

the file system (i.e. we need to ~~use~~ ^{listen} file ~~system~~ from the configuration file then we've to use it).

* Ways of Injecting Dependencies:

→ Dependency Injection can be done in 2 ways i.e.

(i) Using Setter Injection

(ii) Using Constructor Injection

(i) **Setter Injection**: It sets the value. It automatically calls setter method for putting values.

e.g.

```
class Student
{
```

```
    id, name, address
```

```
    setId(id) { }
```

```
    setName(name) { }
```

```
    setAddress(address) { }
```

```
}
```

It will take id value & set it to id & same for the all.

[Value (to put)] While creating an object ^{of} student in that case ~~add~~ to set value in id, it will call setId(id) ~~add~~ method &

```
class Address
{
```

```
    Street, City, State, Country
```

```
    setStreet(Street)
```

```
    setCity(City)
```

```
    setState(State)
```

```
    setCountry(Country)
```

```
}
```

it will inject address object automatically in setAddress method in class Student.

& we'll get all the dependencies.

Then both the objects (Address & Student) are ready to use, we can remove it.

From the container.

(ii) Constructor Injection: While creating an object now it'll call constructor instead of setter.

eg: class Student
{

String id, name, address;

Constructor ← Student(id, name, address)
{
}
}

→ Configuration File: Spring has its own ~~xml file~~ configuration file (i.e. configuration file). ~~We just will inform everything to the container by~~ We have to provide the information of a class to IOC container then only it will create an object & will maintain the life cycle i.e. known as beans.

The class which we provide to IOC container that class is known as beans.

Where we declare beans & its dependency is known as configuration file.

Configuration file is an xml file where we declare our beans (java classes) & dependency (the class on which it depends on which things).

xml file starts with

