Abstraction:

* Hiding implementation and showing only the functionality to user.
* It can be achieved in 2 ways

1. **Abstract classes:**

* If a class is declared with abstract key word then that is called abstract class.
* It contains abstract methods and non-abstract methods.
* We cannot create objects for abstract classes.
* We can achieve 0-100% abstraction.

Ex: public abstract class Addition{

Public void hi(){

s.o.p(“hi”);

}

Public abstract void hello();

}

Public class Summation extends Addition {

Public void hello(){

…….

……..

}

}

1. **Interfaces**:

* It is also similar to class
* It contains only abstract methods and by default each method is having abstract keyword.
* We cannot create objects for interfaces.
* Here we will use “implements” key word for implementing interface.
* We can achieve multiple inheritance through interfaces.
* You can achieve loosely coupling.
* Here we can achieve 100% abstraction.

Note:

Class 🡺 class 🡺 extends keyword

Interface 🡺 class🡺 implements keyword.

Interface 🡺 interface🡺 extends keyword

Public interface Addition.{

Public void hi();

…..

…..

}

Public interface Summation extends Addition{

}

Ex: public interface Addition{

Public void hi();

}

Public class Summation implements Addition{

Public void hi(){

….

….

}

}

A 🡺 B 🡺 C 🡺 tightly coupled

Loosely coupled