|  |  |  |
| --- | --- | --- |
| OOPs Design Principles | | |
| Maximize Cohesion (Always Attempt High Cohesion) | | |
| Coupling between modules/components is their degree of mutual interdependence; lower coupling is better. In other words, coupling is the probability that code unit “B” will “break” after an unknown change to code unit “A”. | | |
| LOW COHESION | in general, Tight coupling means the two classes often change together. In other words, if A knows more than it should about the way in which B was implemented, then A and B are tightly coupled. | |
| HIGH COHESION | In simple words, loose coupling means they are mostly independent. If the only knowledge that class A has about class B, is what class B has exposed through its interface, then class A and class B are said to be loosely coupled. In order to overcome from the problems of tight coupling between objects, spring framework uses dependency injection mechanism with the help of POJO/POJI model and through dependency injection it’s possible to achieve loose coupling. | |
| DIFFERENCE BETWEEN HIGH AND LOW COHESION ? | |  |
| * High cohesion is when you have a class that does a well-defined job. Low cohesion is when a class does a lot of jobs that don’t have much in common. * High cohesion gives us better-maintaining facility and Low cohesion results in monolithic classes that are difficult to maintain, understand and reduce reusability. | | |