Dependency Injection: 🡺 IOC [inversion of Container]

Hibernate   
Plugins

Spring HAS-A [Aggregation]

Bean 🡺 Object 🡺 methods level 🡺 @Bean 🡺 Default singleton object

Component 🡺 Object 🡺 class level 🡺 @Component

* Spring is light weighted application

IOC:

Bean creation 🡺 Configuring the bean 🡺 Managing the bean till its usage

We can read data directly from config file/ property files.

A 🡺 B 🡺 C 🡺 D 🡺 Tightly coupled

A a = new A();

@Autowired

A a;

Class A {

}

Class B {

@Autowired

A a;

a.method();

}

Class Student{

Private int id;

Private String name;

Private Address addres;

//setter

//getter

}

Class Address{

Private int addressId;

Private Strnig plotNo;

//setter method

//getter method

}

.java 🡺 .class

Multiple .class 🡺 .jar file 1.6V,

Multiple .jar files 🡺 .war file

Multiple .war files 🡺 .ear file

Maven Build tool 🡺 POM.xml 🡺 .jar/.war files

Tomcat server: 8080

C 🡺 users 🡺 name 🡺 .m2 5.14 spring

Hibernate, mysql database,

* Dependency management 🡺
* Spring boot 2.5.6V
* Autoconfiguration 🡺 @EnableAutoConfiguration
* Embedded tomcat server and Maven
* Spring boot actuators 🡺 Monitoring tool /threads, /health

Hibernate Framework: Object Oreinted 🡺 ORM 🡺 Object Relational mapping

Dialects 🡺 mySQL

Ddl-auto: update

JDBC 🡺 backend database 🡺 Configuration, Queries

JPA Repository 🡺 specifications 🡺 Java persistence API

JPA query

JPQL query

SQL query   
HQL

@RequestMapping 🡺 urls related

@Controller 🡺

@ResponseBody 🡺 Java to Json object 🡺 HttpResponse

@RequestBody 🡺 HttpRequest data 🡺Json format to Java Object convert

@RestController 🡺 @Controller +@ResponseBody

@EnableAutoConfiguration 🡺

@ComponentScan

@SpringBootConfiguration 🡺

@SpringBootApplication 🡺 @Enable + Compon + SpringBootConfig

@Service

@Repository

@Transactional

@Autowired

@Qualifier

@Bean

@Component

@PostMapping 🡺 Save

@GetMapping 🡺 Fetch operations

@PutMapping 🡺 update

@DeleteMapping 🡺 delete

@RequestParam 🡺 ? key= value

@PathVariable 🡺 /headset

@ControllerAdvice

@ExceptionalHandler

@RestControllerAdvice

@Configuration

{

}

Student 🡺 Address 🡺 College 🡺 Course

Employee 🡺 Address 🡺 company 🡺 Projects

Diagram

Description automatically generated