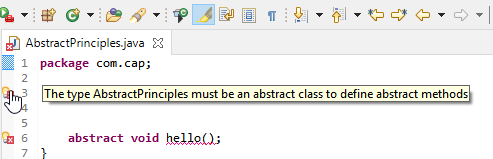
4: Abstract Class Principles

1. If a class has abstract methods then its class should be a Abstract class
2. **package** com.cap;
3. **class** AbstractPrinciples {
4. **abstract** **void** hello();
5. }



2. Abstract class cannot be instantiated

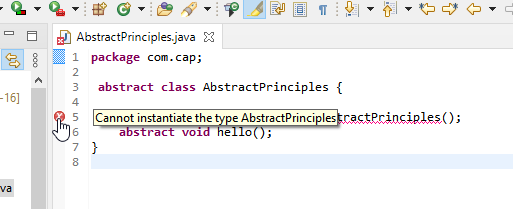
**package** com.cap;

**abstract** **class** AbstractPrinciples {

AbstractPrinciples obj=**new** AbstractPrinciples();

**abstract** **void** hello();

}



3: When we extend abstract class we must override all the methods of abstract class.

**package** com.cap;

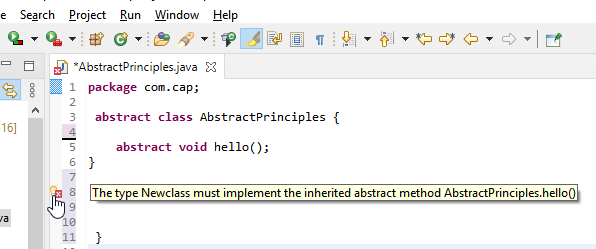
**abstract** **class** AbstractPrinciples {

**abstract** **void** hello();

}

**class** Newclass **extends** AbstractPrinciples{

}



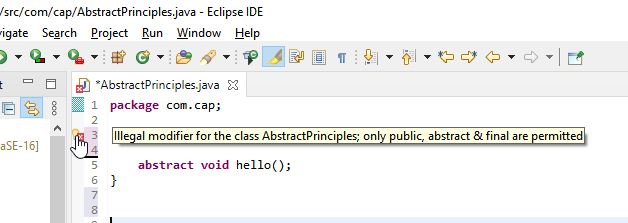
4. Abstract class cannot be private

**package** com.cap;

**private** **abstract** **class** AbstractPrinciples {

**abstract** **void** hello();

}



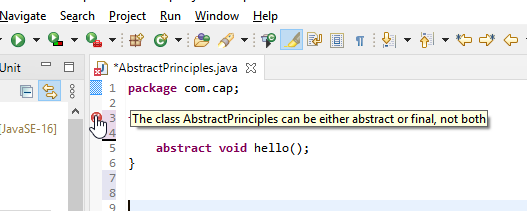
5. Abstract class cannot be final

**package** com.cap;

**final** **abstract** **class** AbstractPrinciples {

**abstract** **void** hello();

}



6.You can have an Abstract class without any abstract method in it.

**package** com.cap;

**abstract** **class** AbstractPrinciples {

**public** **void** hello() {

System.***out***.println("Hello World");

}

}