30 November 2021 21:40

In Last session we discussed below annotations @Component

@Service

@Repository

@Controller @RestController

@Configuration

@Bean (Method Level Annotation)

Auto Wiring

-> As part of application we will develop several classes

-> One class method needs to talk to another class method to execute functionality

-> In order to access one class method in another class method we need to perform

dependency injection

-> In Spring Framework IOC container will perform dependency injection

-> To give instructions to IOC to perform Dependency Injection we will use Auto-Wiring

=> Auto Wiring is used to inform IOC to perform Dependency Injection

=> To enable Auto Wiring we will use @AutoWired annotation

=> @AutoWired annotation we can use at below 3 places

1. Setter method level (Setter Injection)
2. Constructor Level (Constructor Injection)
3. Field Level (Field Injection)

============================================================================

@Repository

public class UserDao {

public boolean save() {

// logic to insert user into DB

return true;

}

}

=============================================================================

@Service

public class UserService {

@Autowired

private UserDao userDao;

/\*@Autowired

public void setUserDao(UserDao userDao) {

this.userDao = userDao;

}\*/

/\*@Autowired

public UserService(UserDao userDao) {

this.userDao = userDao;

}\*/

public void saveUser() {

boolean status = userDao.save(); if (status) {

System.out.println("Record Inserted");

} else {

System.out.println("Record Not Inserted");

}

}

}

==========================================================================

@SpringBootApplication

public class Application {

public static void main(String[] args) {

ConfigurableApplicationContext context = SpringApplication.run(Application.class, args);

UserService userService = context.getBean(UserService.class); userService.saveUser();

}

}

=============================================================================