Singleton design pattern-

It means define the class which has single instance that provide the global point of access to it called as singleton design pattern. Why?

**package** com.test;

**public** **class** A {

}

**package** com.test;

**public** **class** Demo {

**public** **static** **void** main(String[] args) {

A a=**new** A();

A a1=**new** A();

System.***out***.println(a.hashCode());

System.***out***.println(a1.hashCode());

}

}

366712642

1829164700

**package** com.test;

**public** **class** A {

**static** A *aa*=**new** A();//static member of class

. **private** A(){//private constructor

}

**public** **static** A getObject() {//static method

**return** *aa*;

}

}

**package** com.test;

**public** **class** Demo {

**public** **static** **void** main(String[] args) {

A a=A.*getObject*();//call method

A a1=A.*getObject*();

System.***out***.println(a.hashCode());

System.***out***.println(a1.hashCode());

}

}

output

366712642

366712642