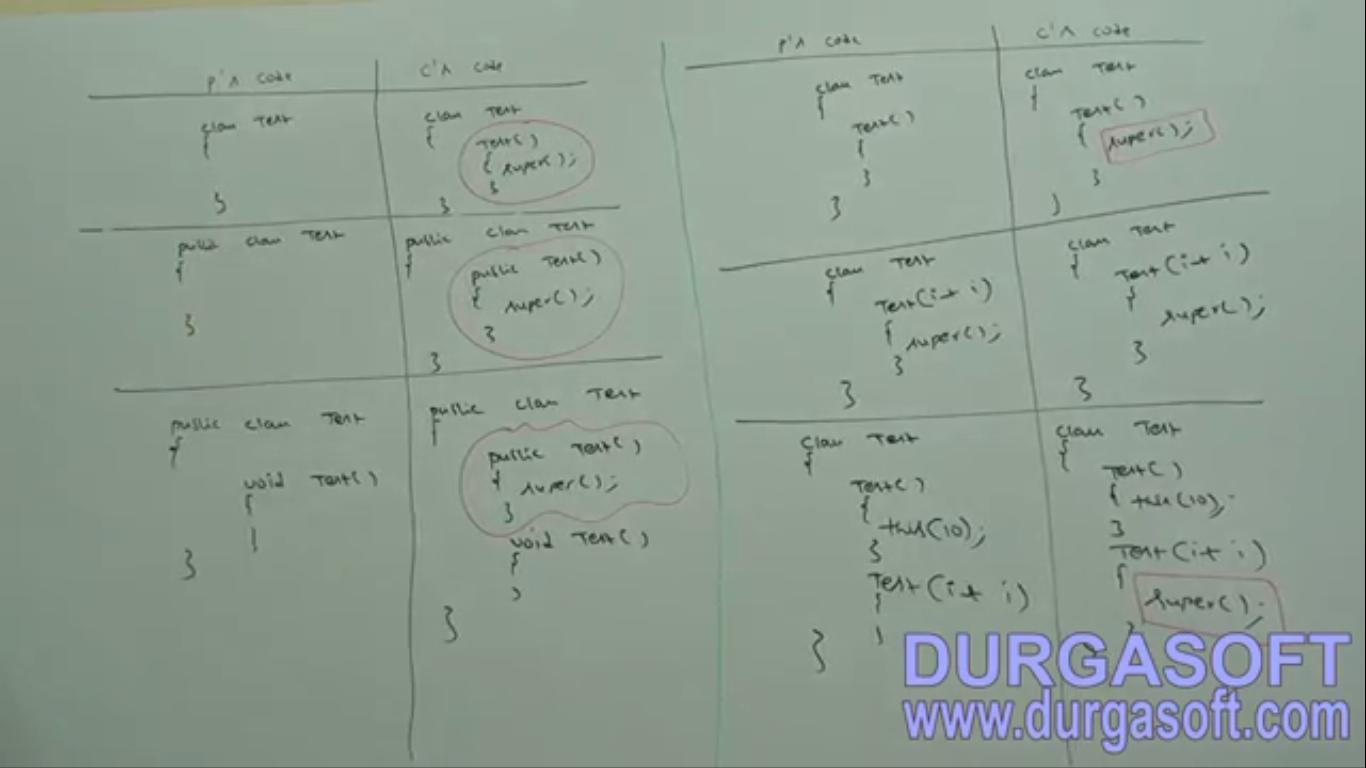
# What is the prototype of default constructor?

1. It’s always **no-arg** constructor.
2. The access modifier of default constructor is exactly same as that of containing class.
3. It contains only one line 🡪 super(); 🡪 call to super class’s “no-arg constructor”

**NOTE**: Only compiler provided constructor is default constructor.   
**Let’s practice the above concept** **NOTE:** super() and this() are possible in constructor only. If we try to use them in some other places then compile time error will be thrown.

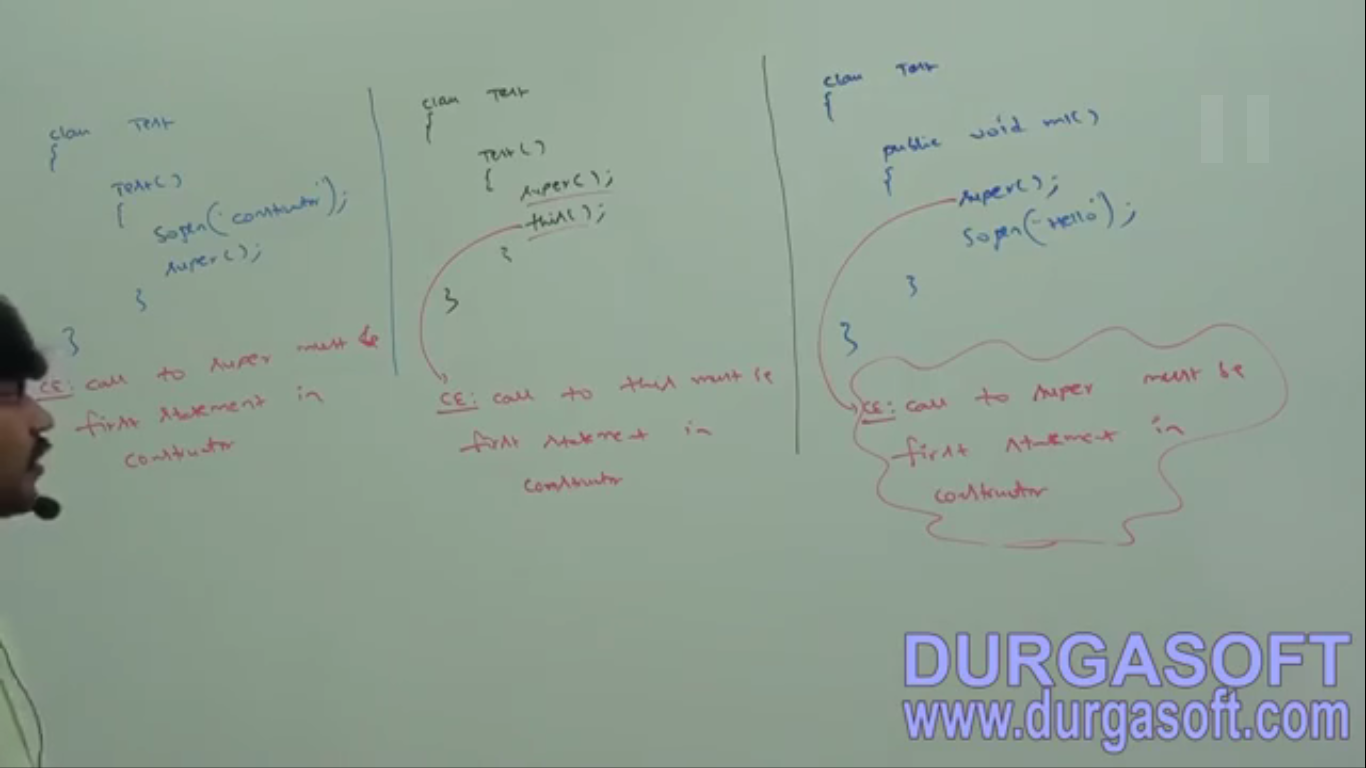
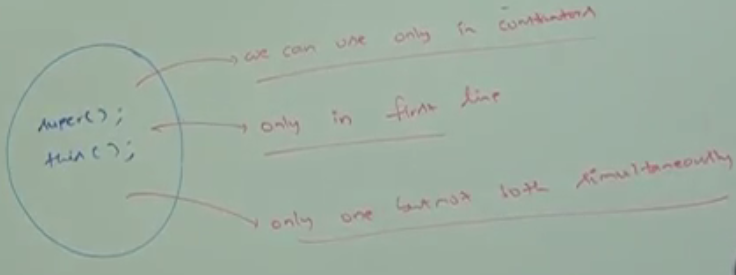
Compiler’s code

Compiler’s code

Programmer’s code

Programmer’s code

**Based on the above discussion, 3 cases spoute out**

super() is not 1st statement

this() and super() both simultaneously not possible

this is not constructor

|  |  |
| --- | --- |
| super(), this() | super, this |
| These are constructor calls to super class and current class | **Keywords**: used to super class and current class instance members not static members as this keyword and super keyword are available only in instance areas such as instance block and instance method |
| Can be used only in constructor as first line | Anywhere except static areas |
| Can be used only once. | Any # of times |