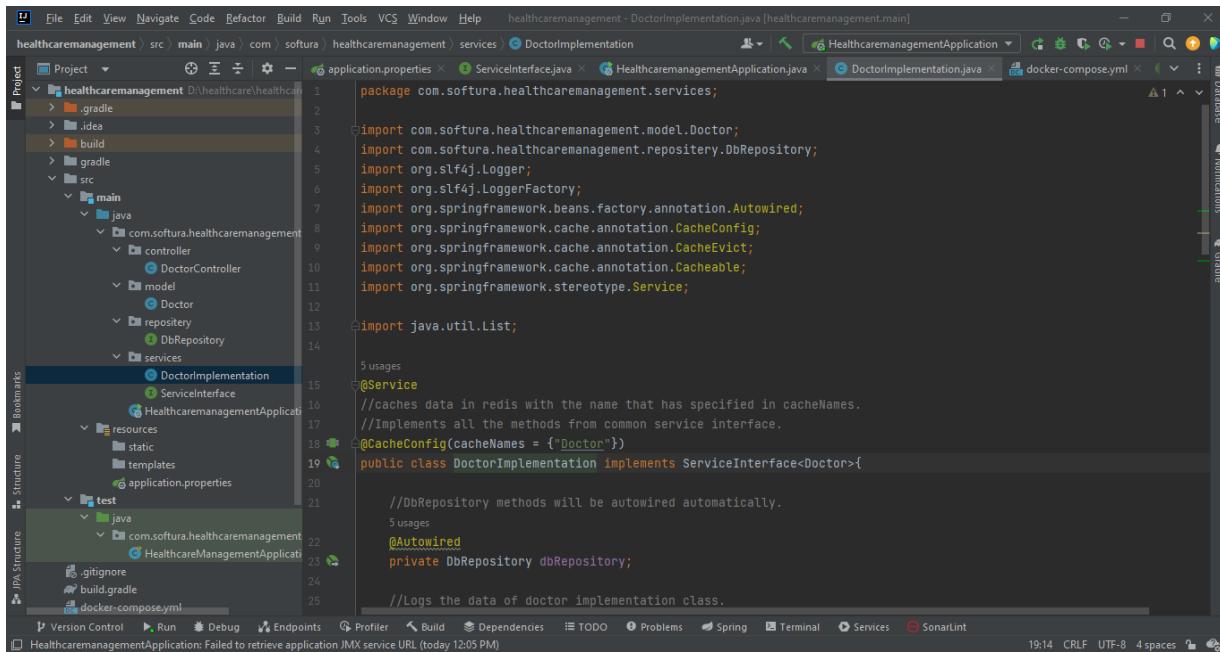


Running locally: -

Code Screen Shorts: -



IntelliJ IDEA interface showing the `DoctorImplementation.java` file. The code implements the `ServiceInterface` for the `Doctor` service. It uses `DbRepository` for database operations and `Redis` for caching. The code includes annotations like `@Service`, `@CacheConfig`, and `@CacheEvict`.

```
package com.softura.healthcaremanagement.services;

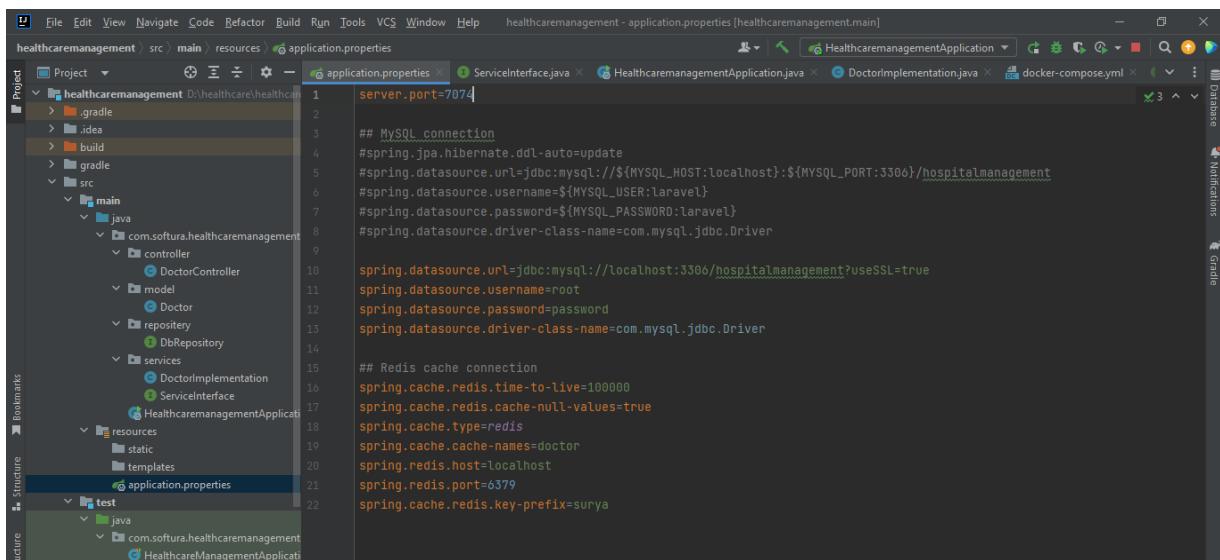
import com.softura.healthcaremanagement.model.Doctor;
import com.softura.healthcaremanagement.repository.DbRepository;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.cache.annotation.CacheConfig;
import org.springframework.cache.annotation.CacheEvict;
import org.springframework.cache.annotation.Cacheable;
import org.springframework.stereotype.Service;

import java.util.List;

5 usages
@Service
//caches data in redis with the name that has specified in cacheNames.
//Implements all the methods from common service interface.
@CacheConfig(cacheNames = {"Doctor"})
public class DoctorImplementation implements ServiceInterface<Doctor>{

    //DbRepository methods will be autowired automatically.
    5 usages
    @Autowired
    private DbRepository dbRepository;

    //Logs the data of doctor implementation class.
}
```



IntelliJ IDEA interface showing the `application.properties` configuration file. It defines the application port, MySQL connection details, and Redis cache settings.

```
server.port=7071

## MySQL connection
spring.jpa.hibernate.ddl-auto-update
spring.datasource.url=jdbc:mysql://${MYSQL_HOST}: ${MYSQL_PORT}/hospitalmanagement
spring.datasource.username=${MYSQL_USER:laravel}
spring.datasource.password=${MYSQL_PASSWORD:laravel}
spring.datasource.driver-class-name=com.mysql.jdbc.Driver

spring.datasource.url=jdbc:mysql://localhost:3306/hospitalmanagement?useSSL=true
spring.datasource.username=root
spring.datasource.password=password
spring.datasource.driver-class-name=com.mysql.jdbc.Driver

## Redis cache connection
spring.cache.redis.time-to-live=100000
spring.cache.redis.cache-null-values=true
spring.cache.type=redis
spring.cache.cache-names=doctor
spring.redis.host=localhost
spring.redis.port=6379
spring.cache.redis.key-prefix=surya
```

Outputs: -

The screenshot shows the Postman Scratch Pad interface. On the left sidebar, there are tabs for 'Collections', 'APIs', 'Environments', 'Mock Servers', 'Monitors', and 'History'. Below the sidebar, there's a section titled 'You don't have any collections' with a sub-section 'Collections let you group related requests, making them easier to access and run.' and a 'Create Collection' button.

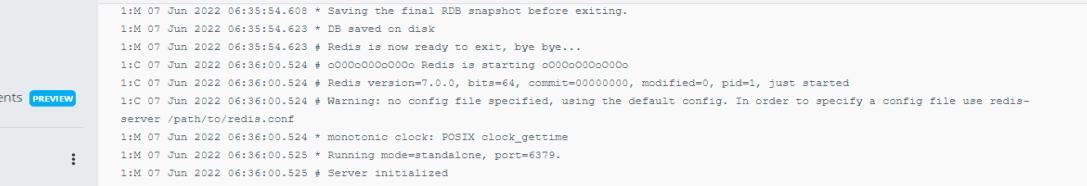
The main workspace has a title bar 'Working locally in Scratch Pad. Switch to a Workspace' with a 'New' and 'Import' button. Below the title bar is a toolbar with various API methods: GET, POST, PUT, DELETE, etc. The URL in the address bar is 'http://localhost:7074/doctors/getByid/v1.0/1003'. To the right of the URL are 'Save' and 'Send' buttons.

The request configuration panel shows a 'GET' method selected. Below it are tabs for 'Params', 'Authorization', 'Headers (6)', 'Body', 'Pre-request Script', 'Tests', and 'Settings'. The 'Params' tab is active, showing a single entry 'Key' with 'value' as the value. The 'Cookies' tab is also visible.

The 'Body' tab is selected in the preview panel, which displays the JSON response:

```
1  {
2   "id": 1003,
3   "name": "Lakshmi",
4   "mobileNumber": 8855001144,
5   "status": "Available",
6   "startTime": "12:00:00",
7   "endTime": "08:30:00"
8 }
```

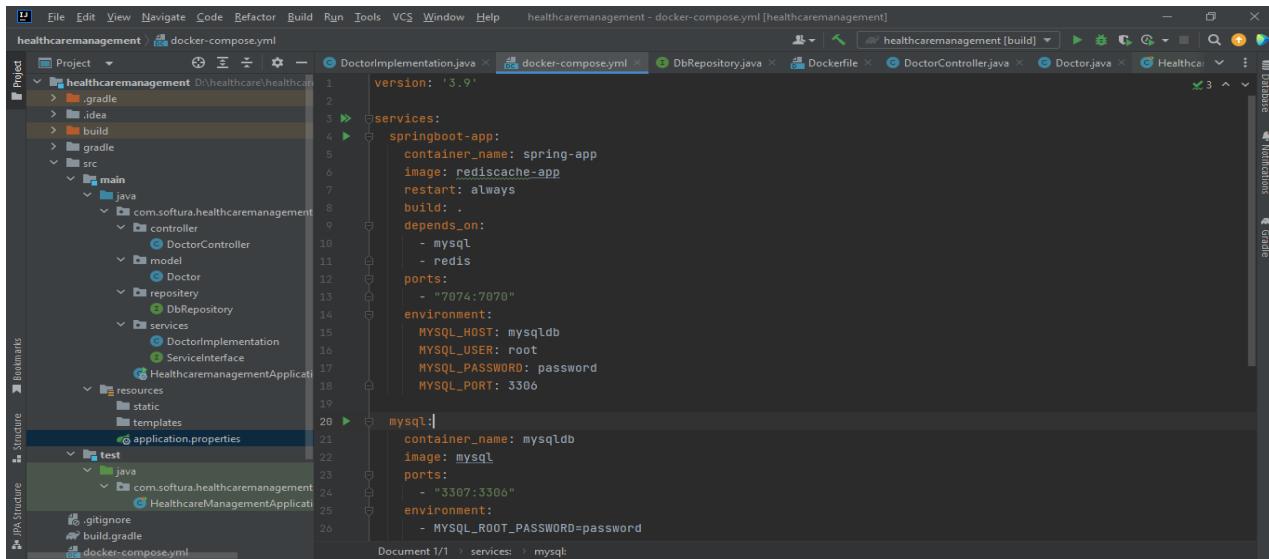
The response details at the bottom show 'Status: 200 OK' with a green checkmark, 'Time: 23 ms', and 'Size: 282 B'. There are 'Save Response' and 'Copy' buttons to the right.



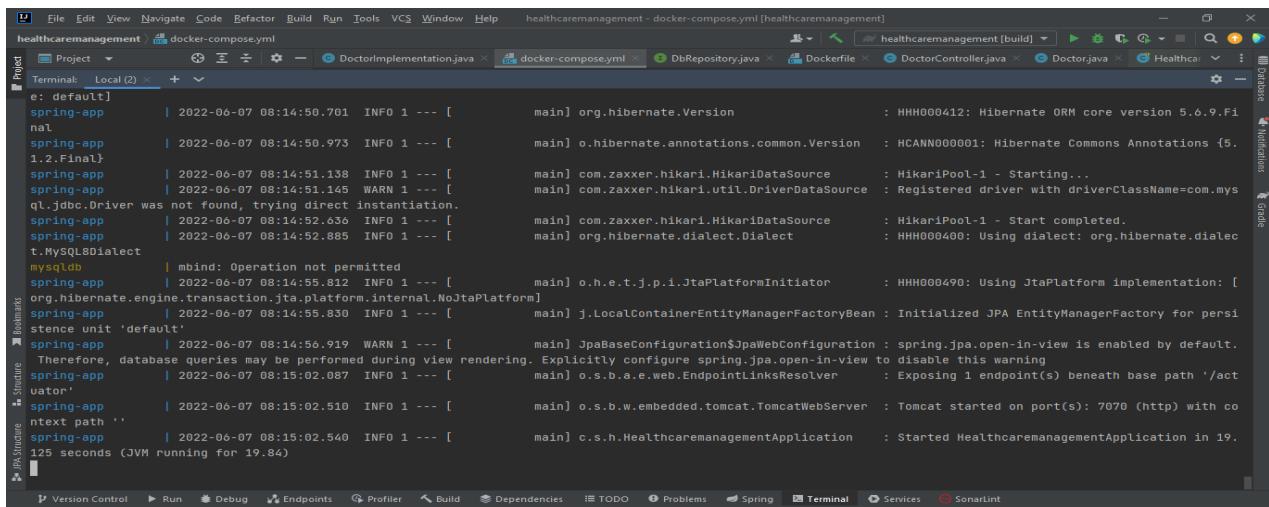
The screenshot shows the logs of a RediSQL Docker container named "redis". The logs output the following text:

```
1:M 07 Jun 2022 06:35:54.608 * Saving the final RDB snapshot before exiting.
1:M 07 Jun 2022 06:35:54.623 * DB saved on disk
1:M 07 Jun 2022 06:35:54.623 # Redis is now ready to exit, bye bye...
1:C 07 Jun 2022 06:36:00.524 # o000o000o000 Redis is starting o000o000o000
1:C 07 Jun 2022 06:36:00.524 # Redis version=7.0.0, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 07 Jun 2022 06:36:00.524 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
1:M 07 Jun 2022 06:36:00.524 * monotonic clock: POSIX clock_gettime
1:M 07 Jun 2022 06:36:00.525 # Running mode=standalone, port=6379,
1:M 07 Jun 2022 06:36:00.525 # Server initialized
1:M 07 Jun 2022 06:36:00.525 # WARNING overcommit_memory is set to 0! Background save may fail under low memory condition. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effect.
1:M 07 Jun 2022 06:36:00.525 * The AOF directory appendonlydir doesn't exist
1:M 07 Jun 2022 06:36:00.525 * Loading RDB produced by version 7.0.0
1:M 07 Jun 2022 06:36:00.525 * RDB age 6 seconds
1:M 07 Jun 2022 06:36:00.525 * RDB memory usage when created 0.90 Mb
1:M 07 Jun 2022 06:36:00.525 * Done loading RDB, keys loaded: 1, keys expired: 0.
1:M 07 Jun 2022 06:36:00.525 * DB loaded from disk: 0.000 seconds
1:M 07 Jun 2022 06:36:00.525 * Ready to accept connections
1:M 07 Jun 2022 07:36:01.041 * 1 changes in 3600 seconds. Saving...
1:M 07 Jun 2022 07:36:01.057 * Background saving started by pid 28
28:C 07 Jun 2022 07:36:01.092 * DB saved on disk
28:C 07 Jun 2022 07:36:01.094 * Fork COW for RDB: current 0 MB, peak 0 MB, average 0 MB
1:M 07 Jun 2022 07:36:01.159 * Background saving terminated with success
```

Docker-compose code: -



```
healthcaremanagement / docker-compose.yml [healthcaremanagement]
Project File Edit View Navigate Code Refactor Build Run Tools VCS Window Help healthcaremanagement - docker-compose.yml [healthcaremanagement]
healthcaremanagement > .gradle > .idea > build > gradle > src > main > java > com.softura.healthcaremanagement > controller > DoctorController > model > Doctor > repository > DbRepository > services > Doctorimplementation > Serviceinterface > HealthcaremanagementApplication > resources > static > templates > application.properties
1 version: '3.9'
2
3 services:
4   springboot-app:
5     container_name: spring-app
6     image: rediscache-app
7     restart: always
8     build: .
9     depends_on:
10       - mysql
11       - redis
12     ports:
13       - "7074:7070"
14     environment:
15       MYSQL_HOST: mysqldb
16       MYSQL_USER: root
17       MYSQL_PASSWORD: password
18       MYSQL_PORT: 3306
19
20   mysql:
21     container_name: mysqldb
22     image: mysql
23     ports:
24       - "3307:3306"
25     environment:
26       MYSQL_ROOT_PASSWORD=password
Document 1/1 > services: > mysql
```



```
healthcaremanagement / docker-compose.yml [healthcaremanagement]
Project File Edit View Navigate Code Refactor Build Run Tools VCS Window Help healthcaremanagement - docker-compose.yml [healthcaremanagement]
Terminal Local (2) >
e: default]
spring-app | 2022-06-07 08:14:50.701 INFO 1 --- [           main] org.hibernate.Version                 : HHH000412: Hibernate ORM core version 5.6.9.Final
na1
spring-app | 2022-06-07 08:14:50.973 INFO 1 --- [           main] o.hibernate.annotations.common.Version  : HCANN000001: Hibernate Commons Annotations 5.1.2.Final
spring-app | 2022-06-07 08:14:51.138 INFO 1 --- [           main] com.zaxxer.hikari.HikariDataSource    : HikariPool-1 - Starting...
spring-app | 2022-06-07 08:14:51.145 WARN 1 --- [           main] com.zaxxer.hikari.util.DriverDataSource : Registered driver with driverClassName=com.mysql.jdbc.Driver was not found, trying direct instantiation.
spring-app | 2022-06-07 08:14:52.636 INFO 1 --- [           main] com.zaxxer.hikari.HikariDataSource    : HikariPool-1 - Start completed.
spring-app | 2022-06-07 08:14:52.885 INFO 1 --- [           main] org.hibernate.dialect.Dialect        : HHH000400: Using dialect: org.hibernate.dialect.MySQL8Dialect
mysqldb | mbind: Operation not permitted
spring-app | 2022-06-07 08:14:55.812 INFO 1 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator   : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
spring-app | 2022-06-07 08:14:55.830 INFO 1 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
spring-app | 2022-06-07 08:14:56.919 WARN 1 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed during view rendering. Explicitly configure spring.jpa.open-in-view to disable this warning
spring-app | 2022-06-07 08:15:02.087 INFO 1 --- [           main] o.s.b.a.e.web.EndpointLinksResolver   : Exposing 1 endpoint(s) beneath base path '/actuator'
spring-app | 2022-06-07 08:15:02.510 INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 7070 (http) with context path ''
spring-app | 2022-06-07 08:15:02.540 INFO 1 --- [           main] c.s.h.HealthcaremanagementApplication : Started HealthcaremanagementApplication in 19.125 seconds (JVM running for 19.84)
```


Test case: -

The screenshot shows an IDE interface with the following details:

- Project Structure:** The project is named "HealthcareManagementApplication". It contains a "resources" folder with static and templates subfolders, and an "application.properties" file. The "test" folder contains a "java" folder with a package "com.softura.healthcaremanagement" containing the "HealthcareManagementApplicationTests" class.
- Code Editor:** The "HealthcareManagementApplicationTests.java" file is open. The code includes annotations like @InjectMocks, @Mock, and @BeforeEach, and a test method "void contextLoads()".
- Run Tab:** The "HealthcaremanagementApplication" run configuration is selected. The output shows a successful test run:

```
Shutdown initiated...
2022-06-07 14:19:23.519  INFO 22076 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource      : HikariPool-1 -
Shutdown completed.
BUILD SUCCESSFUL in 27s
4 actionable tasks: 2 executed, 2 up-to-date
2:19:23 PM: Execution finished 'test --tests "com.softura.healthcaremanagement.HealthcareManagementApplicationTests"'.
```
- Test Results:** A summary shows 1 test passed in 0.110s, with a 100% success rate.

