

Spring Boot + GCP Redis (Memorystore) Deployment Guide

Prerequisites

- Google Cloud project (PROJECT_ID)
- Billing enabled
- Cloud Shell or gcloud CLI
- Docker installed (for local builds)
- Java 17+ and Spring Boot 3.2+

Step 1: Create Redis Instance on GCP

```
gcloud config set project [PROJECT_ID]
gcloud services enable redis.googleapis.com
gcloud redis instances create my-redis \
  --size=1 \
  --region=us-central1 \
  --tier=basic \
  --zone=us-central1-a

# Get Redis host/port
gcloud redis instances describe my-redis --region=us-central1
```

Step 2: Create Spring Boot App

Dependencies (build.gradle.kts):

```
dependencies {
    implementation("org.springframework.boot:spring-boot-starter-web")
    implementation("org.springframework.boot:spring-boot-starter-data-redis")
    implementation("org.springframework.boot:spring-boot-starter-actuator")
}

application.properties:
spring.redis.host=${REDIS_HOST}
```

Spring Boot + GCP Redis (Memorystore) Deployment Guide

```
spring.redis.port=${REDIS_PORT:6379}
```

```
@SpringBootApplication
```

```
@EnableCaching
```

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

```
@RestController
```

```
@RequestMapping("/api")
```

```
public class GreetingController {  
    @Cacheable("greetings")  
    @GetMapping("/greet/{name}")  
    public String greet(@PathVariable String name) {  
        System.out.println("Processing request...");  
        return "Hello, " + name + "! at " + LocalDateTime.now();  
    }  
}
```

Step 3: Dockerize Spring Boot App

Dockerfile:

```
FROM gcr.io/distroless/java17  
COPY build/libs/*.jar app.jar  
ENTRYPOINT ["java", "-jar", "app.jar"]
```

Step 4: Deploy to Cloud Run

Build and push Docker image

Spring Boot + GCP Redis (Memorystore) Deployment Guide

```
gcloud builds submit --tag gcr.io/[PROJECT_ID]/springboot-redis
```

Deploy to Cloud Run

```
gcloud run deploy springboot-redis-app \  
  --image gcr.io/[PROJECT_ID]/springboot-redis \  
  --platform managed \  
  --region us-central1 \  
  --allow-unauthenticated \  
  --set-env-vars REDIS_HOST=[REDIS_HOST_FROM_STEP_1]
```

Testing

```
curl https://[CLOUD_RUN_URL]/api/greet/sree
```

First request logs "Processing request..."

Subsequent requests use Redis cache

Optional: Local Redis for Dev

```
docker run -d -p 6379:6379 redis:7
```

spring.redis.host=localhost (for local testing)

Clean Up

```
gcloud redis instances delete my-redis --region=us-central1
```

```
gcloud run services delete springboot-redis-app --region=us-central1
```