


```
spring.redis.host=localhost
spring.redis.port=6379
spring.redis.password= # Redis
spring.redis.database=0 #
```

وفقًا للمعتمدة والمعلومات المضيفة بتحديث قم ، مثل خدمة أو بُعد عن خادم تستخدم كنت إذا
لذلك.

3. مع الأساسي الاستخدام RedisTemplate

مكون: أو خدمة في حقن هذا يمكنك الاستوى. المنخفضة للعمليات RedisTemplate يوفر

```
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.data.redis.core.RedisTemplate;
import org.springframework.stereotype.Service;

@Service
public class RedisService {
    private final RedisTemplate<String, String> redisTemplate;

    @Autowired
    public RedisService(RedisTemplate<String, String> redisTemplate) {
        this.redisTemplate = redisTemplate;
    }

    public void saveData(String key, String value) {
        redisTemplate.opsForValue().set(key, value);
    }

    public String getData(String key) {
        return redisTemplate.opsForValue().get(key);
    }
}
```

- مثل أخرى أنواع استخدام يمكنك نصي. نوع من هي والقيم المفاتيح أن يعني <String, String> عام: هو RedisTemplate <String, Object>.
- opsForValue() هو المفاتيح البسيطة للعمليات هو opsForList(), opsForSet(), opsForHash(), بيانات هي الكل مختم لف مع للعمل إلخ،.

4. الكائنات مع الاسخدام

ولكن تلقائيًا، بتكوينه يقوم `RedisTemplate` بحولين. مع `RedisTemplate` بتكوين قم `RedisTemplate`، كائنات واسترجاع لحفظ ضروريًا: ذلك لأن إذا تخطي صه يمكنك

```
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.data.redis.connection.RedisConnectionFactory;
import org.springframework.data.redis.core.RedisTemplate;
import org.springframework.data.redis.serializer.Jackson2JsonRedisSerializer;
import org.springframework.data.redis.serializer.StringRedisSerializer;

@Configuration
public class RedisConfig {

    @Bean
    public RedisTemplate<String, Object> redisTemplate(RedisConnectionFactory connectionFactory) {
        RedisTemplate<String, Object> template = new RedisTemplate<>();
        template.setConnectionFactory(connectionFactory);
        template.setKeySerializer(new StringRedisSerializer());
        template.setValueSerializer(new Jackson2JsonRedisSerializer<>(Object.class));
        template.afterPropertiesSet();
        return template;
    }
}
```

الكائنات: واسترجاع حفظ يمكنك الآن

```
public class Person {
    private String firstName;
    private String lastName;

    // ( )
    public Person() {}

    public Person(String firstName, String lastName) {
        this.firstName = firstName;
        this.lastName = lastName;
    }

    //
    public String getFirstName() { return firstName; }
```

```

    public void setFirstName(String firstName) { this.firstName = firstName; }
    public String getLastName() { return lastName; }
    public void setLastName(String lastName) { this.lastName = lastName; }
}

```

@Service

```

public class PersonRedisService {
    private final RedisTemplate<String, Object> redisTemplate;

    @Autowired
    public PersonRedisService(RedisTemplate<String, Object> redisTemplate) {
        this.redisTemplate = redisTemplate;
    }

    public void savePerson(String key, Person person) {
        redisTemplate.opsForValue().set(key, person);
    }

    public Person getPerson(String key) {
        return (Person) redisTemplate.opsForValue().get(key);
    }
}

```

5. الـمستودعات مع الـاستخدام

ومستودعًا؛ لكيانًا حدد. `RedisTemplate<String, Object>` مستودعات استخدام الـتعبئة، من أعلى مستوى لاستخدام

```

import org.springframework.data.annotation.Id;
import org.springframework.data.redis.core.RedisHash;

@RedisHash("Person") // Redis "Person"
public class Person {
    @Id
    private String id; // Redis "Person:<id>"
    private String firstName;
    private String lastName;

    // (
}

```

```

import org.springframework.data.repository.CrudRepository;

```

```
public interface PersonRepository extends CrudRepository<Person, String> {
}
```

هكذا: اسخدمه

```
@Service
public class PersonService {
    private final PersonRepository repository;

    @Autowired
    public PersonService(PersonRepository repository) {
        this.repository = repository;
    }

    public void savePerson() {
        Person person = new Person("John", "Doe");
        repository.save(person);
        System.out.println( "      ID: " + person.getId());
    }

    public void findPerson(String id) {
        Person person = repository.findById(id).orElse(null);
        if (person != null) {
            System.out.println(person.getFirstName() + " " + person.getLastName());
        }
    }
}
```

[] @RedisHash هاش إلى الـ كـي ان يربط بـ RedisHash.
 [] تهيئة. أي دون من [] عمليات يوفر الـ مسـتودع

6. تطبيـق كـ تشغيـل

ثم بُعد. عن مـثـيـل إلى اتـصـل أو []: docker run -d -p 6379:6379 redis
 بـك: الـ خاص [] [] تطبيـق ابـدأ

```
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
```

```

public class Application {
    public static void main(String[] args) {
        SpringApplication.run(Application.class, args);
    }
}

```

7. الـمقدمة الـمميزات

- redisTemplate.expire(key, 10, TimeUnit.MINUTES).
- مـباشـرة. الـهـاشـات مـع لـلـعـمـل opsForHash() اسـتـخـدم الـهـاشـات:
- لـلـرسـائـل. RedisMessageListenerContainer اسـتـخـدم الـنـشـر الـاسـتـمـاع:

الـعـمـل سـيـر عـلـى مـثـال

1. redisTemplate.opsForValue().set() أو repository.save() مـع Person لـكـائـن اـحـفـظ.
2. redisTemplate.opsForValue().get() أو repository.findById() مـع اسـتـرجـعـهـا.
3. الـبـيـانـات. لـرؤـيـة redis-cli مـع مـثـلاً بـك الـخـاص □□□□□ مـثـيـل مـن تـحـقـق.

تـحـاول الـذي مـا أـخـبـرنـي مـحـدد، شـيـء أو الـنـشـر الـاسـتـمـاع، الـمـؤـقـت، الـتـخـزـيـن اسـتـكـشـاف تـريـد كـنـت إذا الـأسـاسـيـات! يـغـطـي هـذا تـحـقـيـقـه؟ تـحـاول مـا مـع هـذا يـتـوافـق كـيـف أكـثـر. وسـأـغـوص تـحـقـيـقـه،