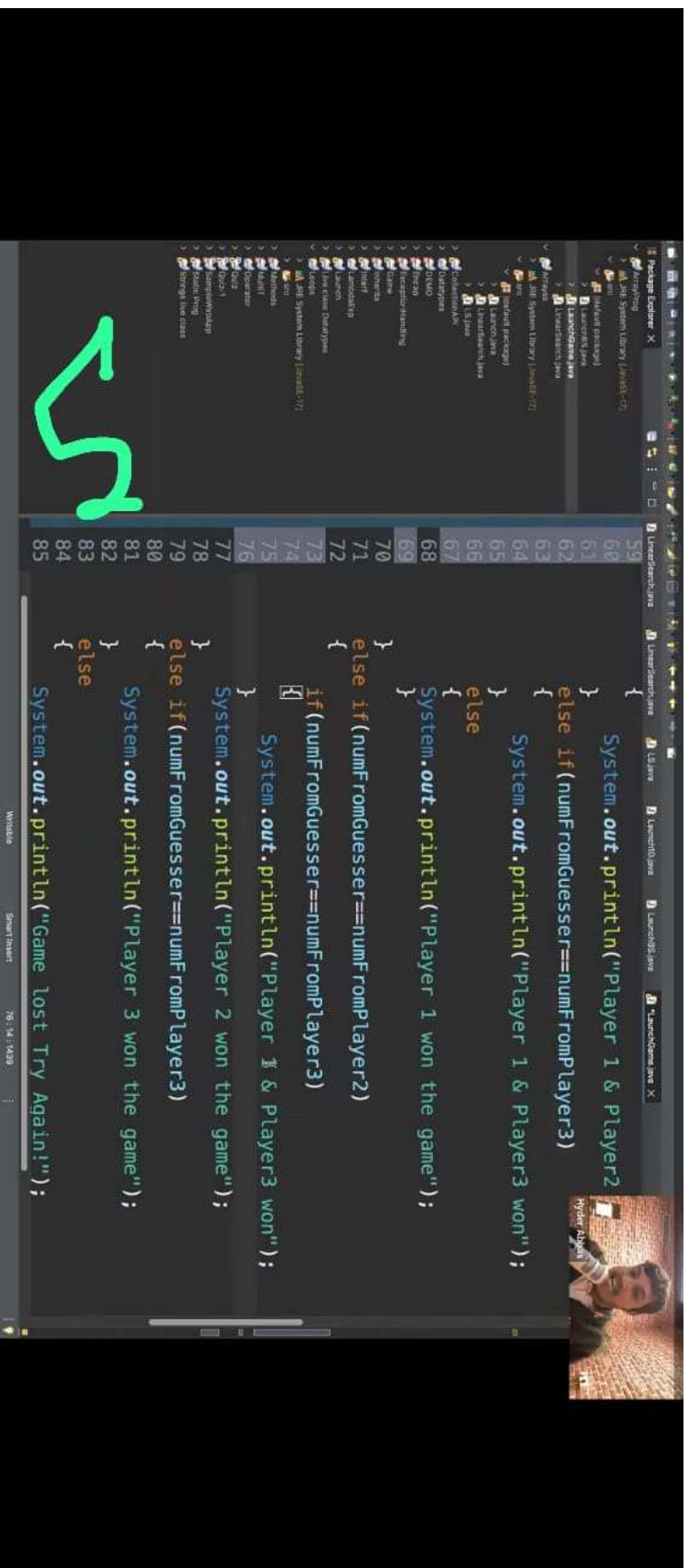
Java String Part1

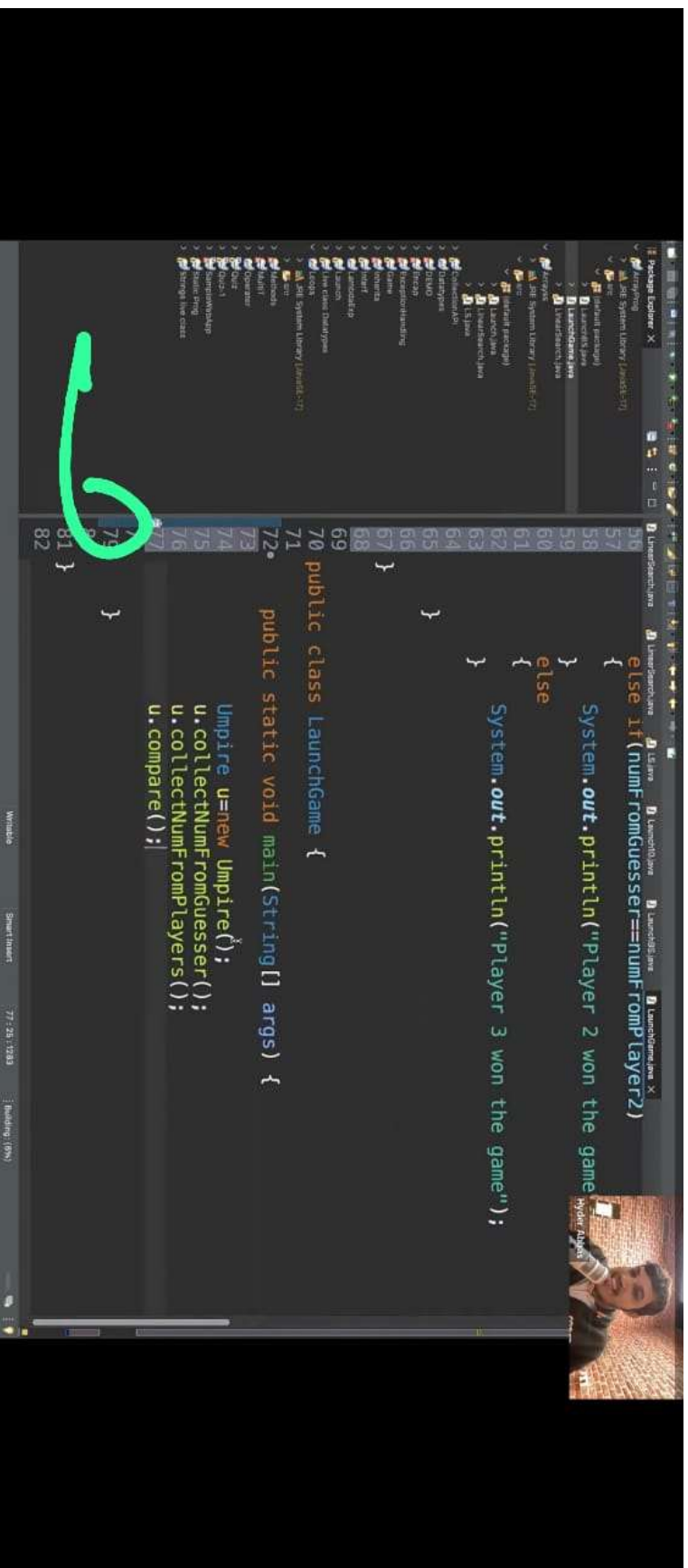












Untitled - Notepad
File Edit View

int => datatype, reserve word

String => It is classname, String is basically an inbuilt class for which object can be created.

package

java

|=> lang

|=> String



Ln 4, Col 8
20°C
Cloudy



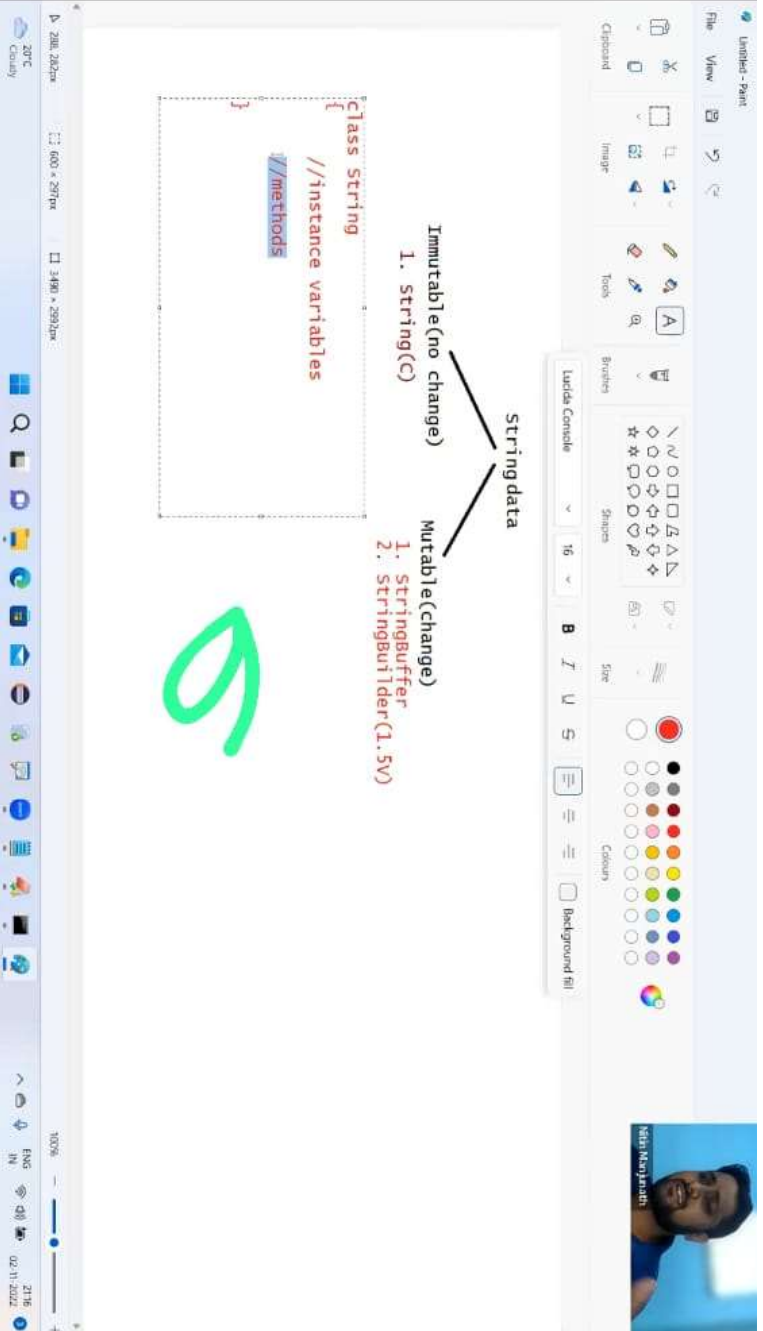
140%

Windows [CTRL]

UTF-8

20:04
02-11-2022





Untitled - Paint

FileView

ClipboardImageToolsBrushesShapesSizeColors

```
1 //instance variab
//methods
}

String s1="sachin";
s1.concat("tendulkar");
System.out.println(s1);
//sachin
```

stack

HeapArea

String
s1
(Immutable)

sachin

~~sachin
tendulkar~~

GarbageObject(no reference)

GarbageCollector

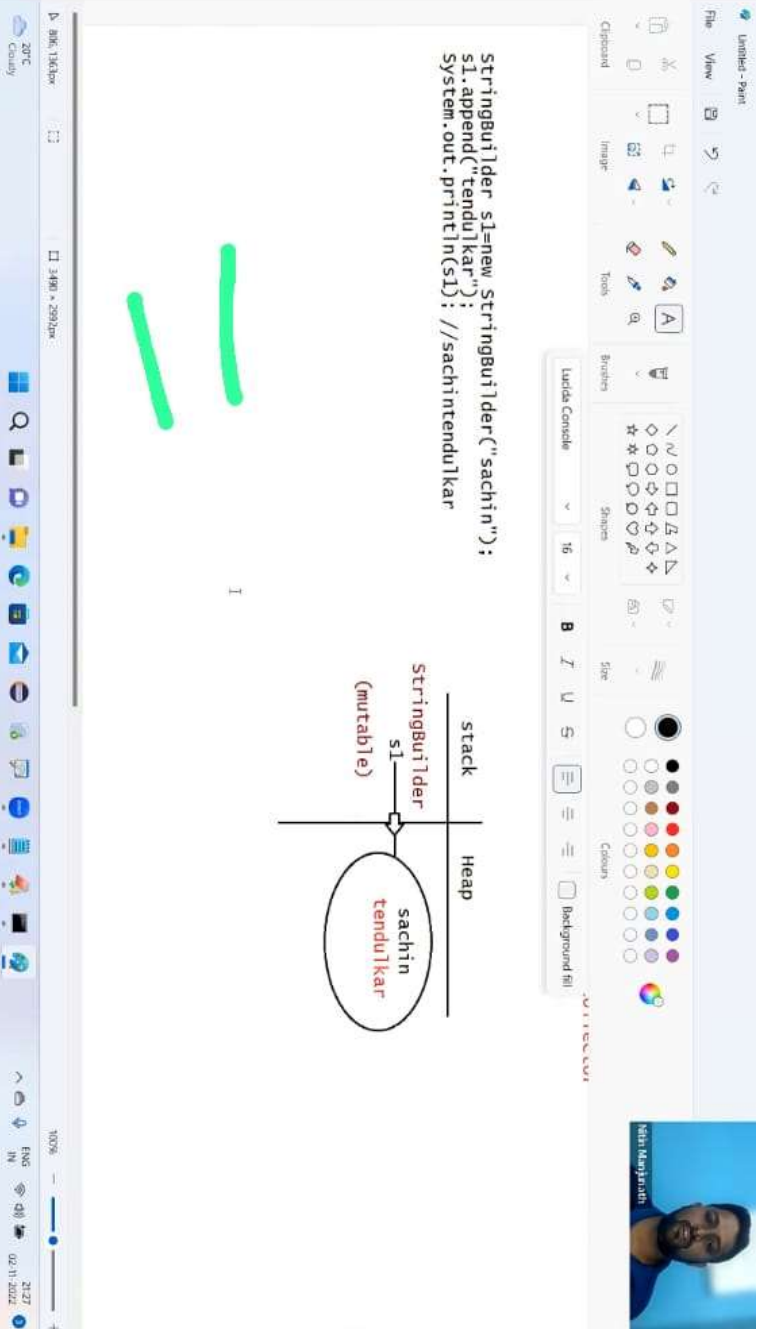
10

20°C
Cloudy

1 x Box3490 x 2592px

100%21:2302-11-2022

Nishu Majumdar



In java String object is by default immutable, meaning once the object is created we cannot change. String object, if we try to change then those changes will be reflected on the new object not on the existing object.

```
case 1:: String s= "sachin";  
s.concat("tendulkar");(new object got created with modification so immutable)  
System.out.println(s);  
  
output::sachin
```

vs

```
StringBuilder sb=new StringBuilder("sachin");  
sb.append("tendulkar");(on the same object modification so mutable)  
System.out.println(sb);  
  
output:: sachintendulkar
```



2.11.2022, string, arrays - Print

FileView

Clipboard

Image

Tools

Brushes

Shapes

Size

Colors

Lucida Console

16

B

I

U

S

Background fill

20°C

Coolby

3490 x 2592px

File Sync Backup

100%

21:39

02-11-2022

```
String s1="sachin";
String s2=new String("sachin");
System.out.println(s1==s2); //false
System.out.println(s1.equals(s2));
```

String class equals() will compare the content present inside the String "sachin".equals("sachin")
true

stack

(String)
s1

(String)
s2

Heap

sachin

sachin

13

Command Prompt

Nishu Mayaram

VS

```
StringBuilder sb=new StringBuilder("sachin");  
sb.append("tendulkar");//(on the same object modification so mutable)  
System.out.println(sb);
```

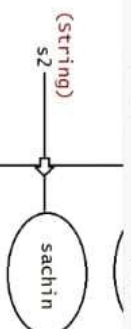
output:: sachintendulkar

```
case 2:: String s1 = new String("sachin");  
String s2 = new String("sachin");  
System.out.println(s1==s2); //false  
System.out.println(s1.equals(s2)); //true  
=> String class .equals method will compare the content of the object  
if same return true otherwise return false
```

24

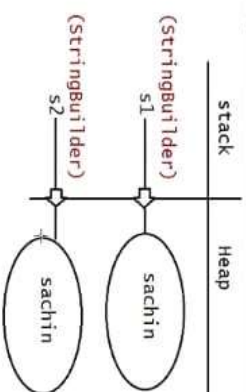


String class equals()
will compare the content
present inside the String
"sachin".equals("sachin")
true



```
StringBuffer s1=new StringBuffer("sachin");
StringBuffer s2=new StringBuffer("sachin");
System.out.println(s1 == s2);
System.out.println(s1.equals(s2));
```

StringBuffer class equals() compare the
reference(address of object) not the
Content of StringBuffer.
false



5-1



output:: sachintendulkar

case 2:: String s1 = new String("sachin");

String s2 = new String("sachin");

System.out.println(s1==s2); //false

System.out.println(s1.equals(s2)); //true

=> String class .equals method will compare the content of the object if same return true otherwise return false

VS

StringBuilder sb1 = new StringBuilder("sachin");

StringBuilder sb2 = new StringBuilder("sachin");

System.out.println(sb1==sb2); //false

System.out.println(sb1.equals(sb2)); //false

=> StringBuilder class .equals method is not overridden so it will use Object class .equals() which is meant for reference comparison.

if different object returns false, even if the contents are same.

16



21112022_string_string - Print

FileView

Clipboard

Image

Tools

Brushes

Shapes

Size

Colors

16

B

I

U

S

Background fill

21112022_string_string - Print

String s1=new String("sachin");

stack

s1

HeapArea

sachin

StringConstantPool (SCP)

sachin

Even though it is a GarbageObject
GarbageCollector can't clean the data
present in "SCP".

5

17th Nov 2022

100%

21:54

02-11-2022

Sanjay Singh

2.11.2022 string.cpp - Print

FileView

ClipboardImageToolsBrushesShapesColors

Even though it is a GarbageObject
GarbageCollector can't clean the data
present in "SCP".

stack

HeapArea

String s1="sachin";

SCP(StringConstantPool)

s1

sachin

18

100% 300xpx 3490 x 340px Src: 97.5KB

20°C Cloudy

100% 20:57 02-11-2022

Nan Darshan

What will be the result of compiling and executing Test class?

```
public class Test {  
    public static void main(String[] args) {
```

```
        int a = 7; // a = 7,8,9
```

```
        boolean res = ++a == 7 && ++a == 9 || ++a == 9; // 7==7(true) && 9==9(true) || ++a=9 = true
```

```
        System.out.println("a = " + a);
```

```
        System.out.println("res = " + res);
```

```
    }  
}
```

A. a=10

res=true

B. a=9

res=true

C. a=10

res=false

D. compilation error

Answer: B

81



Q>

```
class Fork {  
    public static void main(String[] args) {  
        if(args.length == 1 & args[1].equals("test"))  
            System.out.println("test case");  
        } else {  
            System.out.println("production " + args[0]);  
        }  
    }  
}
```

And the command-line invocation:
java Fork live2

What is the result?

- A. test case
- B. production live2
- C. test case live2
- D. Compilation fails
- E. An exception is thrown at runtime

JVM

args[0] = live2

20



```
public static void main(String[] args) {  
    if(args.length == 1 | args[1].equals("test"))  
        System.out.println("test case");  
    } else {  
        System.out.println("production " + args[0]);  
    }  
}
```

And the command-line invocation:
java Fork live2

What is the result?

- A. test case
- B. production live2
- C. test case live2
- D. Compilation fails
- E. An exception is thrown at runtime

JVM

args[0] = live2

args.length= 1

Answer: E(ArrayIndexOutOfBoundsException becoz args[1] is not available)

21



Q>

```
public class Test {  
    public static void main(String[] args) {  
        int aVar = 9; // aVar = 10  
        if(aVar++ < 10) // if(9<10)true  
            System.out.println(aVar + " Hello World!");  
        else  
            System.out.println(aVar + " Hello Universe!");  
    }  
}
```

- A. 10 Hello World!
- B. 9 Hello Universe!
- C. 10 Hello World
- D. Compilation fails.

Answer: A

22




```
int[] a = {1,2,3,4,5};  
for (xxx)  
{  
    System.out.print(a[e]);  
}  
}
```

Which option can replace xxx to enable the code to print 135?

- A. `int e = 0; e <= 4; e++`
- B. `int e = 0; e < 5; e += 2`
- C. `int e = 1; e <= 5; e += 1`
- D. `int e = 1; e < 5; e += 2`

`a[0] = 1`
`a[1] = 2`
`a[2] = 3`
`a[3] = 4`
`a[4] = 5`

Answer: B|becoz it should print data present in even index only)

23



```
public class Test {  
    public static void main(String[] args) {  
        int[] a = {1,2,3,4,5};  
        for (xxx)  
        {  
            System.out.print(a[e]);  
        }  
    }  
}
```

Which option can replace xxx to enable the code to print 135?

- A. `int e = 0; e <= 4; e++`
- B. `int e = 0; e < 5; e += 2`
- C. `int e = 1; e <= 5; e += 1`
- D. `int e = 1; e < 5; e += 2`

a[0] = 1
a[1] = 2
a[2] = 3
a[3] = 4
a[4] = 5

Answer: B|becoz it should print data present in even index only|

24



File Edit View

Q>

```
public class Test {  
    public static void main(String[] args) {  
        int num = 5; // num=5, 4  
        do {  
            System.out.println(num-- + " "); // System.out.println(5)  
        } while (num == 0);  
        4==0(false)  
    }  
}
```

What is the result?

- A. 5 4 3 2 1 0
- B. 5 4 3 2 1
- C. 4 2 1
- D. 5
- E. Nothing is printed
- F. 4
- G. CompileTimeError

Answer: D

|

152



Ln 106, Col 1

20°C
Cloudy



100%

Windows (CTRL)

U7F 6

22:30
02-11-2022

```
public static void main(String[] args) {  
    int ii=0;  
    int jj=7;  
    for (ii=0;ii<jj-1;ii=ii+2){  
        System.out.print(ii);  
    }  
}
```

What is the result?

- A. 2 4
- B. 0 2 4 6
- C. 0 2 4
- D. Compilation fails

JVM

ii=0, 0<6(true)
ii=2, 2<6(true)
ii=4, 4<6(true)
ii=6, 6<6(false)
Answer: 0 2 4

26



File Edit View

```
array[0] = 10;  
array[1] = 20;  
System.out.print(array[0]+":" array[1]);  
}  
}
```

Which code fragment, when inserted at line 3, enables the code to print 10:20?

- A. `int[] array n= new int[2];`
- B. `int[] array;`
`array = new int[2];`
- C. `int array = new int[2];`
- D. `int array [2] ;`

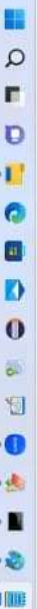
JVM

- A. `int[] array n= new int[2];` //invalid identifier
- B. `int[] array;` //array declaration
`array = new int[2];` //array initialization
- C. `int array = new int[2];` //[] is missing
- D. `int array [2] ;`

23



Ln: 153, Col: 19
20°C
Cloudy



100% Windows [CTRL]
U7F: 6
22:36
02-11-2022

array = new int[2]; //array initialization
C. int array = new int[2]; //[] is missing
D. int array [2]; //new is missing

Q>

```
public class Test {  
    public static void main(String[] args) {  
        boolean opt = true;  
        switch (opt){  
            case true: System.out.print("True");  
                break;  
            default : System.out.print("***");  
        }  
        System.out.print("Done");  
    }  
}
```

Predict the Output?

- A. CompileTime Error.
- B. Some problem occurred by jvm during execution.
- C. True
- D. True

28



Predict the Output?

- A. CompileTime Error.
- B. Some problem occurred by jvm during execution.
- C. True
- Done
- D. True

- Done
- E. None of the above.

Note:

switch(arg)

- | -> byte--> Byte
- | -> short--> Short
- | -> int --> Integer
- | -> char --> Character
- | -> String
- | -> enum

Answer: A

I

29



File Edit View

```
public static void main(String[] args) {  
    int intArr[] = {15,30,45,60,75};  
    intArr[2] =intArr[4];  
    intArr[4] = 90;  
}
```

What are the values of each element in intArr after this code has executed?

- A. 15, 60, 45, 90, 75
- B. 15, 90, 45, 90, 75
- C. 15, 30, 75, 60, 90
- D. 15, 30, 90, 60, 90
- E. 15, 4, 45, 60, 90

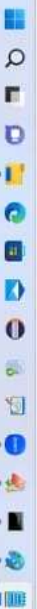
```
intArr[0] = 15  
intArr[1] = 30  
intArr[2] = 45,75  
intArr[3] = 60  
intArr[4] = 75,90
```

Output: 15,30,75,60,90
Answer: C

30



Ln 208, Col 23
20°C
Cloudy



100% Windows (CTRL)
22°C
02-11-2022

eg#3.

```
class Test{  
    public static void main(String[] args){  
        int x; //local variable  
        System.out.println("hello");  
    }  
}
```

- A. CE
- B. some problem at runtime
- C. hello
- D. none of the above

Answer: C (becoz x is not initialized)

eg#4

```
class Test{  
    public static void main(String[] args){  
        int x;  
        System.out.println(x);  
    }  
}
```

CE



- C. hello
- D. none of the above

Answer: C (becoz x is not initialized)

eg#4

```
class Test{  
    public static void main(String[] args){  
        int x;//local variable ----> no default initialization by jvm, programmer should do  
        System.out.println(x);  
    }  
}
```

A. CE

B. some problem at runtime

C. hello

D. 0

E. none of the above

Answer: A (becoz x is not initialized)

eg#5.

class Test{

32

Ln 237, Col 10

20°C
Cloudy



100%

Windows [CTRL]

UTR 6

22:45
02-11-2022



eg#5.

```
class Test{  
    public static void main(String[] args) {  
        int x;  
        if(args.length>0){  
            x=10;  
        }  
        System.out.println(x);  
    }  
}
```

Assume java Test 10 is passed through command line

- A. CE
- B. some problem at runtime
- C. hello
- D. 10
- E. none of the above

Answer: A

CE



eg#6
class Test {

public static void main(String[] args) {

int x;

if(args.length>0){

x=10;

}else{

x=20;

}

System.out.println(x);

}

}
Assume java Test sachin is passed through command line

- A. CE
- B. some problem at runtime
- C. hello
- D. 10
- E. 20
- F. None of the above

54



File Edit View

```
}  
}
```

Assume java Test sachin is passed through command line

- A. CE
- B. some problem at runtime
- C. hello
- D. 10
- E. 20
- F. None of the above

```
java Test sachin
```

```
args[0] = "sachin"
```

```
args.length = 1
```

```
x = 10
```

```
java Test
```

```
args.length = 0
```

```
x = 20
```

35



Ln 288, Col 10

20°C
Cloudy



100%

Windows (CTRL)

U7F: 6

22:53
02-11-2022