V.V.I

Super Keyword

Q. Super Keyword ? full explaination.

Ans→ Super keyword refers to the objects of super class, it is used when we want to call the super class variable, method & constructor through sub class object.

Note:- 1) Whenever the super class & sub class variable and method name both are same than it can be used only.

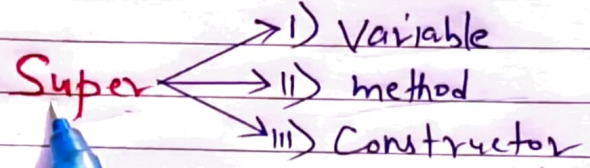
SUBSCRIBE

Note:- i) Whenever the Super class & Sub class Variable and method name both are Same than it can be used only.

ii) To avoid the Confusion between Super class and Sub classes variables and methods that have Same name we should use Super keyWord.

SUBSCRIBE

methods that have same name we should use Super keyword.

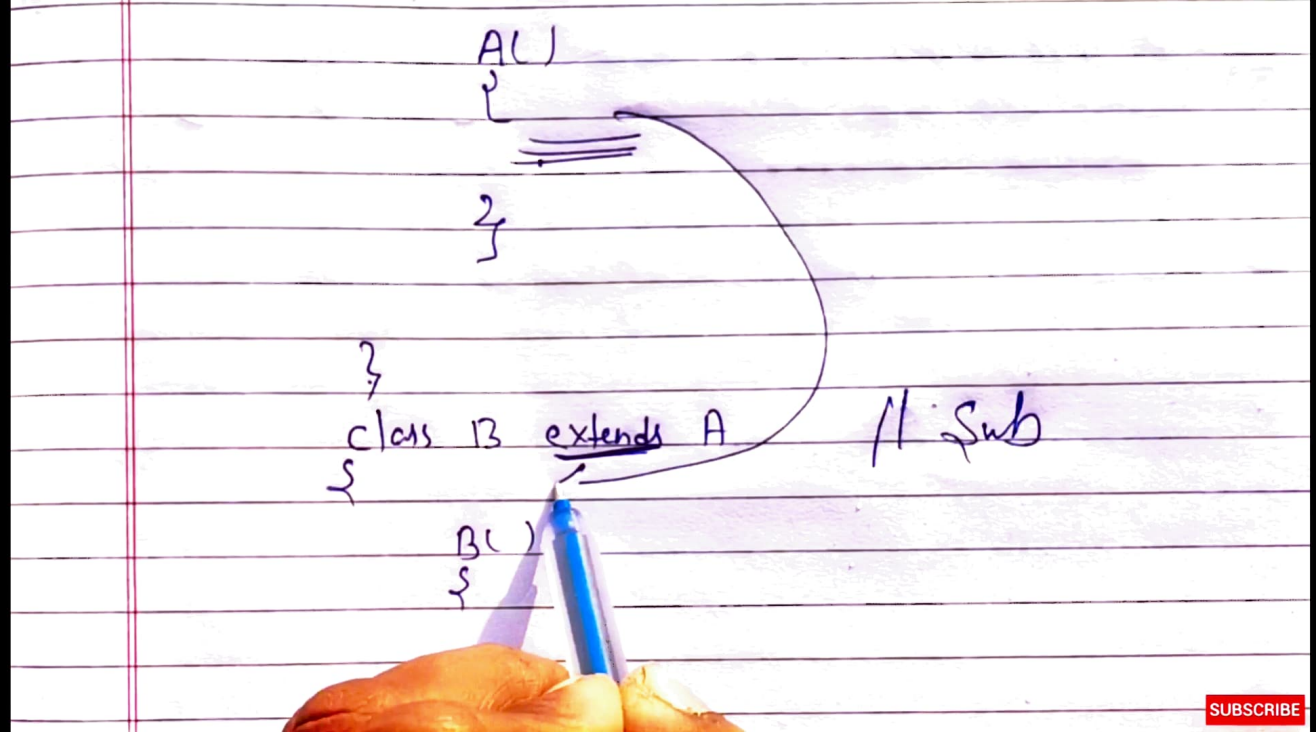


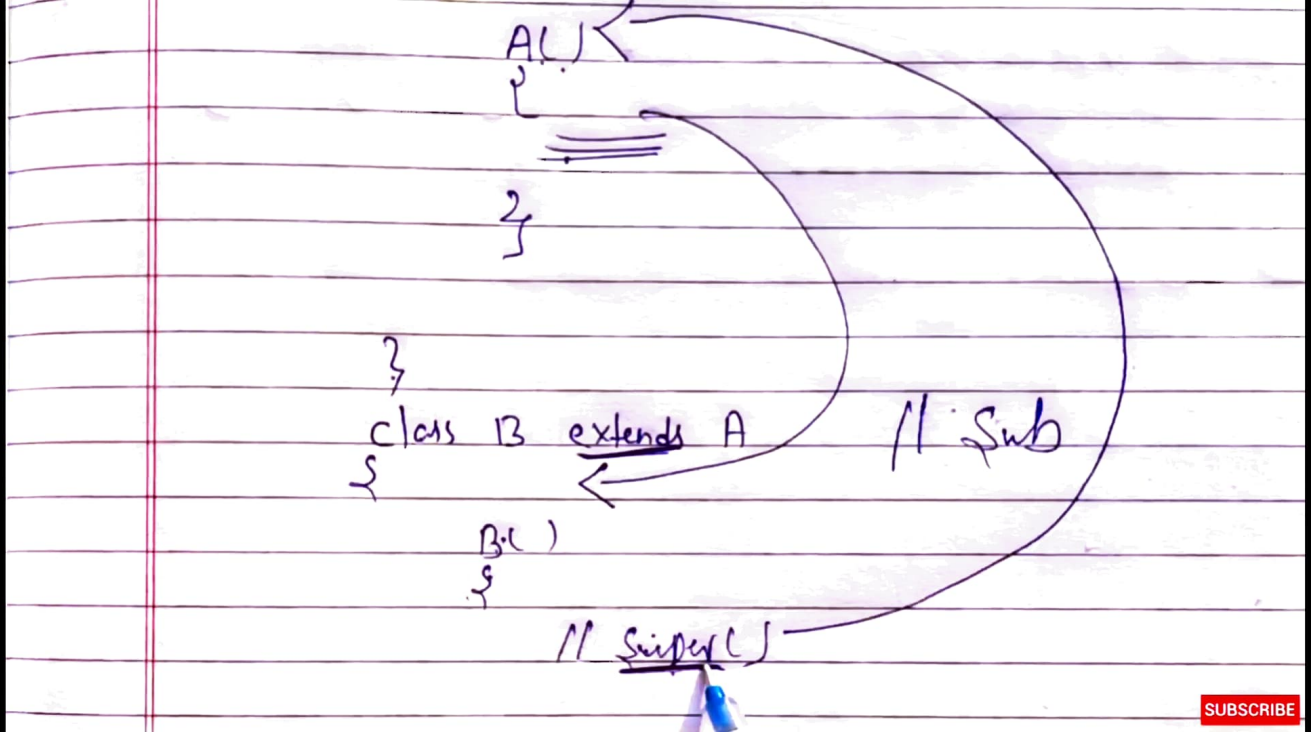
SUBSCRIBE

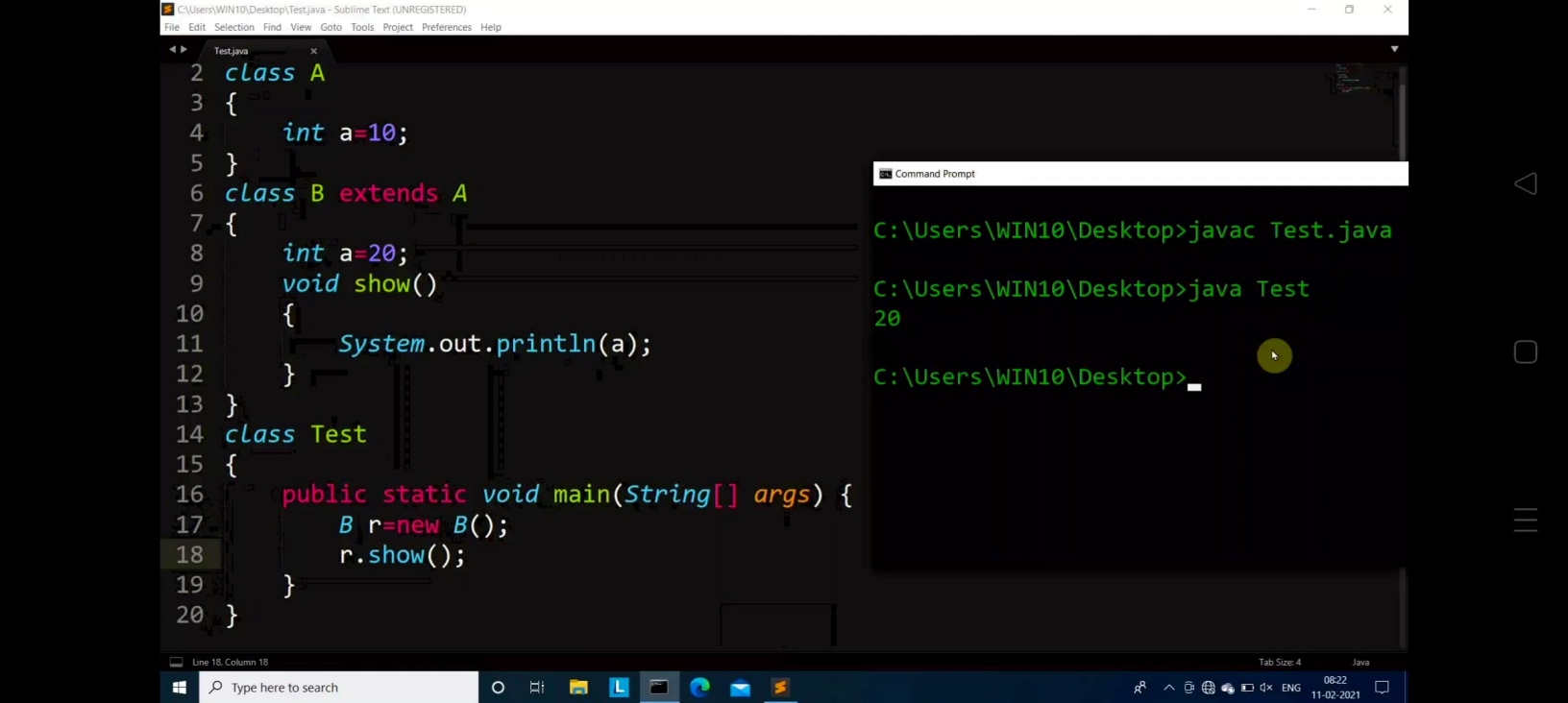
Syntax:- class A. // Super

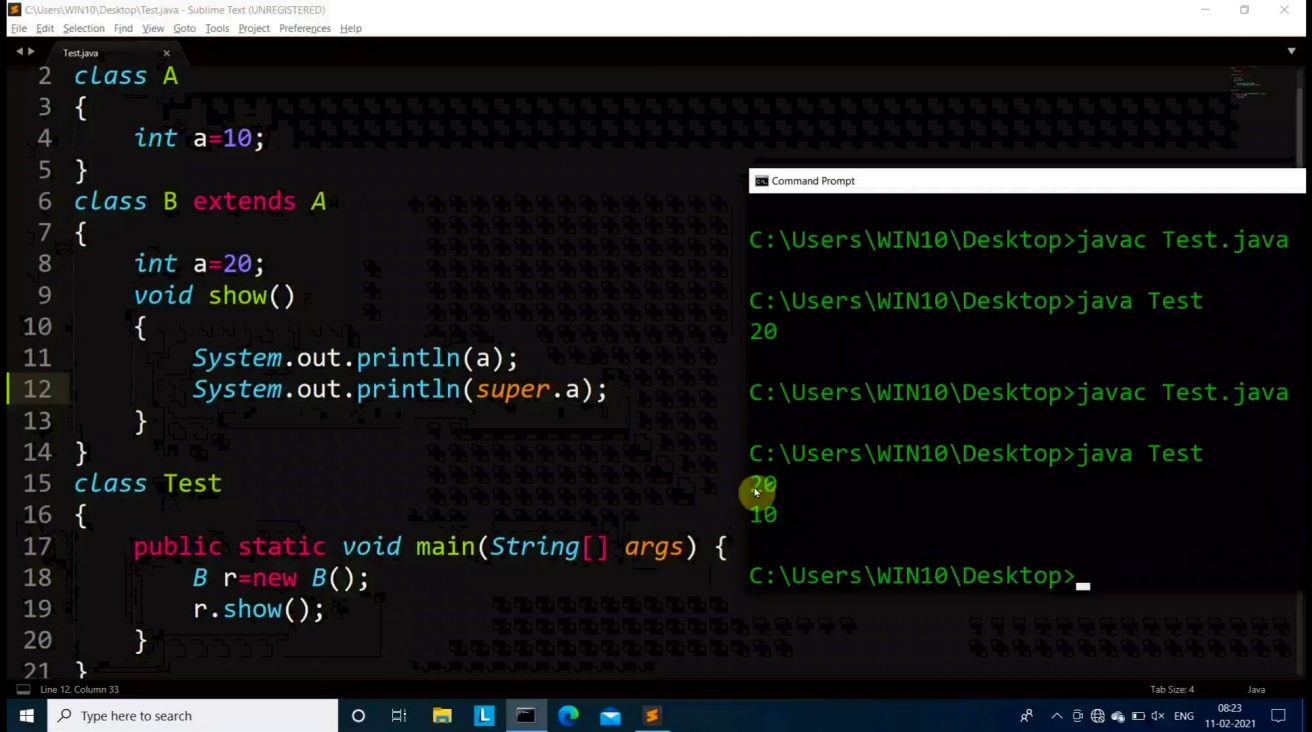
A()

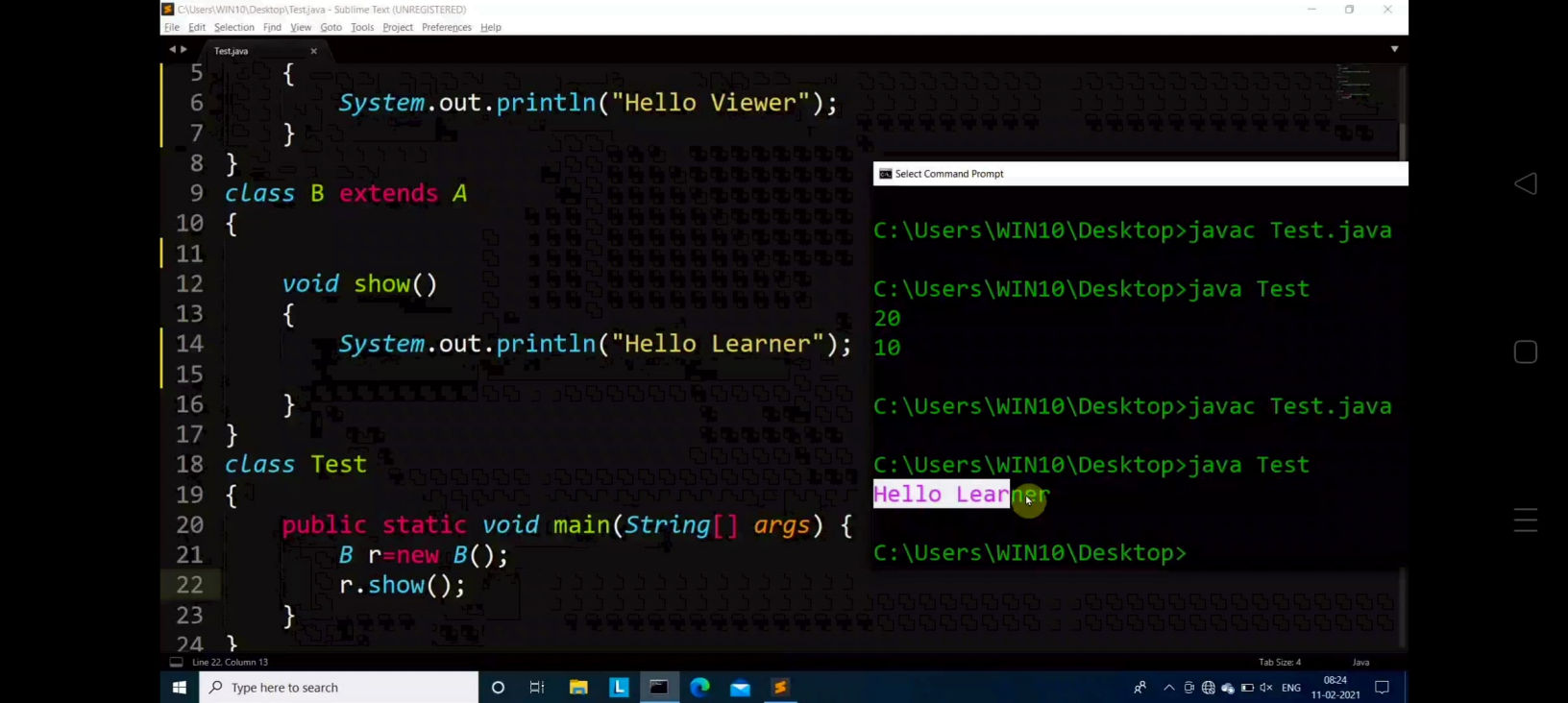
SUBSCRIBE

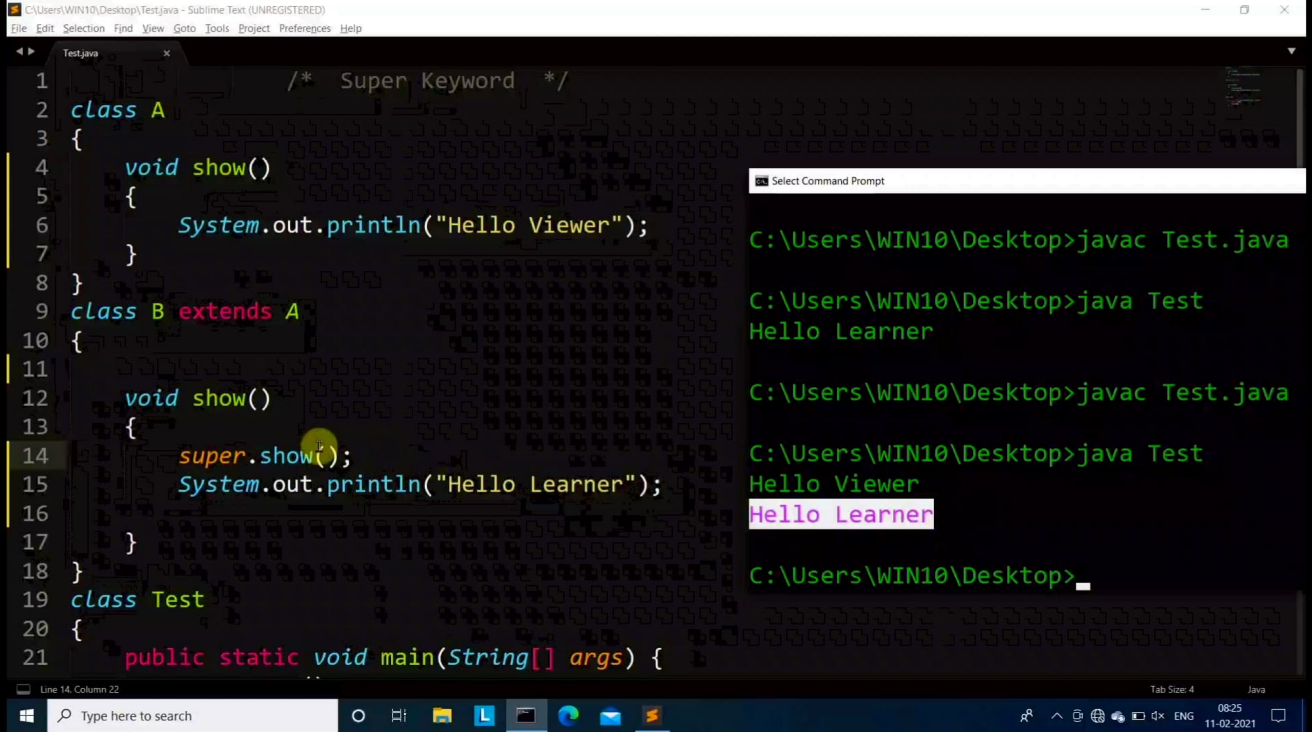

A( )
{
}
}
class B extends A
{
B( )
{
// Sub
SUBSCRIBE

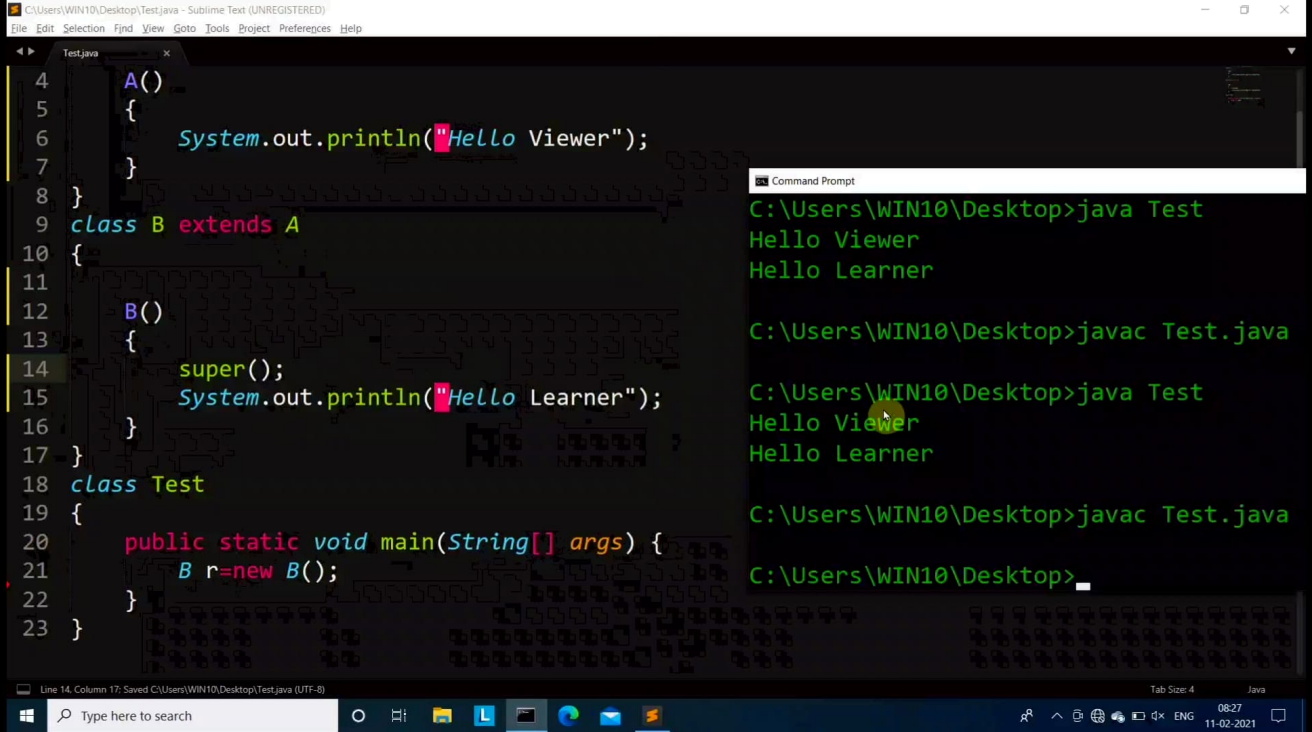

A.U
}
class B extends A // Sub
{
B.( )
{
// super( )

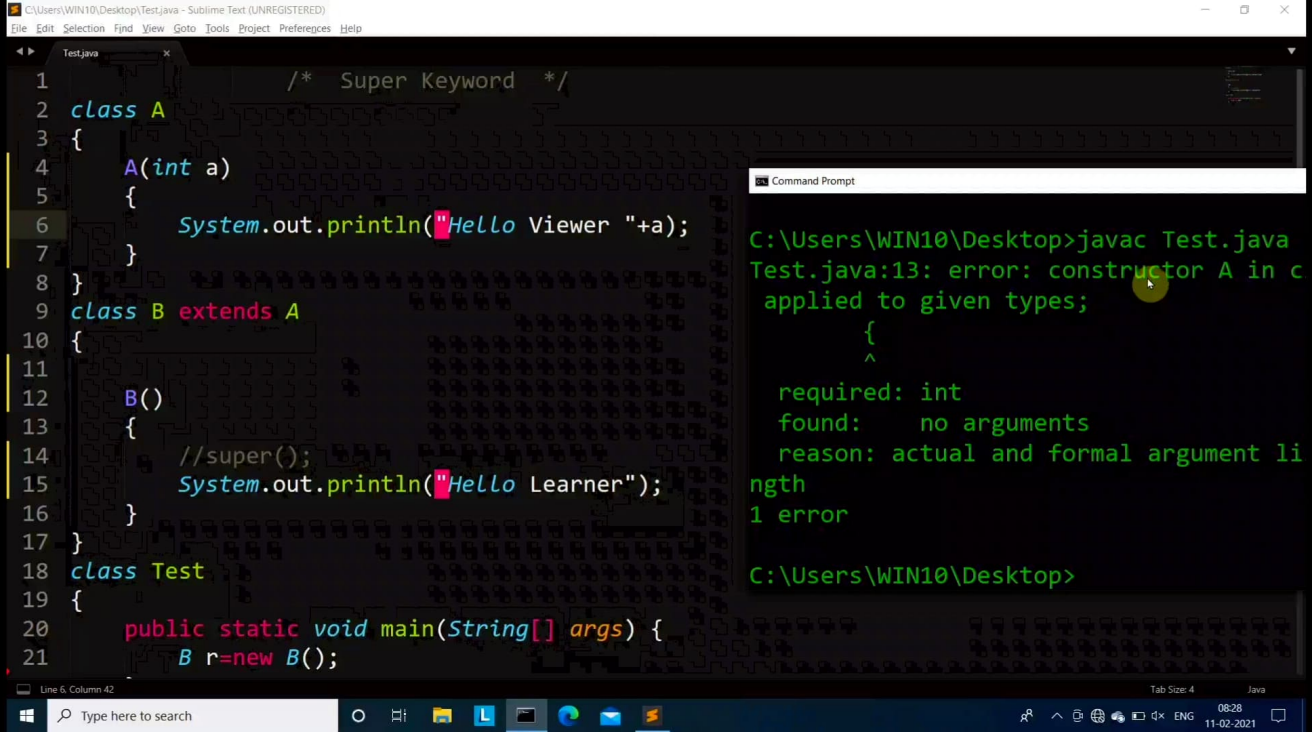

C:\Users\WIN10\Desktop\Test.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Test.java
class A
{
int a=10;
}
class B extends A
{
int a=20;
void show()
{
System.out.println(a);
}
}
class Test
{
public static void main(String[] args) {
B r=new B();
r.show();
}
}
Command Prompt
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
20
C:\Users\WIN10\Desktop>
Line 18, Column 18
Tab Size: 4
Java
Type here to search
ENG
08:22
11-02-2021

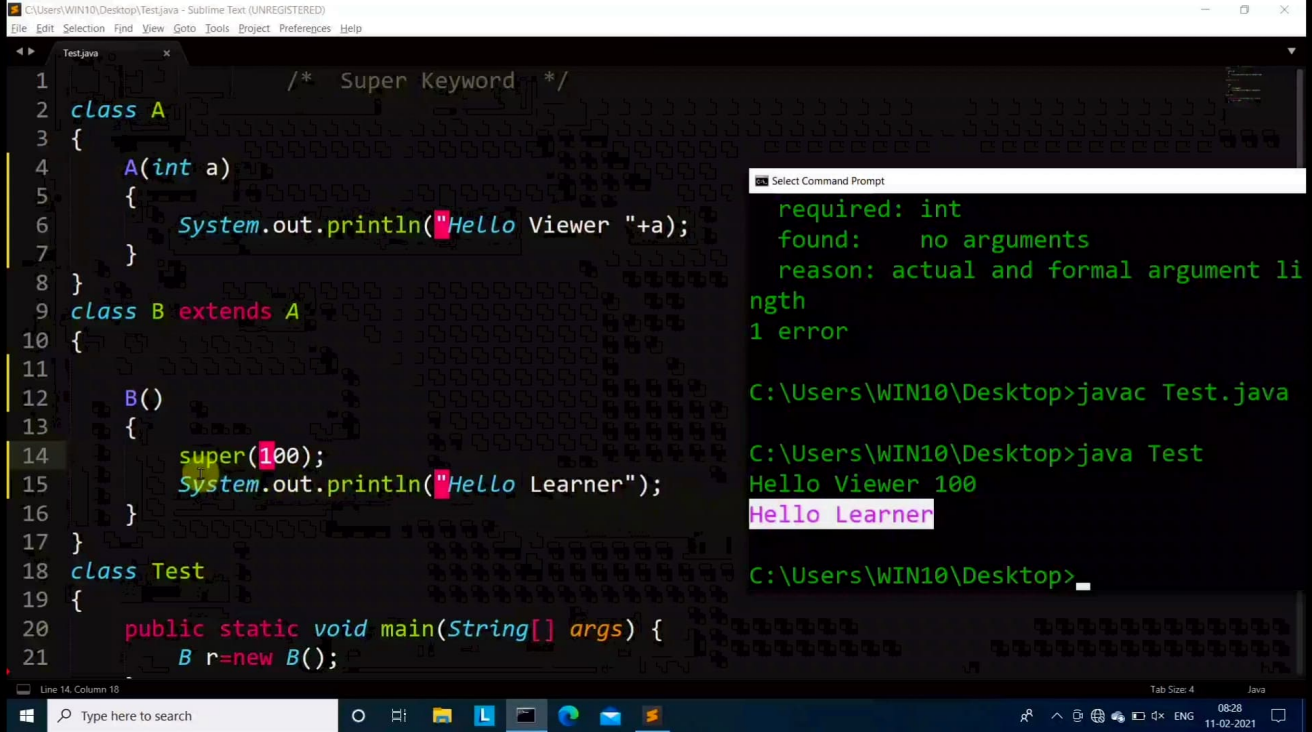

C:\Users\WIN10\Desktop\Test.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Test.java
class A
{
int a=10;
}
class B extends A
{
int a=20;
void show()
{
System.out.println(a);
System.out.println(super.a);
}
}
class Test
{
public static void main(String[] args) {
B r=new B();
r.show();
}
}
Line 12, Column 33
Tab Size: 4
Java
Command Prompt
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
20
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
20
10
C:\Users\WIN10\Desktop>
Type here to search
ENG
08:23
11-02-2021

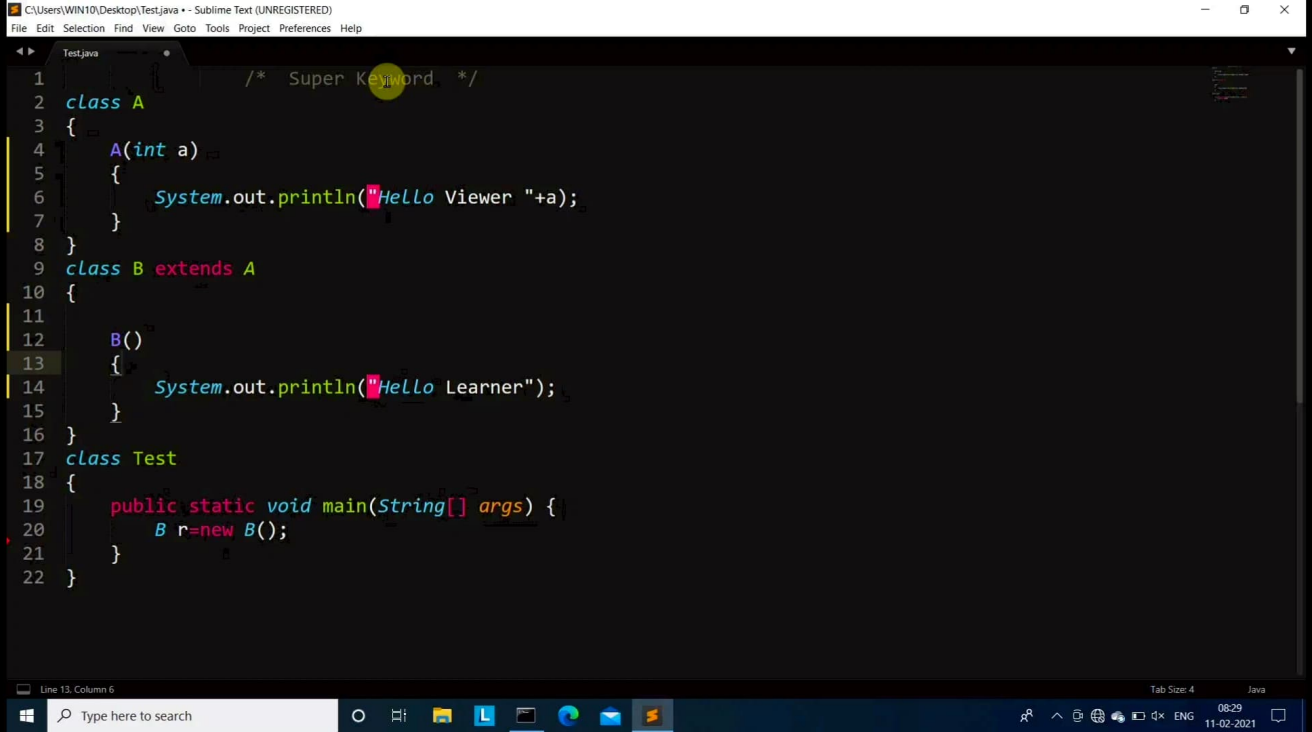

C:\Users\WIN10\Desktop\Test.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Test.java
{
System.out.println("Hello Viewer");
}
}
class B extends A
{
void show()
{
System.out.println("Hello Learner");
}
}
class Test
{
public static void main(String[] args) {
B r=new B();
r.show();
}
}
Line 22, Column 13
Tab Size: 4
Java
Select Command Prompt
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
20
10
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
Hello Learner
C:\Users\WIN10\Desktop>
Type here to search
ENG
08:24
11-02-2021


C:\Users\WIN10\Desktop\Test.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Test.java
/* Super Keyword */
class A
{
void show()
{
System.out.println("Hello Viewer");
}
}
class B extends A
{
void show()
{
super.show();
System.out.println("Hello Learner");
}
}
class Test
{
public static void main(String[] args) {
Line 14, Column 22
Tab Size: 4
Java
Select Command Prompt
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
Hello Learner
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
Hello Viewer
Hello Learner
C:\Users\WIN10\Desktop>
Type here to search
ENG
08:25
11-02-2021


C:\Users\WIN10\Desktop\Test.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Test.java
A()
{
System.out.println("Hello Viewer");
}
}
class B extends A
{
B()
{
super();
System.out.println("Hello Learner");
}
}
class Test
{
public static void main(String[] args) {
B r=new B();
}
}
Command Prompt
C:\Users\WIN10\Desktop>java Test
Hello Viewer
Hello Learner
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
Hello Viewer
Hello Learner
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>
Line 14, Column 17; Saved C:\Users\WIN10\Desktop\Test.java (UTF-8)
Tab Size: 4
Java
Type here to search
ENG
08:27
11-02-2021


C:\Users\WIN10\Desktop\Test.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Test.java
/* Super Keyword */
class A
{
A(int a)
{
System.out.println("Hello Viewer "+a);
}
}
class B extends A
{
B()
{
//super();
System.out.println("Hello Learner");
}
}
class Test
{
public static void main(String[] args) {
B r=new B();
Command Prompt
C:\Users\WIN10\Desktop>javac Test.java
Test.java:13: error: constructor A in c
applied to given types;
{
^
required: int
found: no arguments
reason: actual and formal argument li
ngth
1 error
C:\Users\WIN10\Desktop>
Line 6, Column 42
Tab Size: 4
Java
Type here to search
ENG
08:28
11-02-2021


C:\Users\WIN10\Desktop\Test.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Test.java
/* Super Keyword */
class A
{
A(int a)
{
System.out.println("Hello Viewer "+a);
}
}
class B extends A
{
B()
{
super(100);
System.out.println("Hello Learner");
}
}
class Test
{
public static void main(String[] args) {
B r=new B();
Line 14, Column 18
Tab Size: 4
Java
Select Command Prompt
required: int
found: no arguments
reason: actual and formal argument li
ngth
1 error
C:\Users\WIN10\Desktop>javac Test.java
C:\Users\WIN10\Desktop>java Test
Hello Viewer 100
Hello Learner
C:\Users\WIN10\Desktop>
Type here to search
ENG
08:28
11-02-2021


C:\Users\WIN10\Desktop\Test.java • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Test.java
/* Super Keyword */
class A
{
A(int a)
{
System.out.println("Hello Viewer "+a);
}
}
class B extends A
{
B()
{
System.out.println("Hello Learner");
}
}
class Test
{
public static void main(String[] args) {
B r=new B();
}
}
Line 13, Column 6
Tab Size: 4
Java
Type here to search
ENG
08:29
11-02-2021

Q. What is this keyword? full explaination.

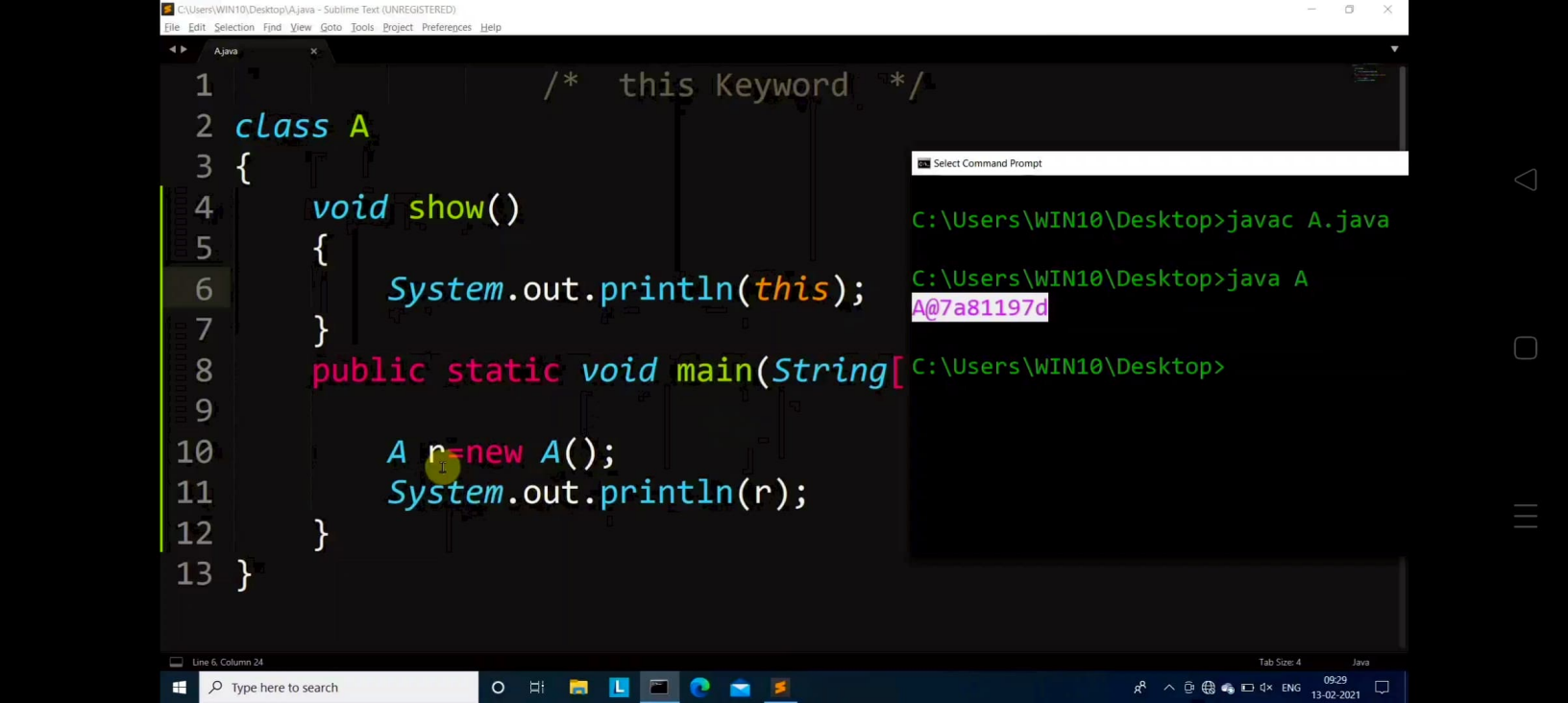
Ans → i) this keyword refers to the current object inside a method or constructor.

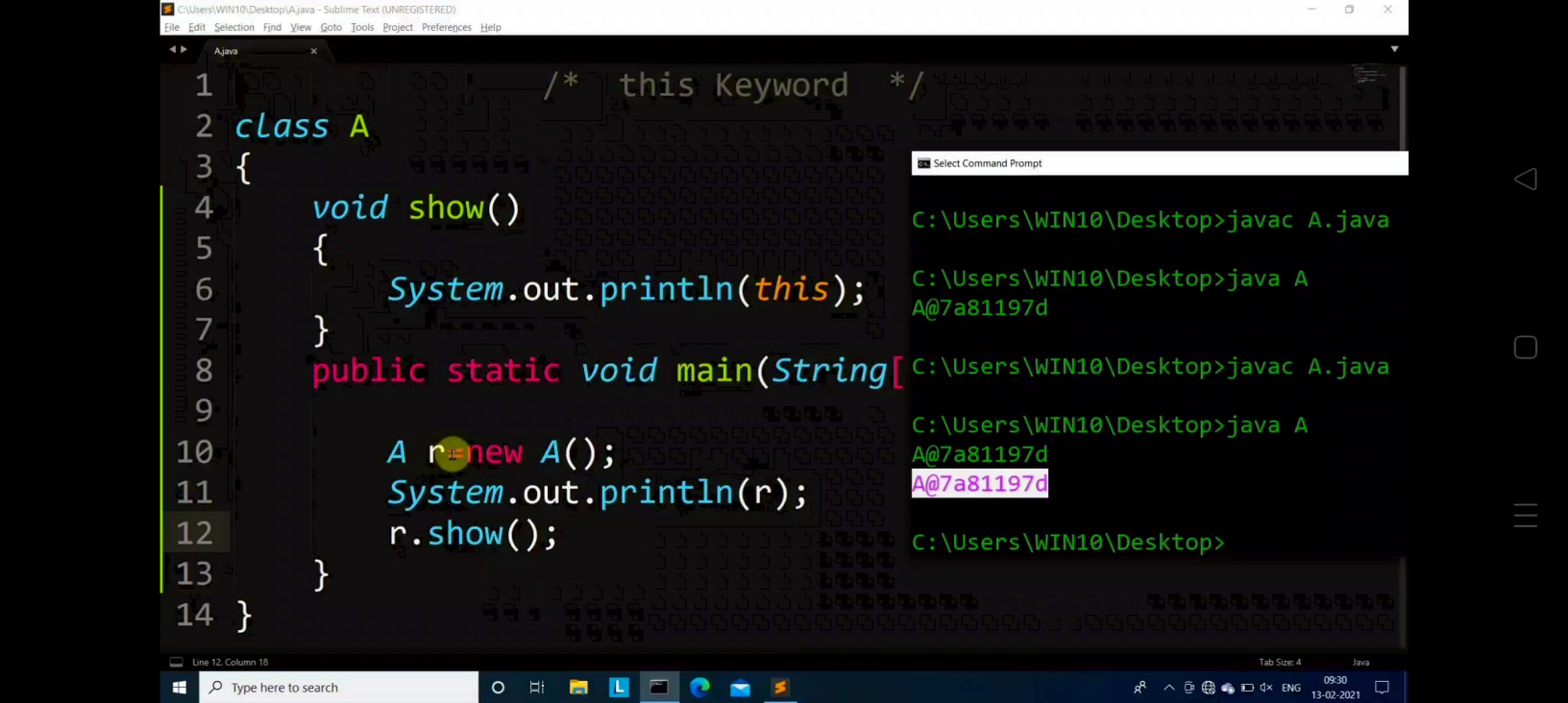
example:- class A
{

}

A r = new A();

ii) Whenever the name of instance and local variables both are same then runtime environment JVM ge... which


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
void show()
{
System.out.println(this);
}
public static void main(String[
A r=new A();
System.out.println(r);
}
}
Select Command Prompt
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
A@7a81197d
C:\Users\WIN10\Desktop>
Line 6, Column 24
Tab Size: 4
Java
Type here to search
ENG
09:29
13-02-2021


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
void show()
{
System.out.println(this);
}
public static void main(String[
A r=new A();
System.out.println(r);
r.show();
}
}
Select Command Prompt
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
A@7a81197d
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
A@7a81197d
A@7a81197d
C:\Users\WIN10\Desktop>
Line 12, Column 18
Tab Size: 4
Java
Type here to search
ENG
09:30
13-02-2021

ii) Whenever the name of instance and local variables both are same then our runtime environment JVM gets confused that which one is local variable & which one is instance variable, to avoid this problem we should use this keyword.

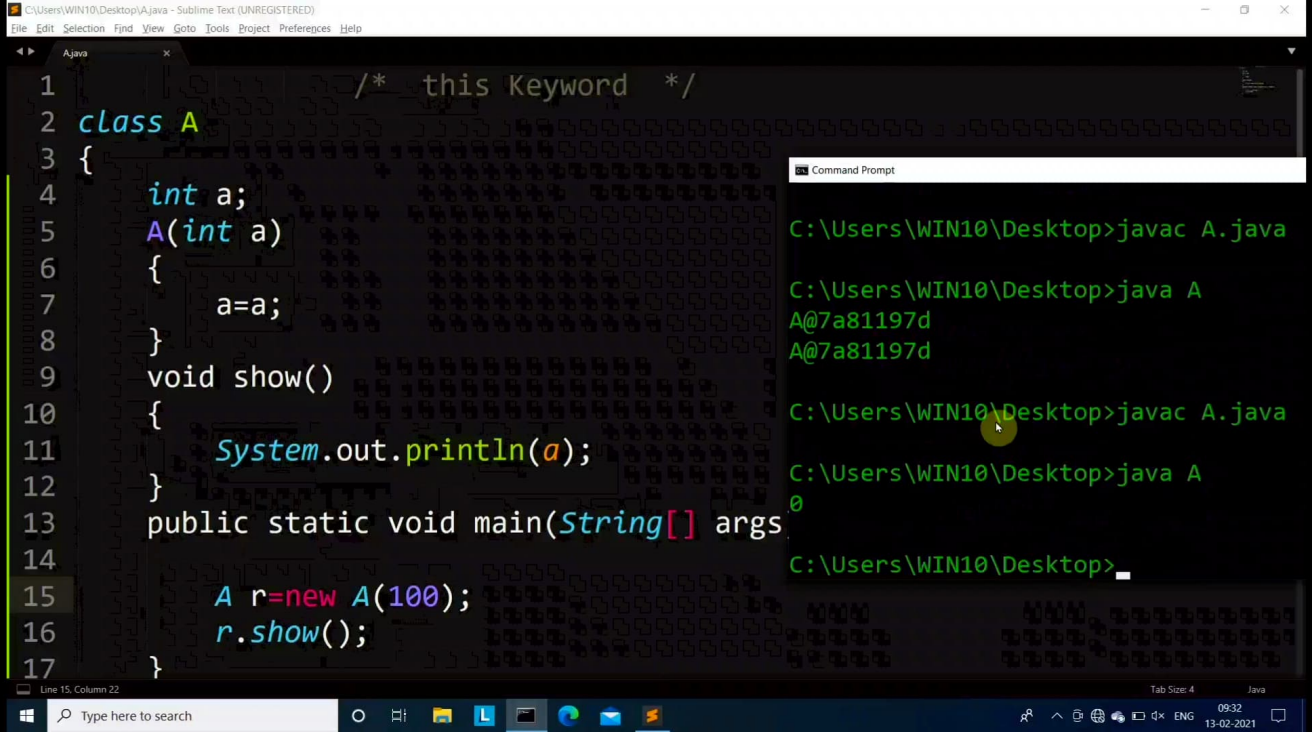
example:- class

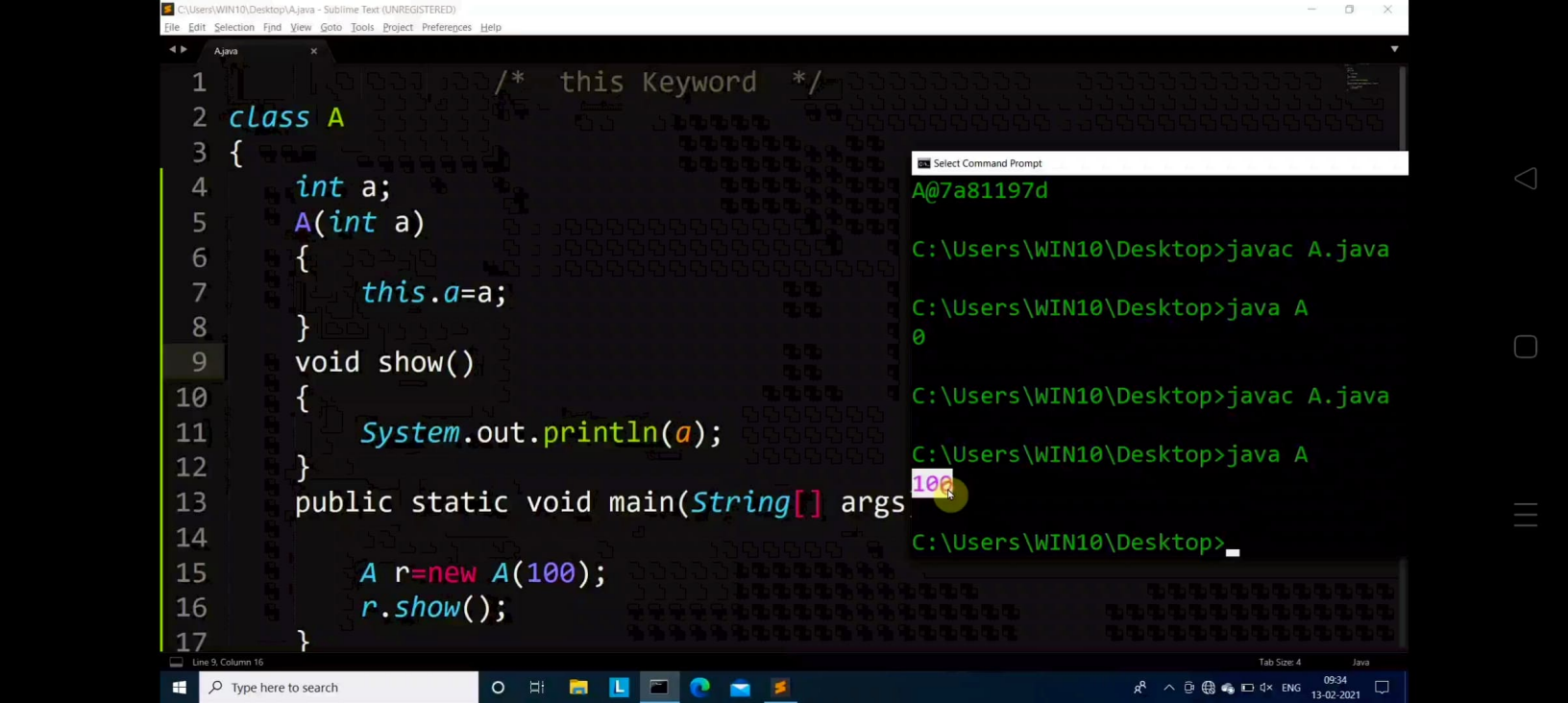
{ int a;

is instance variable, to avoid this problem we should use this keyword.

example:-

```
class A
{
    int a;  // instance
    A(int a)  // local
    {
        a = ;
        S.o.f( );
    }
}
```


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
int a;
A(int a)
{
a=a;
}
void show()
{
System.out.println(a);
}
public static void main(String[] args
A r=new A(100);
r.show();
}
Command Prompt
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
A@7a81197d
A@7a81197d
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
0
C:\Users\WIN10\Desktop>
Line 15, Column 22
Tab Size: 4
Java
Type here to search
ENG
09:32
13-02-2021


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
int a;
A(int a)
{
this.a=a;
}
void show()
{
System.out.println(a);
}
public static void main(String[] args
A r=new A(100);
r.show();
}
Select Command Prompt
A@7a81197d
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
0
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
100
C:\Users\WIN10\Desktop>
Line 9, Column 16
Tab Size: 4
Java
Type here to search
ENG
09:34
13-02-2021

V.V.I

Page No. :
Date : / /

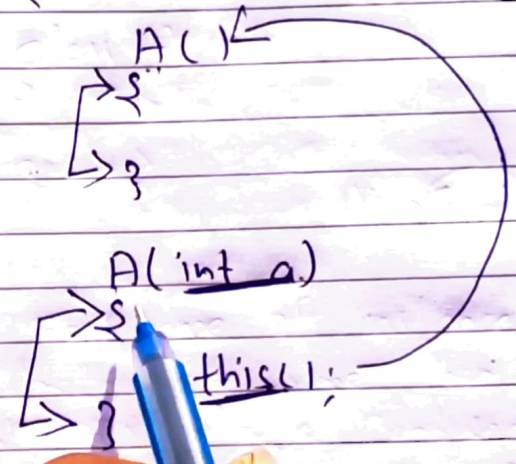
this Keyword

3) It is also used when we want to call the default constructor of its own class.

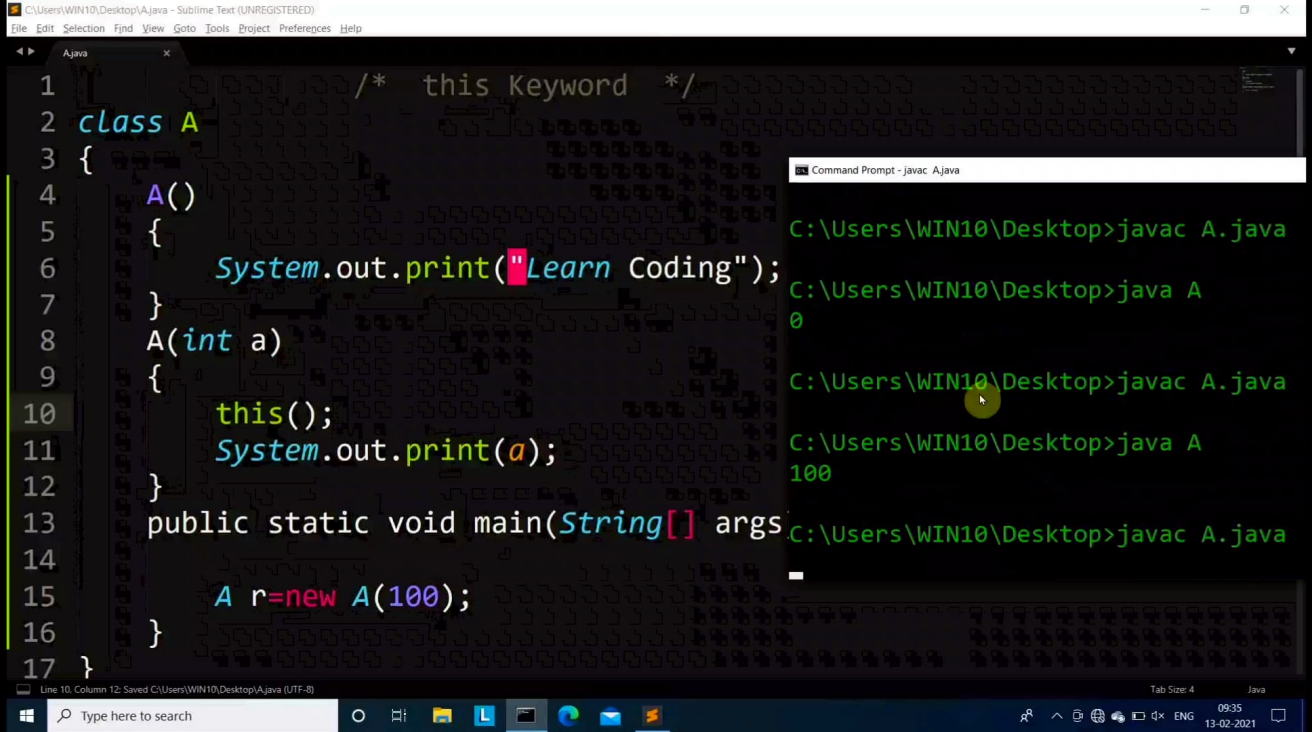
example:-

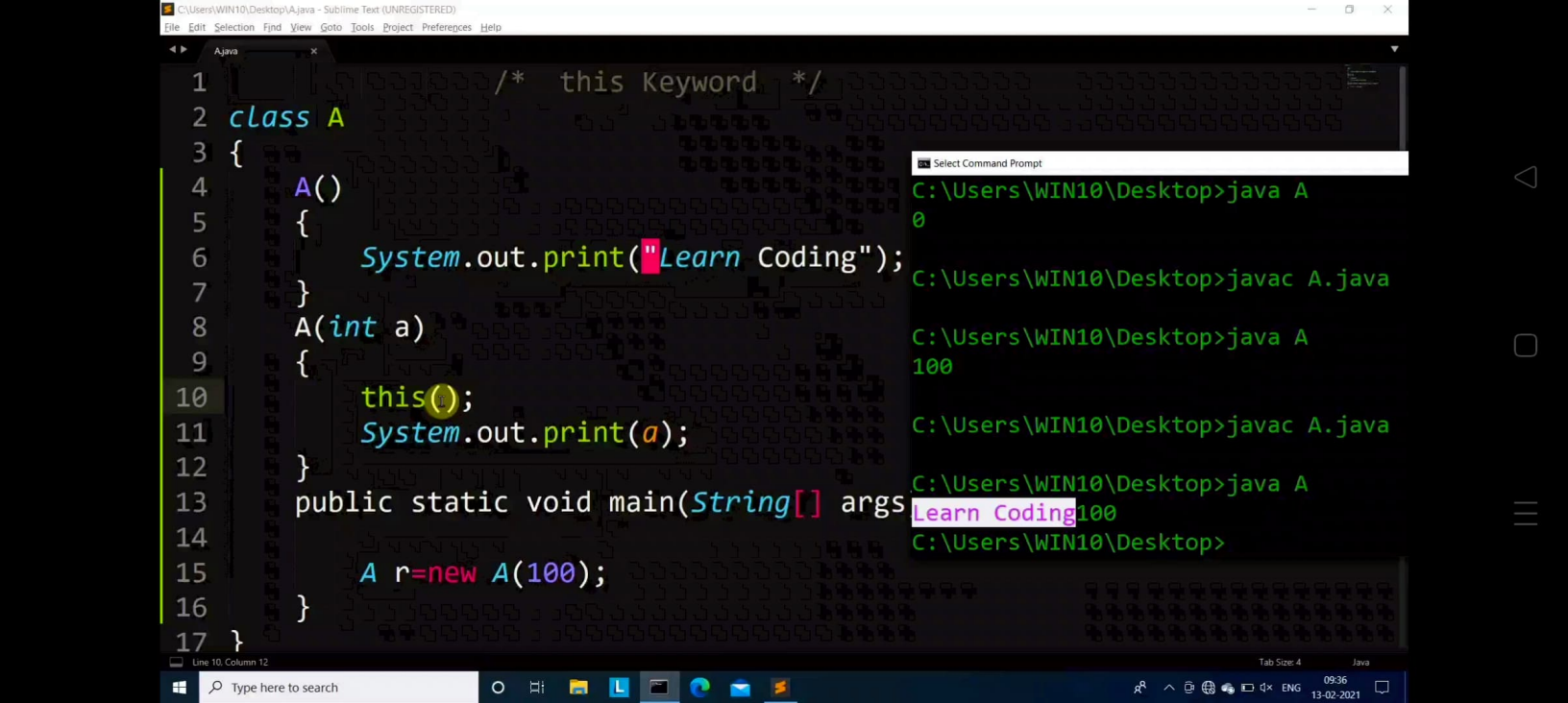
```
class A
{
    A()
    {
    }
```

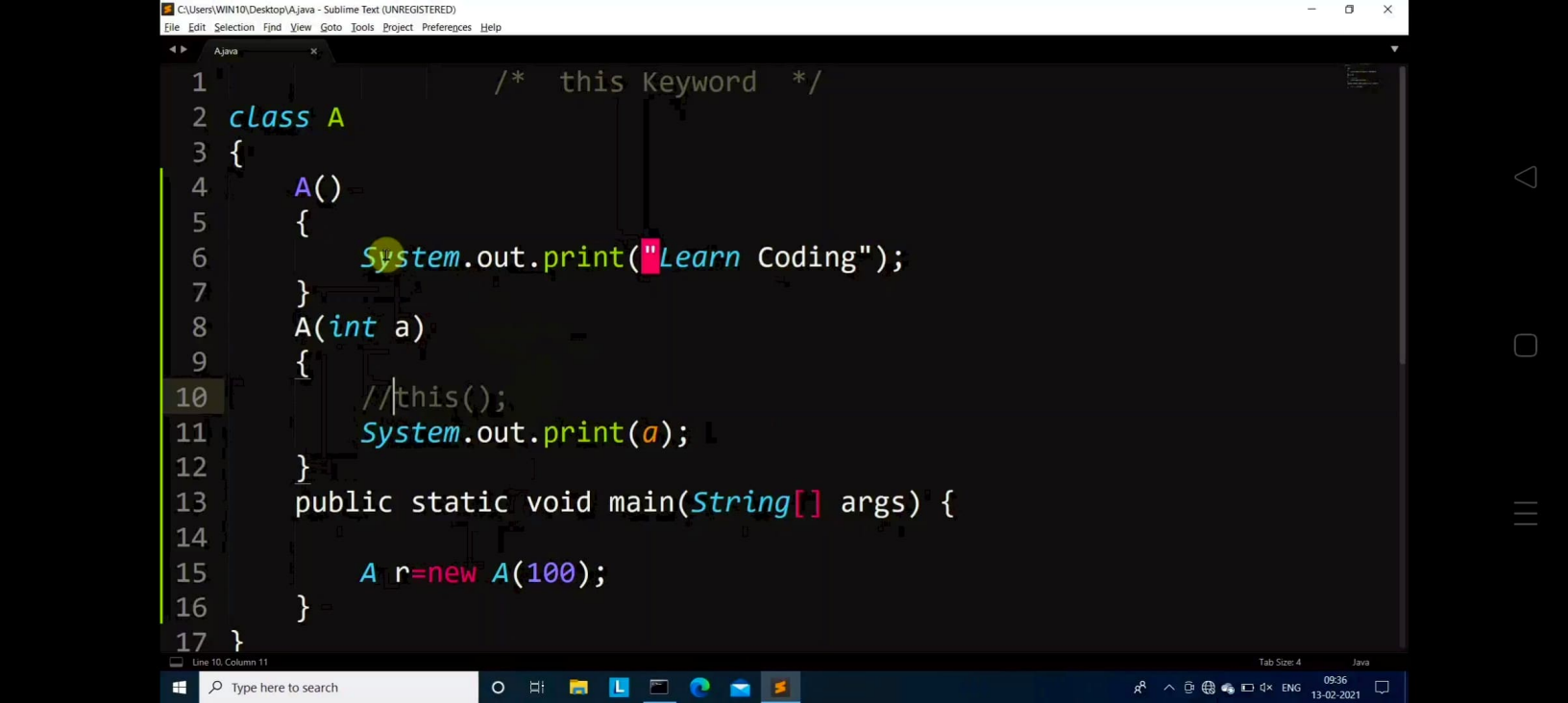
example:- class A.



4) ...rized constructor


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
A()
{
System.out.print("Learn Coding");
}
A(int a)
{
this();
System.out.print(a);
}
public static void main(String[] args
A r=new A(100);
}
}
Line 10, Column 12; Saved C:\Users\WIN10\Desktop\A.java (UTF-8)
Tab Size: 4
Java
Command Prompt - javac A.java
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
0
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
100
C:\Users\WIN10\Desktop>javac A.java
Type here to search
ENG
09:35
13-02-2021

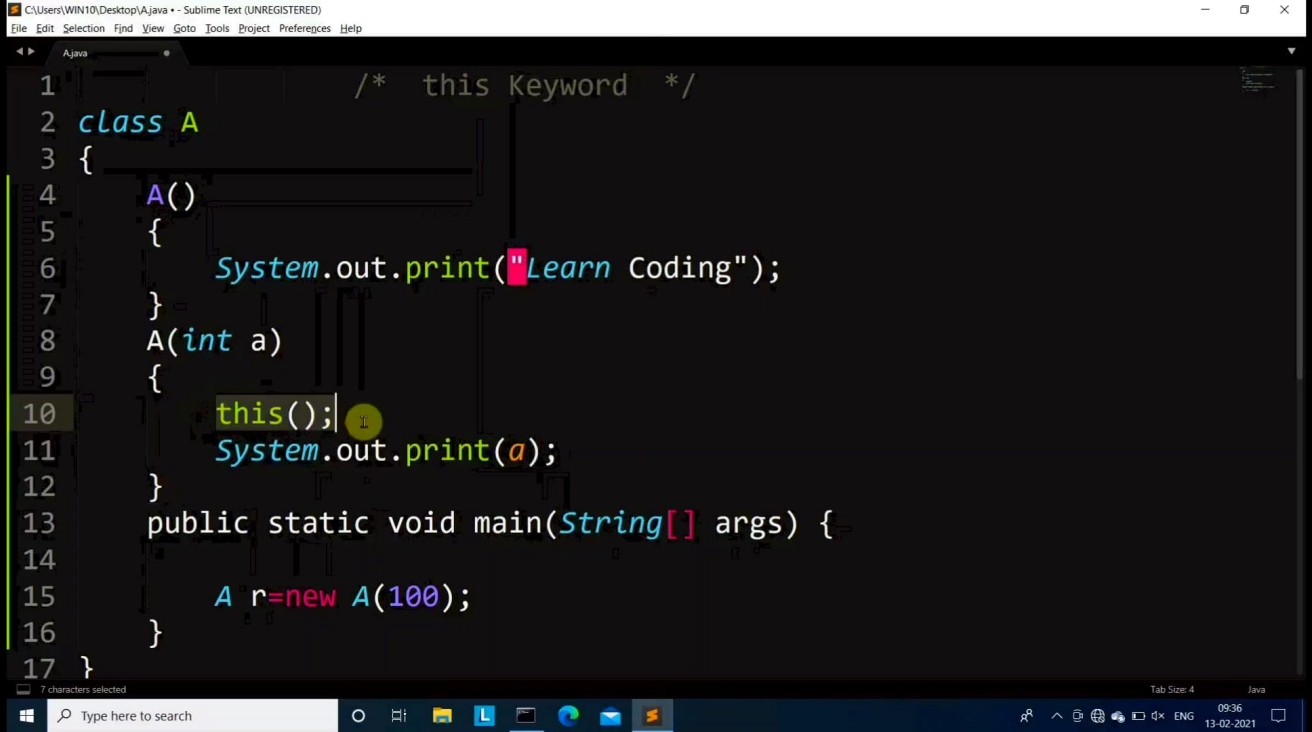

C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
A()
{
System.out.print("Learn Coding");
}
A(int a)
{
this();
System.out.print(a);
}
public static void main(String[] args
A r=new A(100);
}
}
Line 10, Column 12
Tab Size: 4
Java
Select Command Prompt
C:\Users\WIN10\Desktop>java A
0
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
100
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
Learn Coding100
C:\Users\WIN10\Desktop>
Type here to search
ENG
09:36
13-02-2021

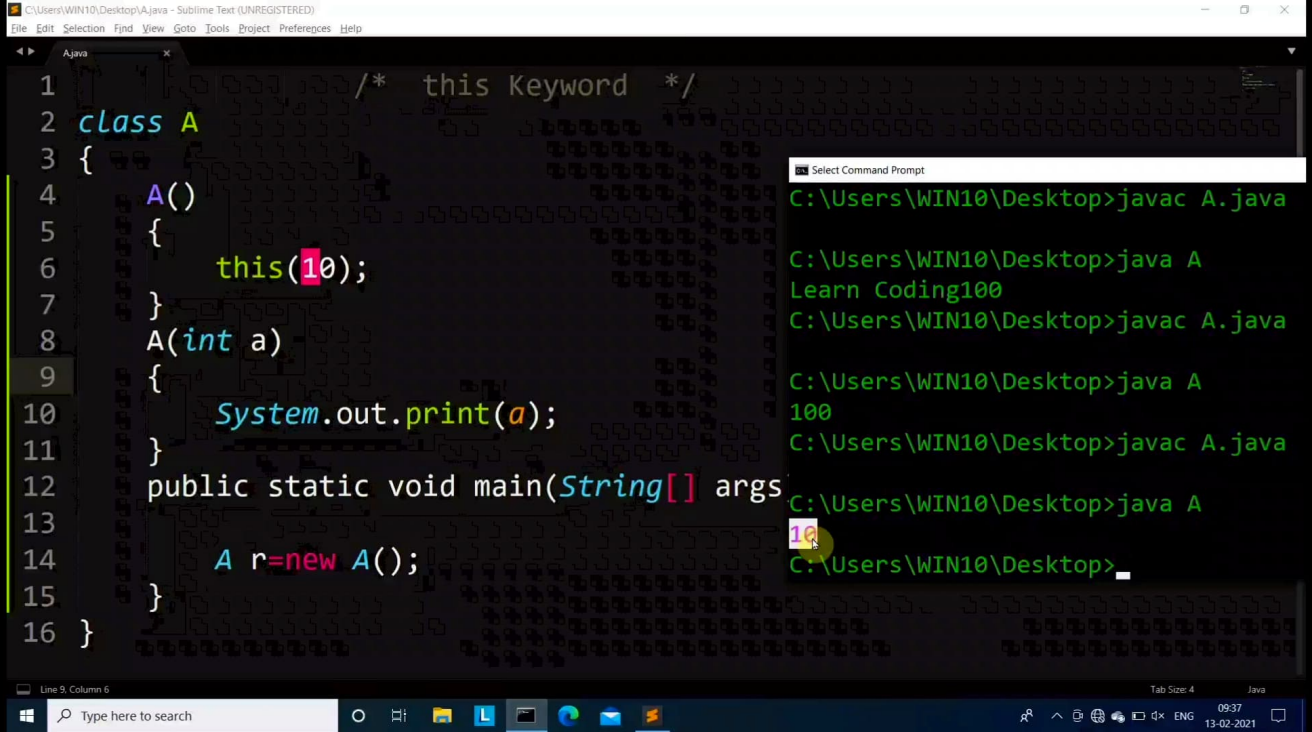

C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
A()
{
System.out.print("Learn Coding");
}
A(int a)
{
//this();
System.out.print(a);
}
public static void main(String[] args) {
A r=new A(100);
}
}
Line 10, Column 11
Tab Size: 4
Java
Type here to search
ENG
09:36
13-02-2021

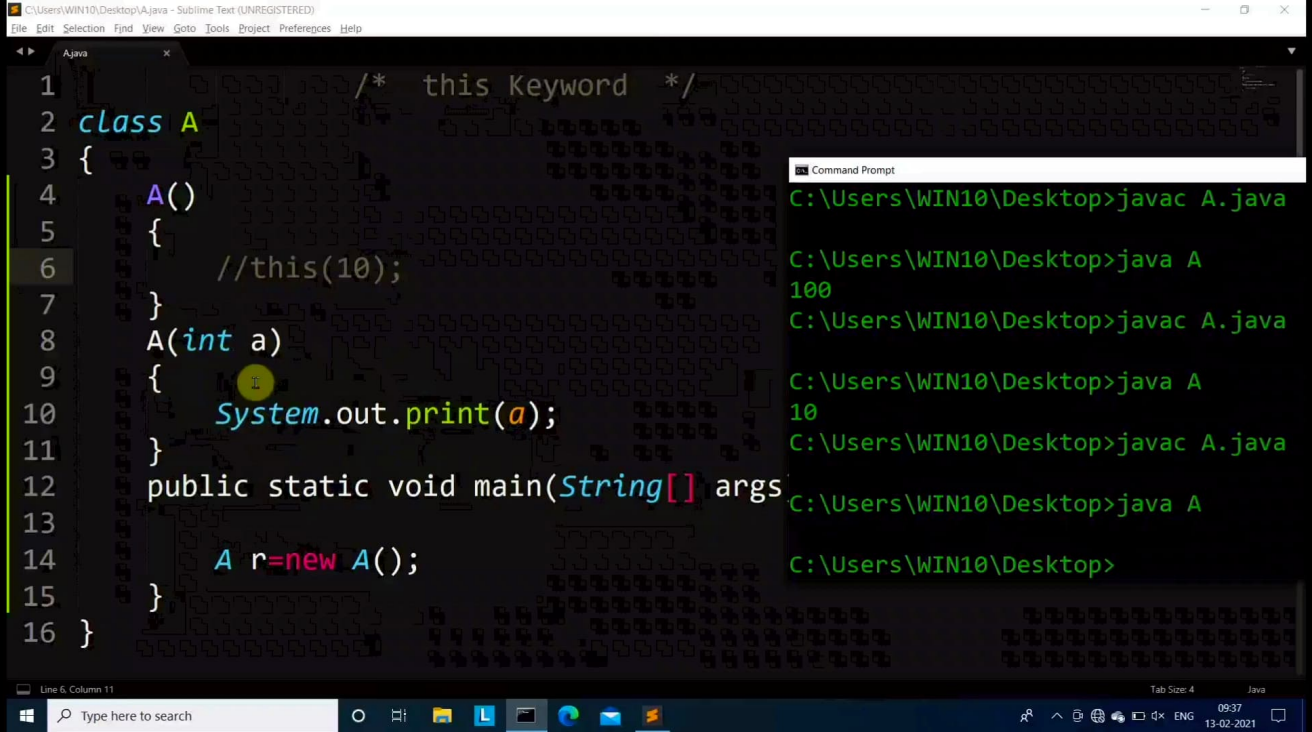
4) It also called parameterized constructor of its own class.

example :-

```
class A.
{   A()
    {   this(10);
    }
    A(int a)
    {
```


C:\Users\WIN10\Desktop\A.java • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
A()
{
System.out.print("Learn Coding");
}
A(int a)
{
this();
System.out.print(a);
}
public static void main(String[] args) {
A r=new A(100);
}
}
7 characters selected
Tab Size: 4
Java
Type here to search
ENG
09:36
13-02-2021

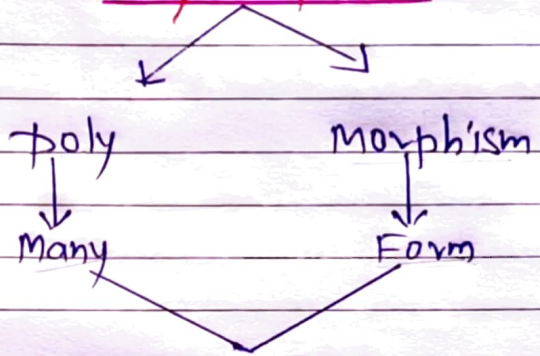

C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
A()
{
this(10);
}
A(int a)
{
System.out.print(a);
}
public static void main(String[] args
A r=new A();
}
}
Line 9, Column 6
Tab Size: 4
Java
Select Command Prompt
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
Learn Coding100
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
100
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
10
C:\Users\WIN10\Desktop>
Type here to search
ENG
09:37
13-02-2021


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* this Keyword */
class A
{
A()
{
//this(10);
}
A(int a)
{
System.out.print(a);
}
public static void main(String[] args
A r=new A();
}
}
Command Prompt
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
100
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
10
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
C:\Users\WIN10\Desktop>
Line 6, Column 11
Tab Size: 4
Java
Type here to search
ENG
09:37
13-02-2021

V.V.I

Page No. :
Date : / /

Polymorphism

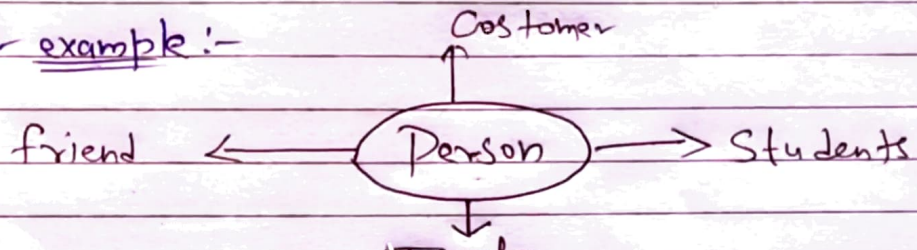


Polymorphism is the greek word whose meaning is "Same (Object having different

SUBSCRIBE

Polymorphism is the greek word whose meaning is "Same object having different behaviour".

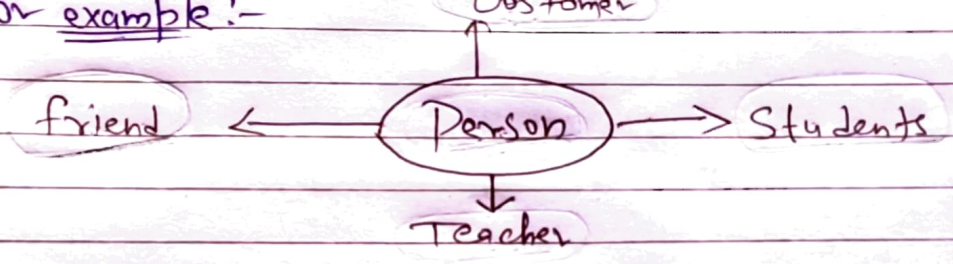
For example :-



i) void person(Teacher)
ii) void p(S)tents)
iii) v(d)

SUBSCRIBE

For example :-



i) void person(Teacher)
ii) void person (Students)
iii) void person (friend)
iv) void person (customer)

SUBSCRIBE

Page No. :
Date : / /

Polymorphism

V.V.I

Types

- Compile time polymorphism
- Runtime polymorphism

SUBSCRIBE

V.V.I

Polymorphism

Page No. :
Date : / /

1) Compile time polymorphism :-

A polymorphism which is exists ~~at~~ at the time of compilation is called Compile time or early binding or Static polymorphism.

example :- Method overloading

↓

Whenever a

SUBSCRIBE

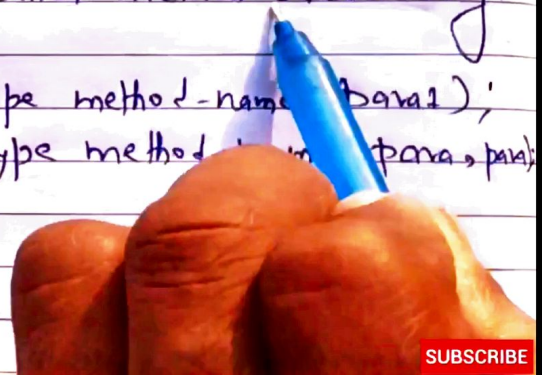
· early binding or Static polymorphism.

example :- Method overloading

↓

Whenever a class contain more than one method with same name and different types of parameters called method overloading.

Syntax :- return-type method-name(para1);
return-type method-name(para1,para2);



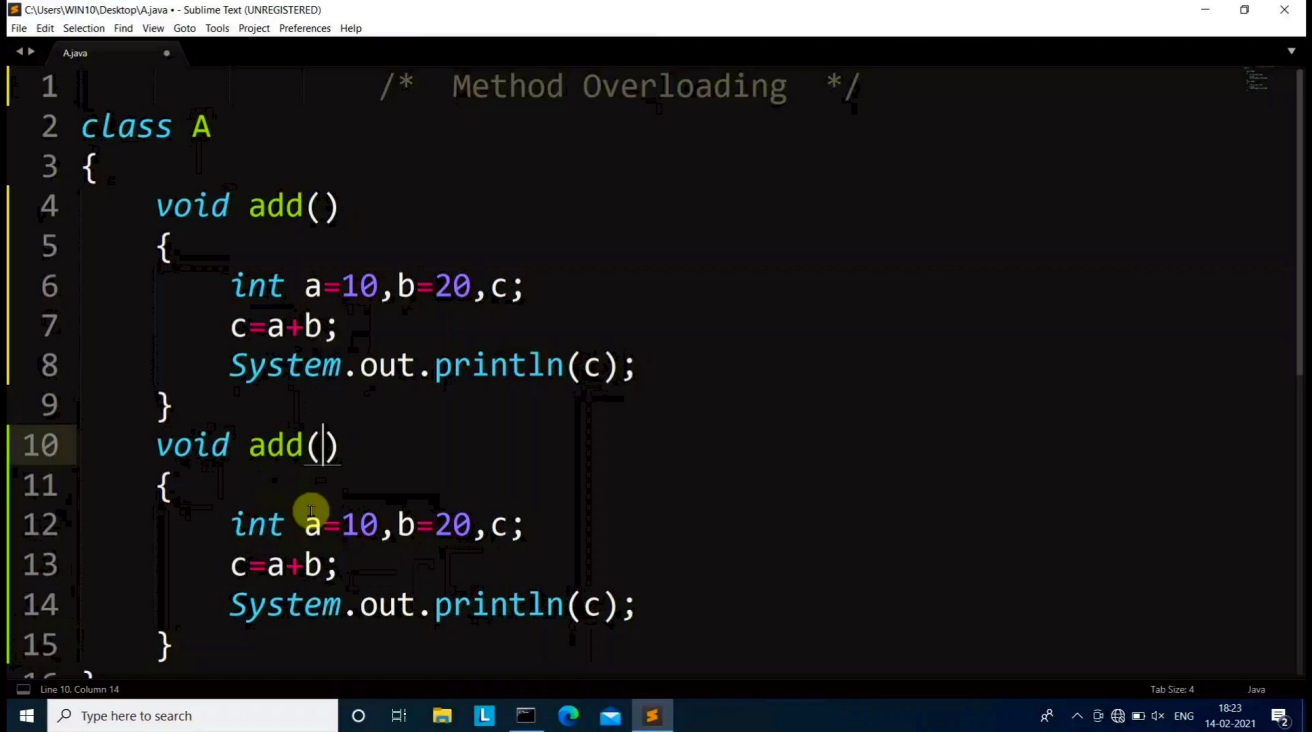
example :- Method Overloading

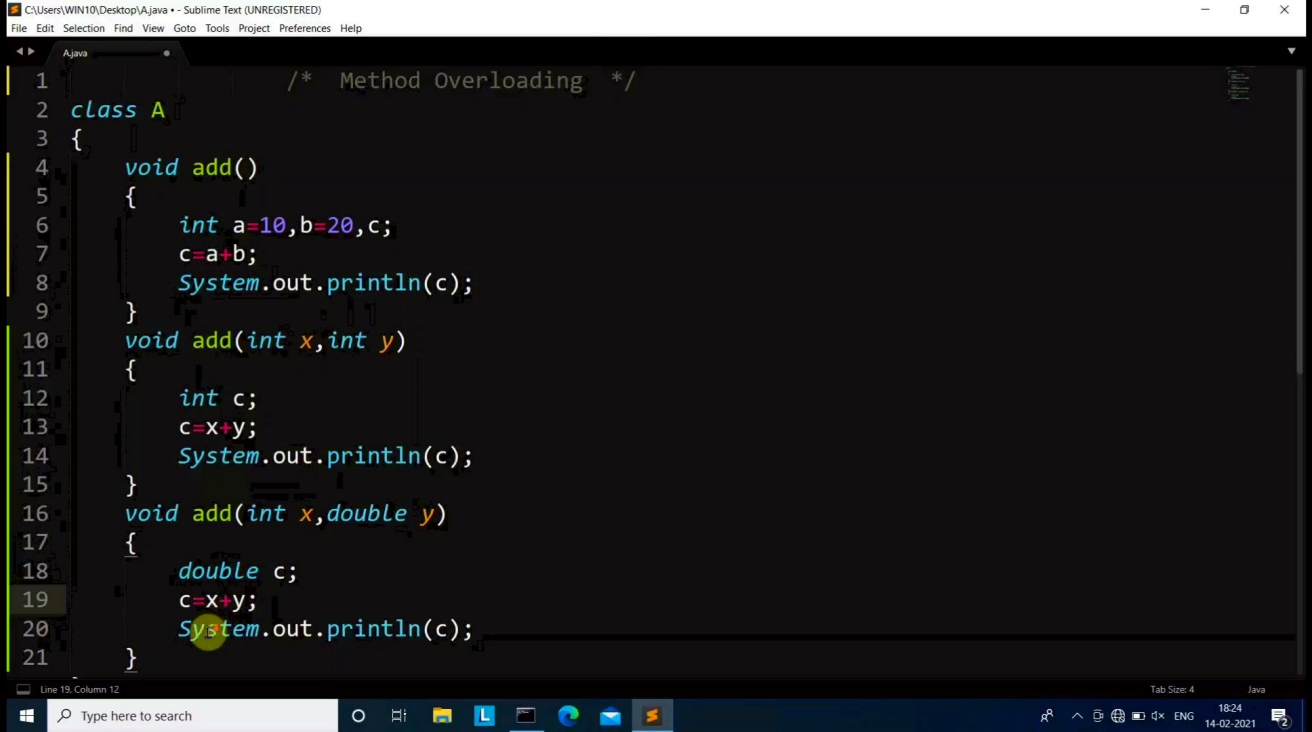
↓

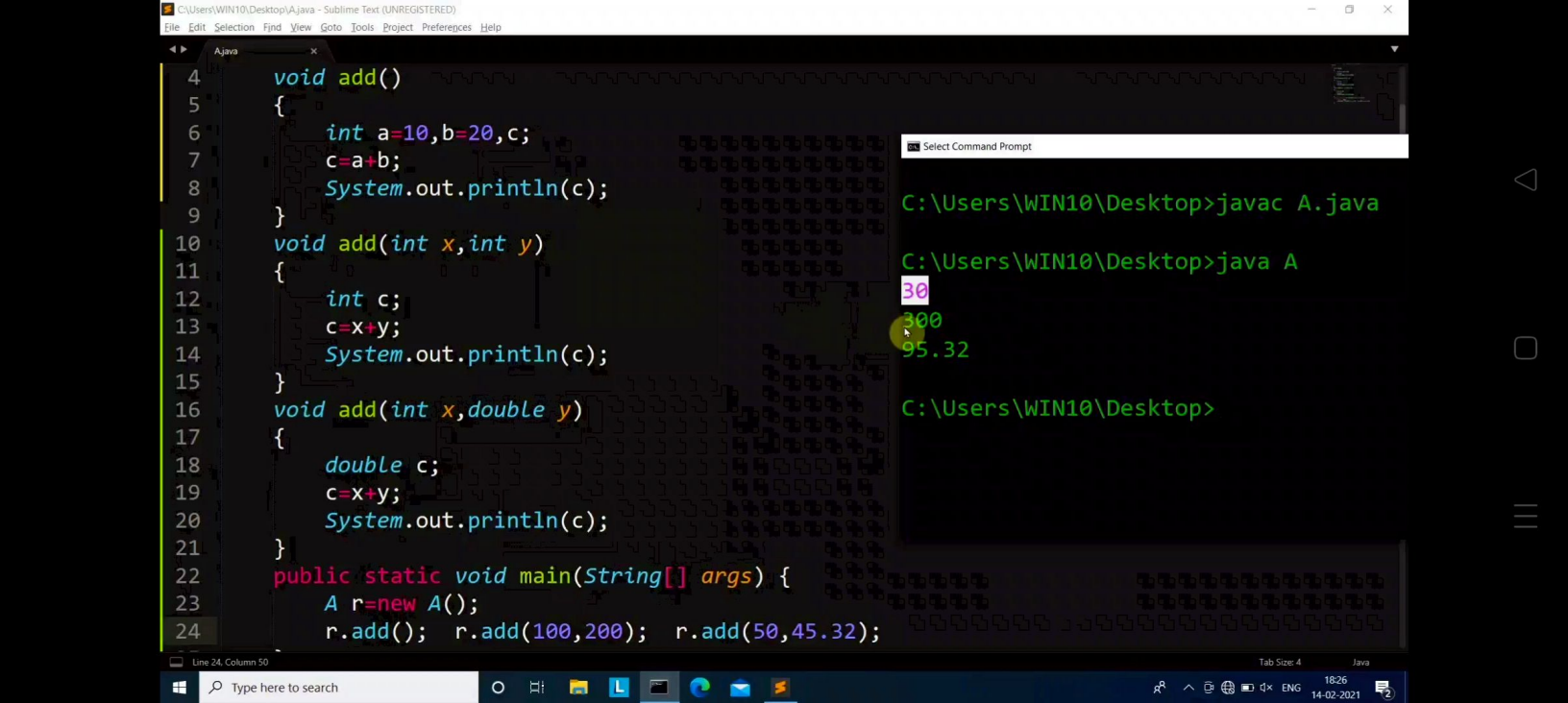
Whenever a class contain more than one method with same name and different types of parameters called method overloading.

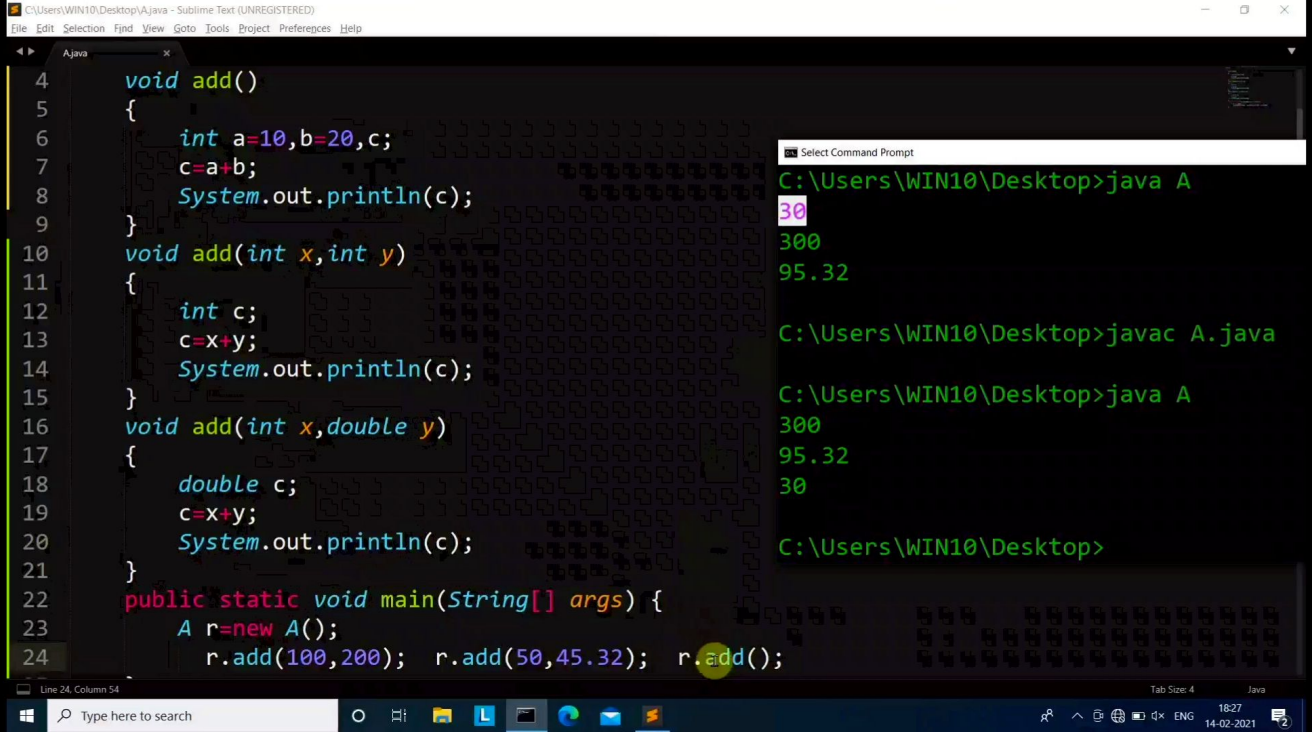
Syntax :- return-type method-name (para1);
return-type method-name (para, para);

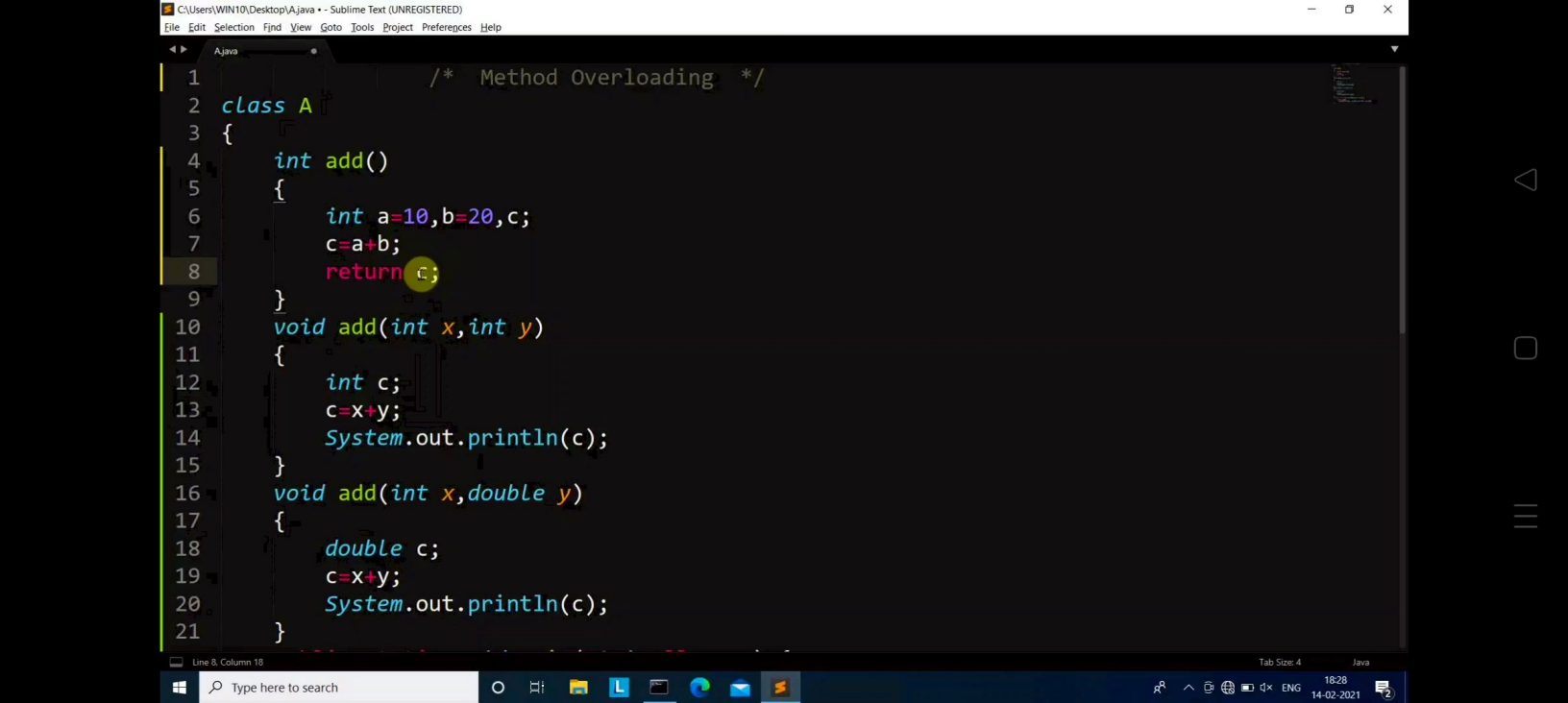
SUBSCRIBE

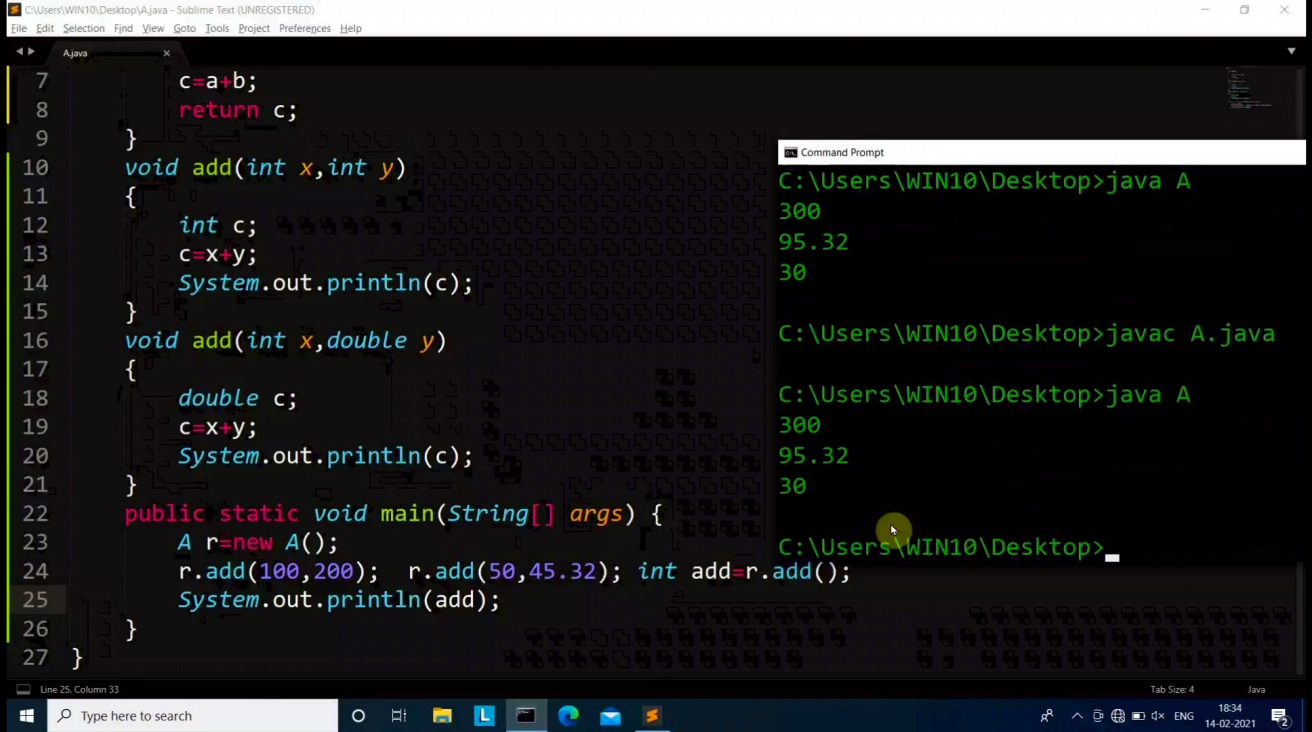

C:\Users\WIN10\Desktop\A.java • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* Method Overloading */
class A
{
void add()
{
int a=10,b=20,c;
c=a+b;
System.out.println(c);
}
void add()
{
int a=10,b=20,c;
c=a+b;
System.out.println(c);
}
Line 10, Column 14
Tab Size: 4
Java
Type here to search
ENG
18:23
14-02-2021


C:\Users\WIN10\Desktop\A.java • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* Method Overloading */
class A
{
void add()
{
int a=10,b=20,c;
c=a+b;
System.out.println(c);
}
void add(int x,int y)
{
int c;
c=x+y;
System.out.println(c);
}
void add(int x,double y)
{
double c;
c=x+y;
System.out.println(c);
}
Line 19, Column 12
Tab Size: 4
Java
Type here to search
ENG
18:24
14-02-2021


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
4 void add()
5 {
6 int a=10,b=20,c;
7 c=a+b;
8 System.out.println(c);
9 }
10 void add(int x,int y)
11 {
12 int c;
13 c=x+y;
14 System.out.println(c);
15 }
16 void add(int x,double y)
17 {
18 double c;
19 c=x+y;
20 System.out.println(c);
21 }
22 public static void main(String[] args) {
23 A r=new A();
24 r.add(); r.add(100,200); r.add(50,45.32);
Select Command Prompt
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
30
300
95.32
C:\Users\WIN10\Desktop>
Line 24, Column 50
Tab Size: 4
Java
Type here to search
ENG
18:26
14-02-2021


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
4 void add()
5 {
6 int a=10,b=20,c;
7 c=a+b;
8 System.out.println(c);
9 }
10 void add(int x,int y)
11 {
12 int c;
13 c=x+y;
14 System.out.println(c);
15 }
16 void add(int x,double y)
17 {
18 double c;
19 c=x+y;
20 System.out.println(c);
21 }
22 public static void main(String[] args) {
23 A r=new A();
24 r.add(100,200); r.add(50,45.32); r.add();
Line 24, Column 54
Tab Size: 4
Java
Select Command Prompt
C:\Users\WIN10\Desktop>java A
30
300
95.32
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
300
95.32
30
C:\Users\WIN10\Desktop>
Type here to search
ENG
18:27
14-02-2021


C:\Users\WIN10\Desktop\A.java • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
/* Method Overloading */
class A
{
int add()
{
int a=10,b=20,c;
c=a+b;
return c;
}
void add(int x,int y)
{
int c;
c=x+y;
System.out.println(c);
}
void add(int x,double y)
{
double c;
c=x+y;
System.out.println(c);
}
Line 8, Column 18
Tab Size: 4
Java
Type here to search
ENG
18:28
14-02-2021


C:\Users\WIN10\Desktop\A.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
A.java
c=a+b;
return c;
}
void add(int x,int y)
{
int c;
c=x+y;
System.out.println(c);
}
void add(int x,double y)
{
double c;
c=x+y;
System.out.println(c);
}
public static void main(String[] args) {
A r=new A();
r.add(100,200); r.add(50,45.32); int add=r.add();
System.out.println(add);
}
}
Line 25, Column 33
Tab Size: 4
Java
Command Prompt
C:\Users\WIN10\Desktop>java A
300
95.32
30
C:\Users\WIN10\Desktop>javac A.java
C:\Users\WIN10\Desktop>java A
300
95.32
30
C:\Users\WIN10\Desktop>
Type here to search
ENG
18:34
14-02-2021

V.V.I

Page No. :
Date : / /

Runtime Polymorphism

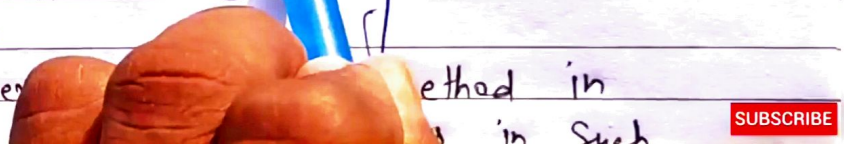
Q. What is runtime polymorphism? in Detail.

Ans→ A polymorphism which exists at the time of execution of program is Called runtime polymorphism.

Example:— method overriding.

↓

Whenever [illegible] method in [illegible]



SUBSCRIBE

time of execution of program is Called runtime polymorphism.

Example:- method overriding.

Whenever we writing method in super and sub classes in such a way that method name and parameter must be same Called method overriding.

Syntax:- class A
{
void

Syntax:-

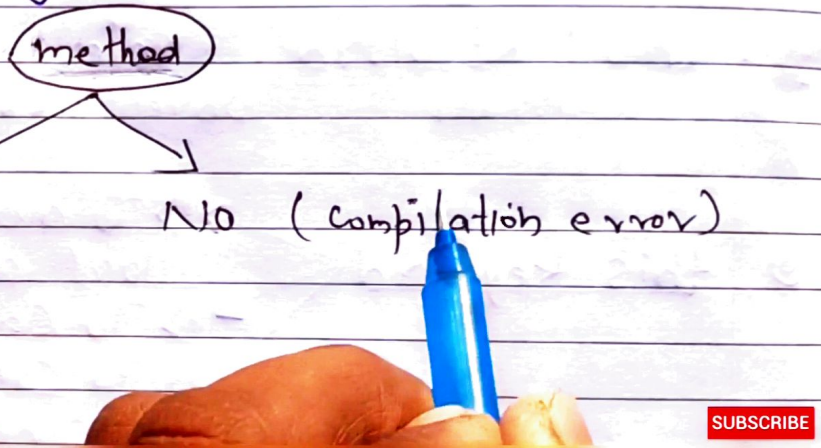
```
class A
{
    void Show()
    {
    }
}
class B extends A
{
    void Show()
    {
    }
}
```

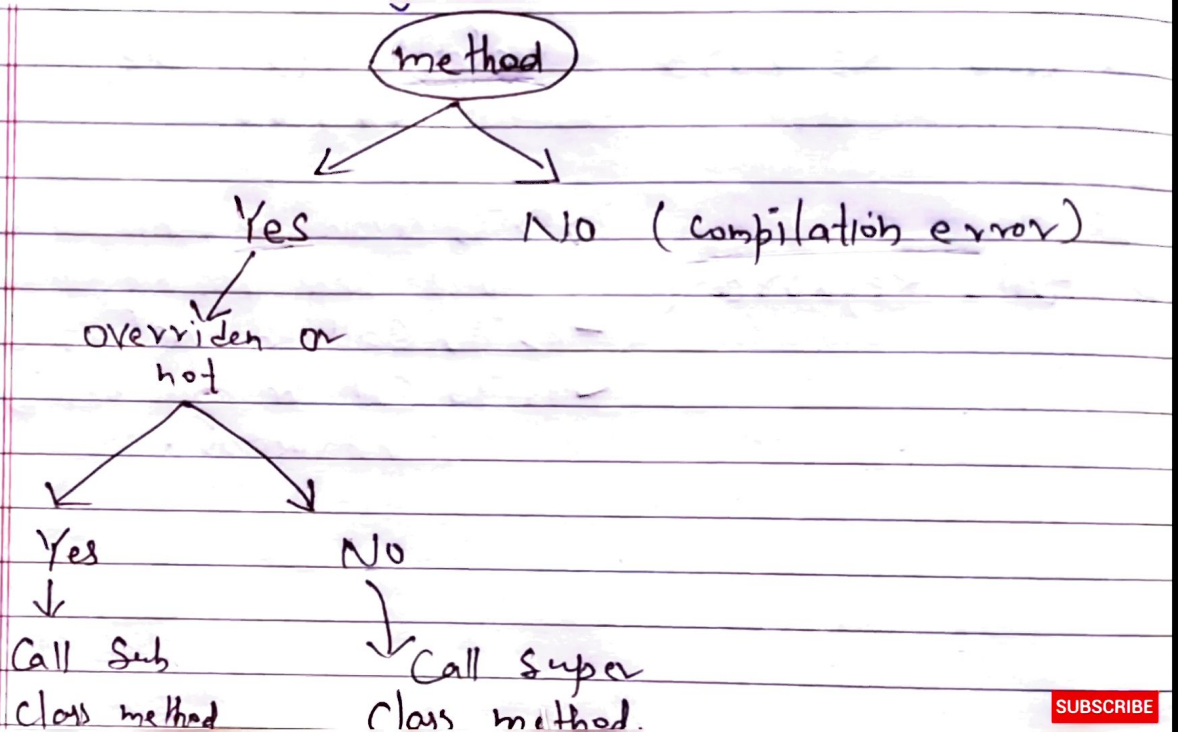
SUBSCRIBE

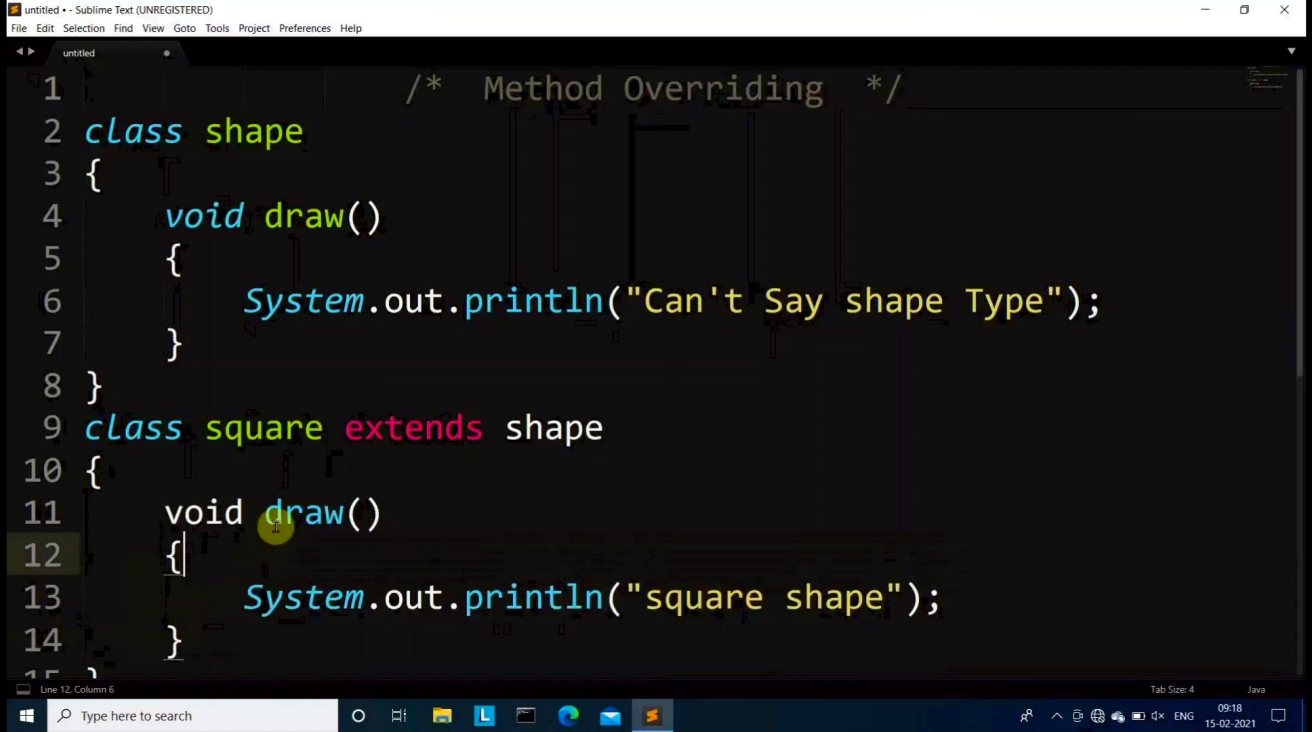
Page No. :
Date : / /

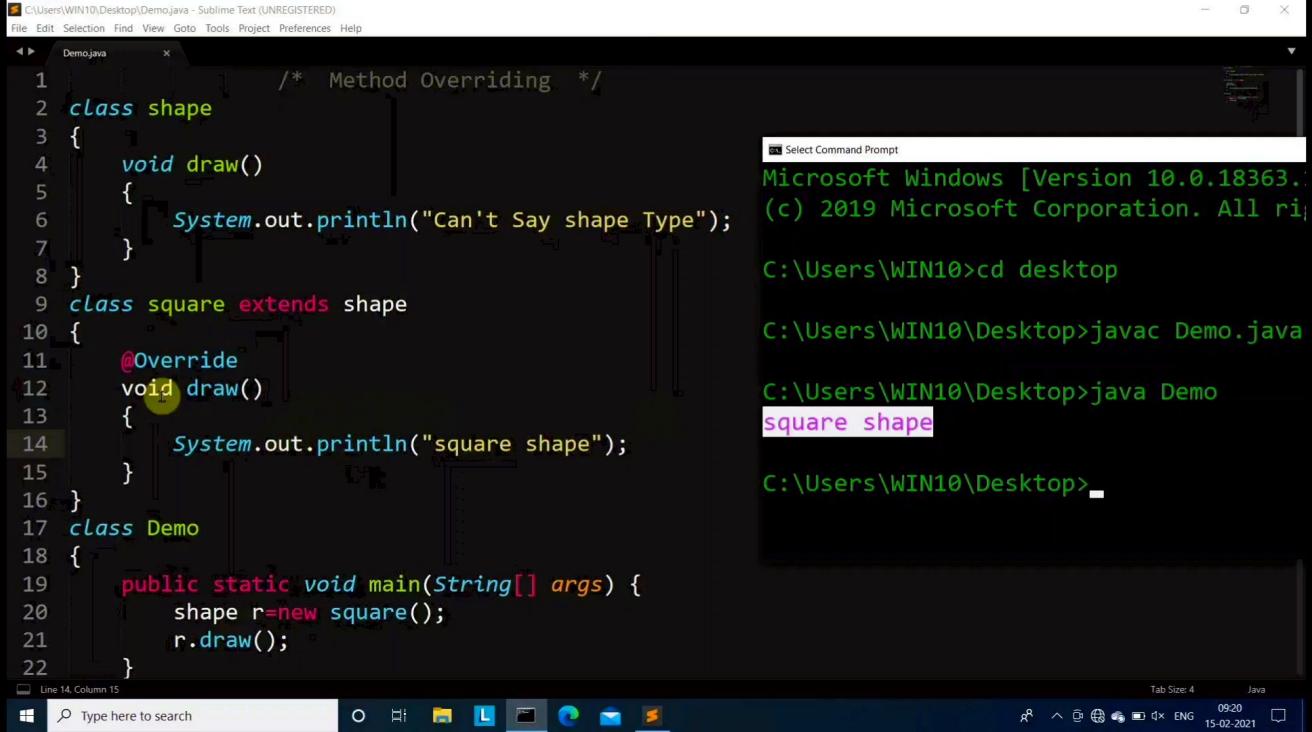
Polymorphism

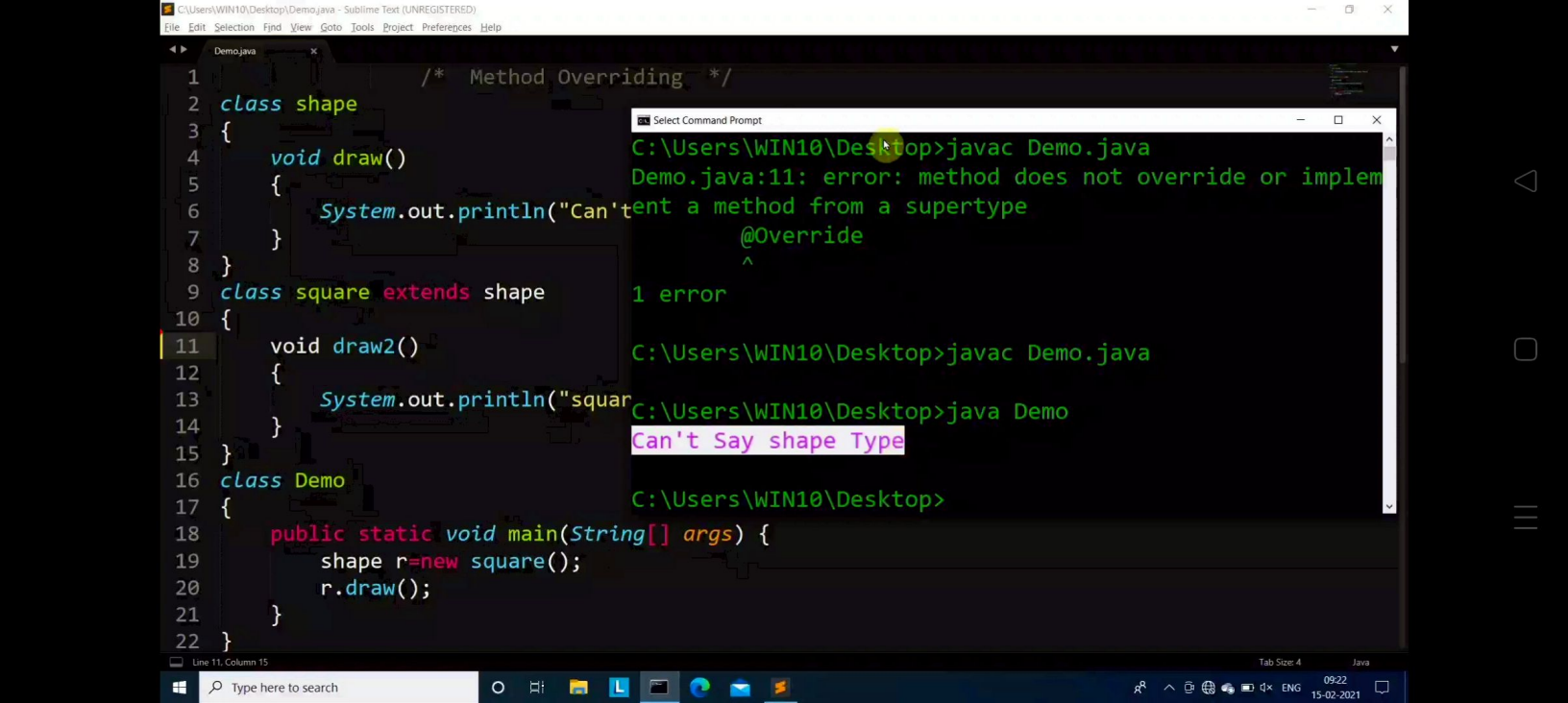
⊙ Method overriding rules :-

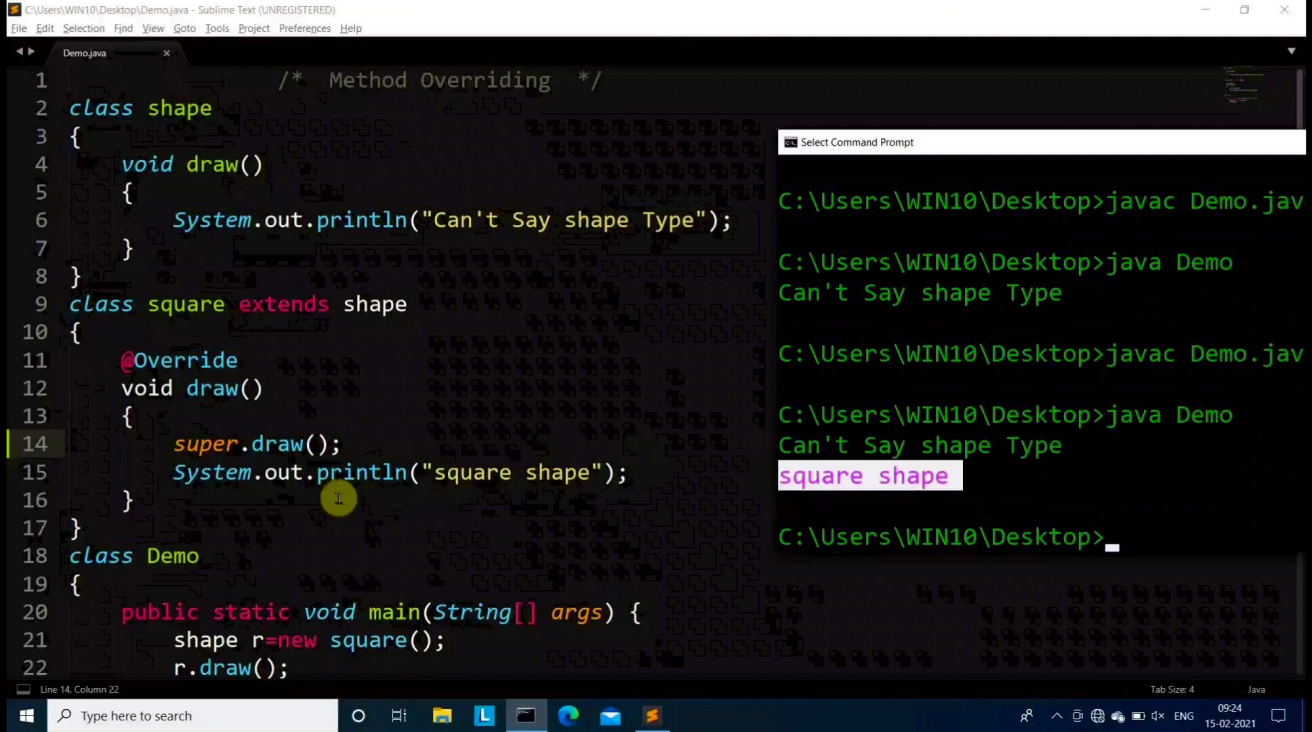



method
Yes
No (compilation error)
overriden or not
Yes
No
Call Sub class method
Call super class method.


untitled • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
untitled
/* Method Overriding */
class shape
{
void draw()
{
System.out.println("Can't Say shape Type");
}
}
class square extends shape
{
void draw()
{
System.out.println("square shape");
}
Line 12, Column 6
Tab Size: 4
Java
Type here to search
ENG
09:18
15-02-2021


C:\Users\WIN10\Desktop\Demo.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Demo.java
/* Method Overriding */
class shape
{
void draw()
{
System.out.println("Can't Say shape Type");
}
}
class square extends shape
{
@Override
void draw()
{
System.out.println("square shape");
}
}
class Demo
{
public static void main(String[] args) {
shape r=new square();
r.draw();
}
Select Command Prompt
Microsoft Windows [Version 10.0.18363.
(c) 2019 Microsoft Corporation. All ri
C:\Users\WIN10>cd desktop
C:\Users\WIN10\Desktop>javac Demo.java
C:\Users\WIN10\Desktop>java Demo
square shape
C:\Users\WIN10\Desktop>
Line 14, Column 15
Tab Size: 4
Java
Type here to search
ENG
09:20
15-02-2021


C:\Users\WIN10\Desktop\Demo.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Demo.java
/* Method Overriding */
class shape
{
void draw()
{
System.out.println("Can't
}
}
class square extends shape
{
void draw2()
{
System.out.println("squar
}
}
class Demo
{
public static void main(String[] args) {
shape r=new square();
r.draw();
}
}
Line 11, Column 15
Tab Size: 4
Java
Select Command Prompt
C:\Users\WIN10\Desktop>javac Demo.java
Demo.java:11: error: method does not override or implem
ent a method from a supertype
@Override
^
1 error
C:\Users\WIN10\Desktop>javac Demo.java
C:\Users\WIN10\Desktop>java Demo
Can't Say shape Type
C:\Users\WIN10\Desktop>
Type here to search
ENG
09:22
15-02-2021


C:\Users\WIN10\Desktop\Demo.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Demo.java
/* Method Overriding */
class shape
{
void draw()
{
System.out.println("Can't Say shape Type");
}
}
class square extends shape
{
@Override
void draw()
{
super.draw();
System.out.println("square shape");
}
}
class Demo
{
public static void main(String[] args) {
shape r=new square();
r.draw();
Select Command Prompt
C:\Users\WIN10\Desktop>javac Demo.jav
C:\Users\WIN10\Desktop>java Demo
Can't Say shape Type
C:\Users\WIN10\Desktop>javac Demo.jav
C:\Users\WIN10\Desktop>java Demo
Can't Say shape Type
square shape
C:\Users\WIN10\Desktop>
Line 14, Column 22
Tab Size: 4
Java
Type here to search
ENG
09:24
15-02-2021

Date : / /

V.V.I

Encapsulation

Q. What is Encapsulation? full explaination.

Ans→ Encapsulation is a mechanism through which ~~is~~ we can wrapping the data members and member methods of class in a single unit called encapsulation.

Note:- 1) Declare the class variables as a private.

2) Declare the class me
public.

SUBSCRIBE

members and member methods of class in a single unit called encapsulation.

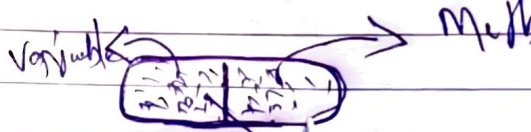
Note:- 1) Declare the class variables as a private.

2) Declare the class methods as a public.

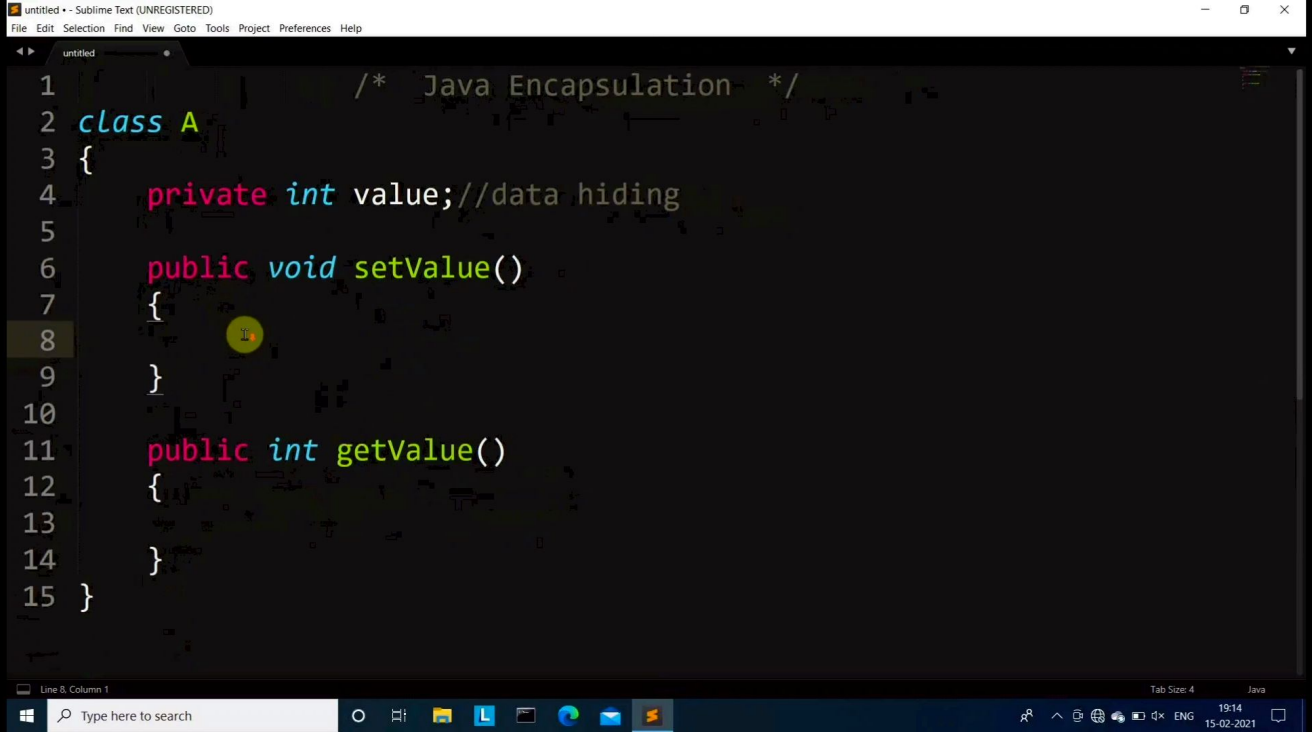
Example:- Class is the best example of encapsulation.

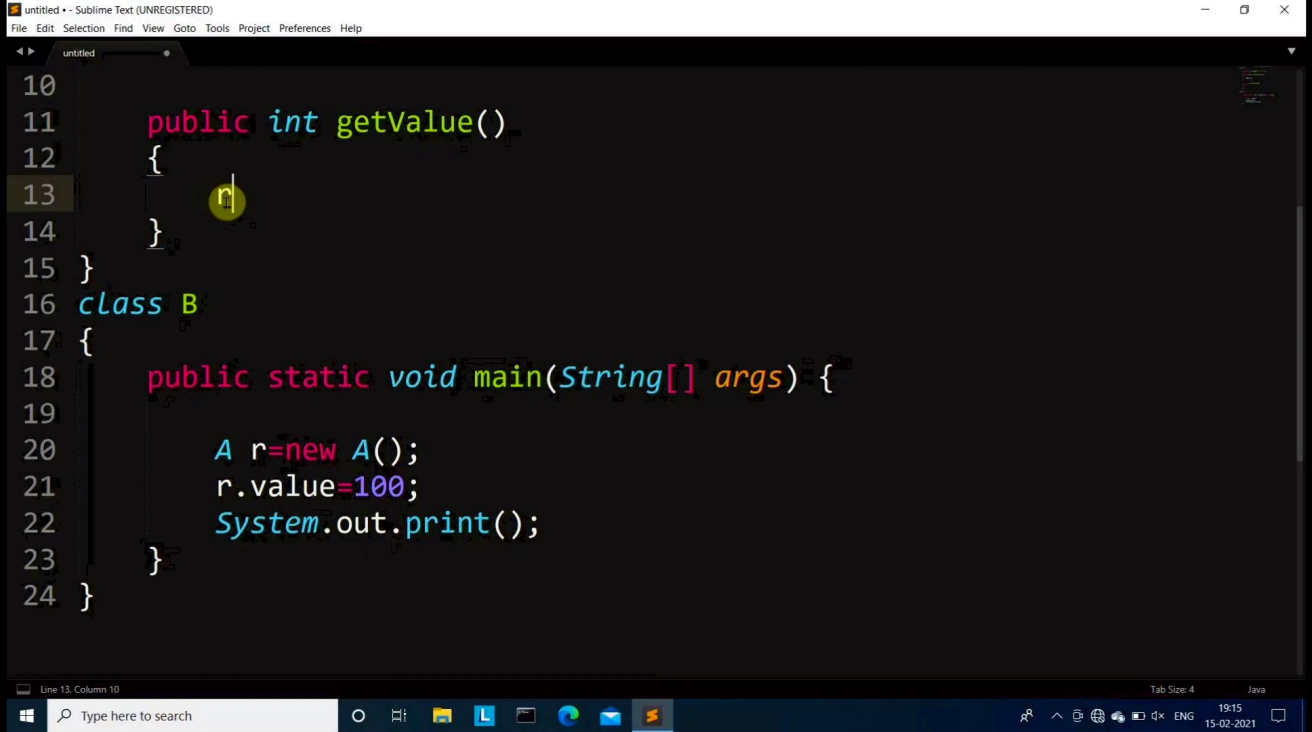
2) Declare the class methods as a public.

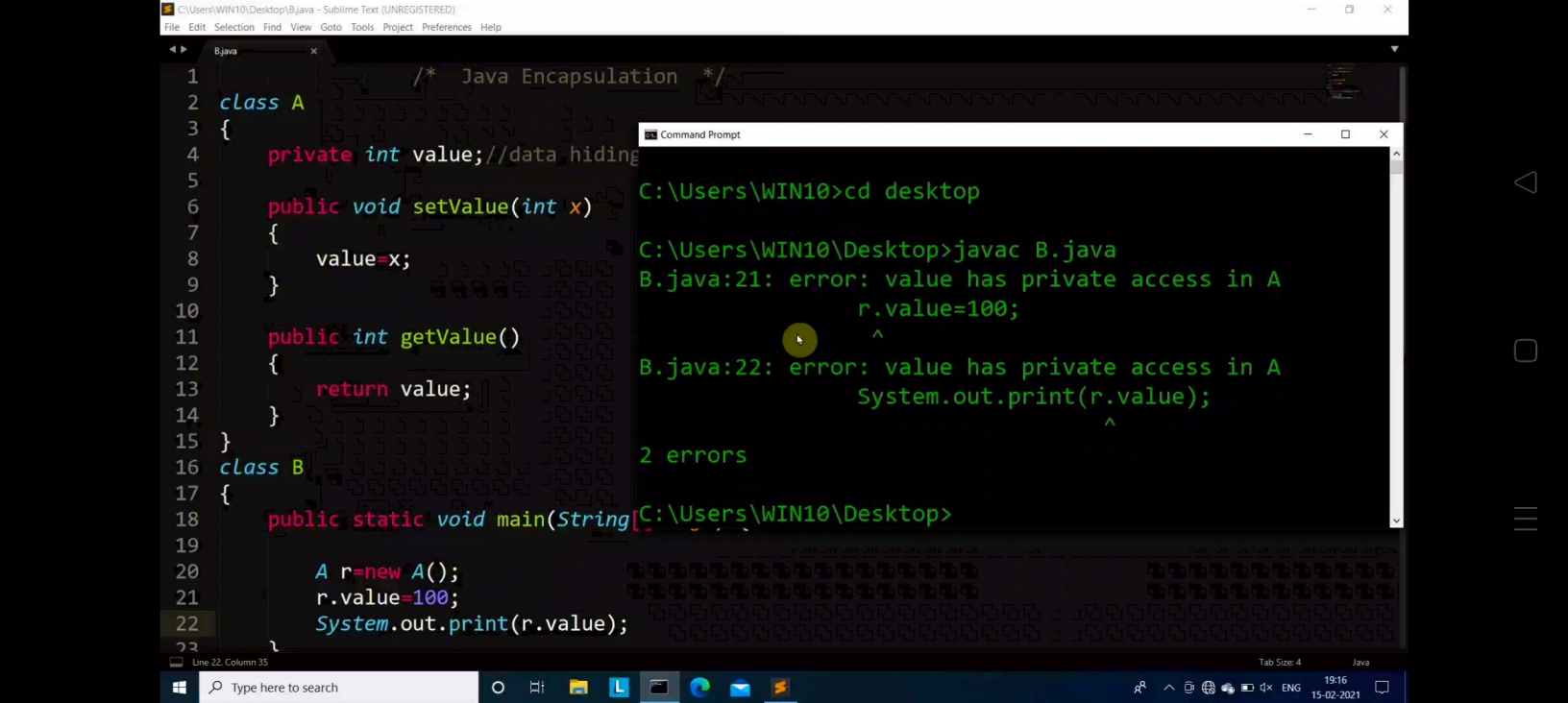
Example:- class is the best example of encapsulation.

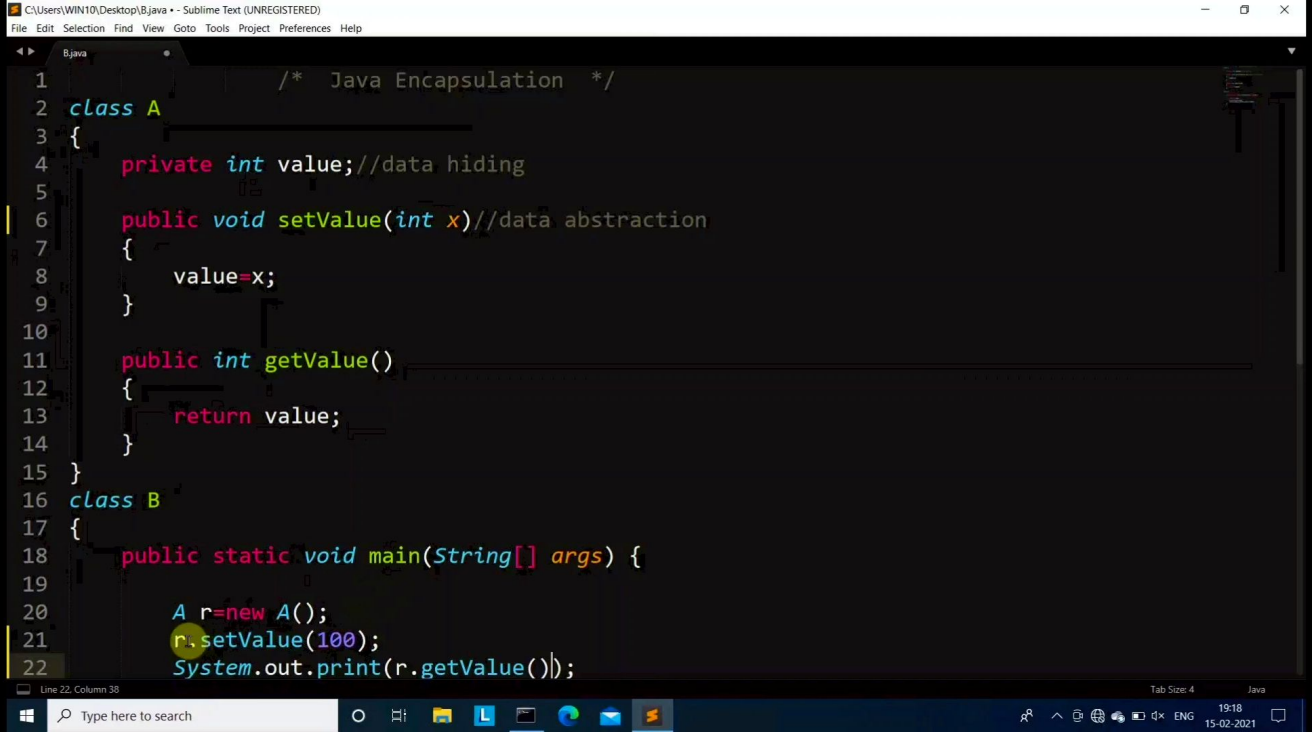


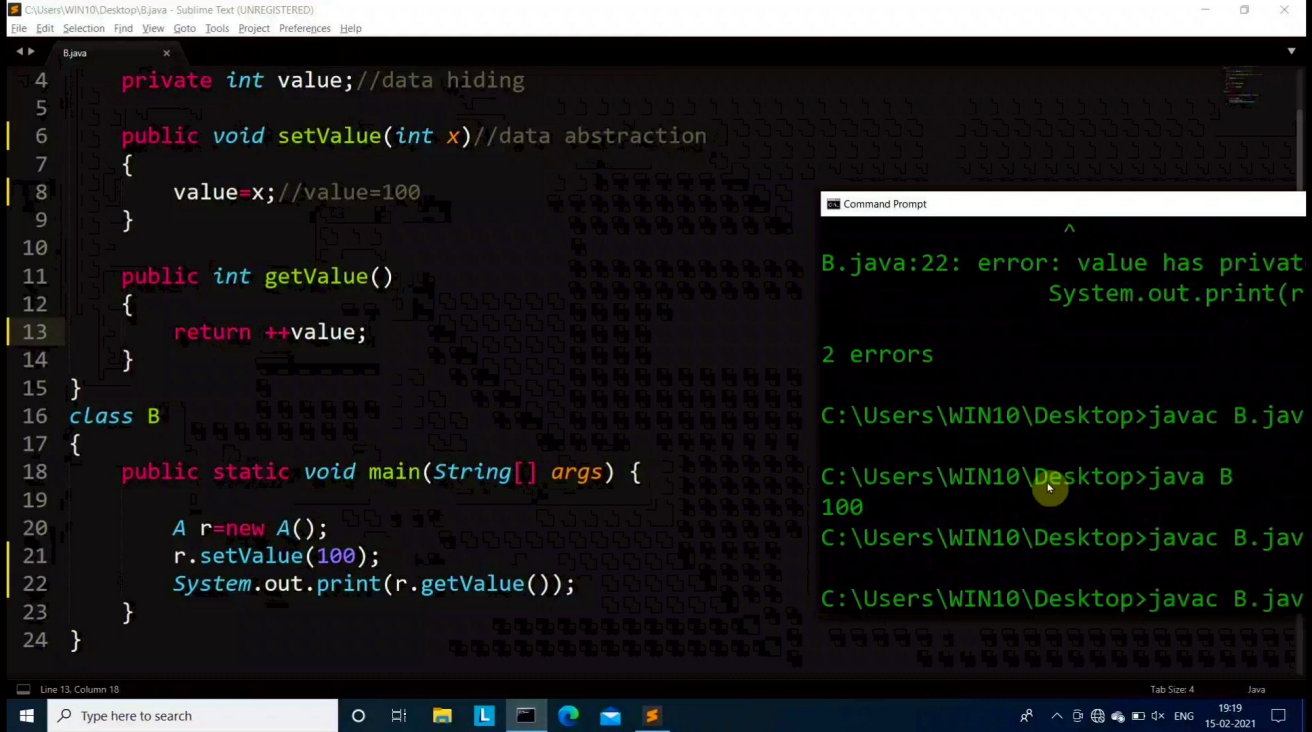
SUBSCRIBE


untitled • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
untitled
/* Java Encapsulation */
class A
{
private int value;//data hiding
public void setValue()
{
}
public int getValue()
{
}
}
Line 8, Column 1
Tab Size: 4
Java
Type here to search
ENG
19:14
15-02-2021


untitled • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
untitled
10
11 public int getValue()
12 {
13 r
14 }
15 }
16 class B
17 {
18 public static void main(String[] args) {
19
20 A r=new A();
21 r.value=100;
22 System.out.print();
23 }
24 }
Line 13, Column 10
Tab Size: 4
Java
Type here to search
ENG
19:15
15-02-2021


C:\Users\WIN10\Desktop\B.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
B.java
/* Java Encapsulation */
class A
{
private int value;//data hiding
public void setValue(int x)
{
value=x;
}
public int getValue()
{
return value;
}
}
class B
{
public static void main(String[
A r=new A();
r.value=100;
System.out.print(r.value);
Command Prompt
C:\Users\WIN10>cd desktop
C:\Users\WIN10\Desktop>javac B.java
B.java:21: error: value has private access in A
r.value=100;
^
B.java:22: error: value has private access in A
System.out.print(r.value);
^
2 errors
C:\Users\WIN10\Desktop>
Line 22, Column 35
Tab Size: 4
Java
Type here to search
ENG
19:16
15-02-2021


C:\Users\WIN10\Desktop\B.java • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
B.java
/* Java Encapsulation */
class A
{
private int value;//data hiding
public void setValue(int x)//data abstraction
{
value=x;
}
public int getValue()
{
return value;
}
}
class B
{
public static void main(String[] args) {
A r=new A();
r.setValue(100);
System.out.print(r.getValue());
Line 22, Column 38
Tab Size: 4
Java
Type here to search
ENG
19:18
15-02-2021


C:\Users\WIN10\Desktop\B.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
B.java
4 private int value;//data hiding
5
6 public void setValue(int x)//data abstraction
7 {
8 value=x;//value=100
9 }
10
11 public int getValue()
12 {
13 return ++value;
14 }
15 }
16 class B
17 {
18 public static void main(String[] args) {
19
20 A r=new A();
21 r.setValue(100);
22 System.out.print(r.getValue());
23 }
24 }
Command Prompt
^
B.java:22: error: value has privat
System.out.print(r
2 errors
C:\Users\WIN10\Desktop>javac B.jav
C:\Users\WIN10\Desktop>java B
100
C:\Users\WIN10\Desktop>javac B.jav
C:\Users\WIN10\Desktop>javac B.jav
Line 13, Column 18
Tab Size: 4
Java
Type here to search
ENG
19:19
15-02-2021