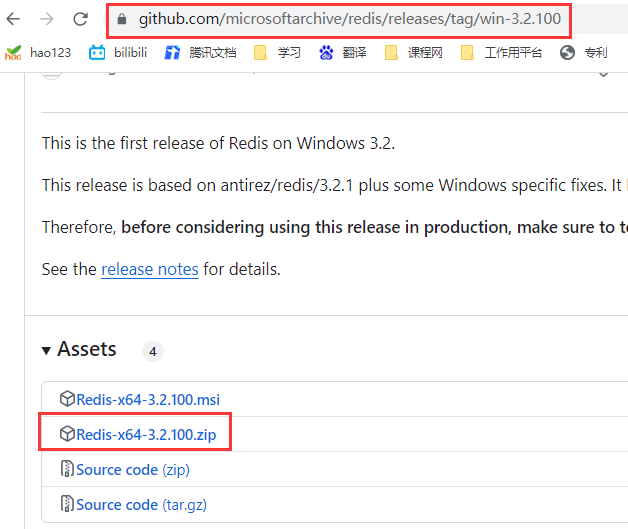
# 实验八 Spring Boot整合Redis

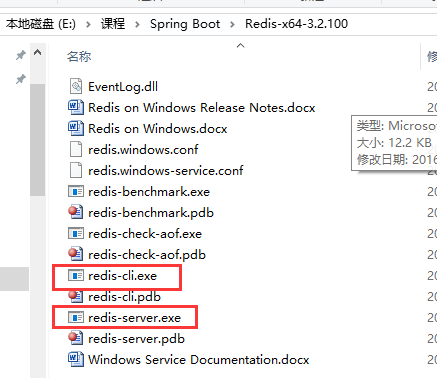
一、Redis下载及安装

1.登录网址：<https://github.com/microsoftarchive/redis/releases/tag/win-3.2.100>

2.下载[Redis-x64-3.2.100.zip](https://github.com/microsoftarchive/redis/releases/download/win-3.2.100/Redis-x64-3.2.100.zip)



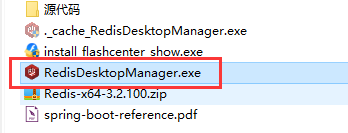
3.解压[Redis-x64-3.2.100.zip](https://github.com/microsoftarchive/redis/releases/download/win-3.2.100/Redis-x64-3.2.100.zip)压缩包



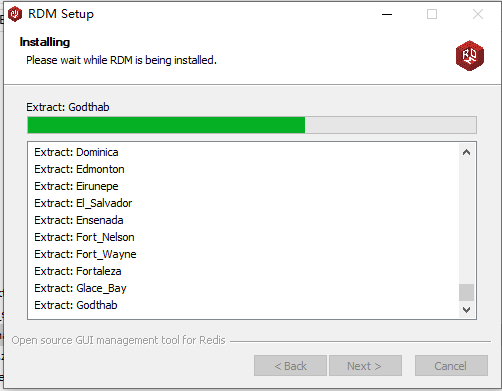
4.双击启动redis-server.exe

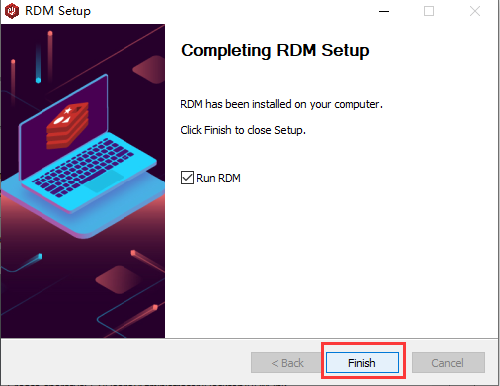


5.下载redisdesktopManager管理器，



6.安装redisdesktopManager

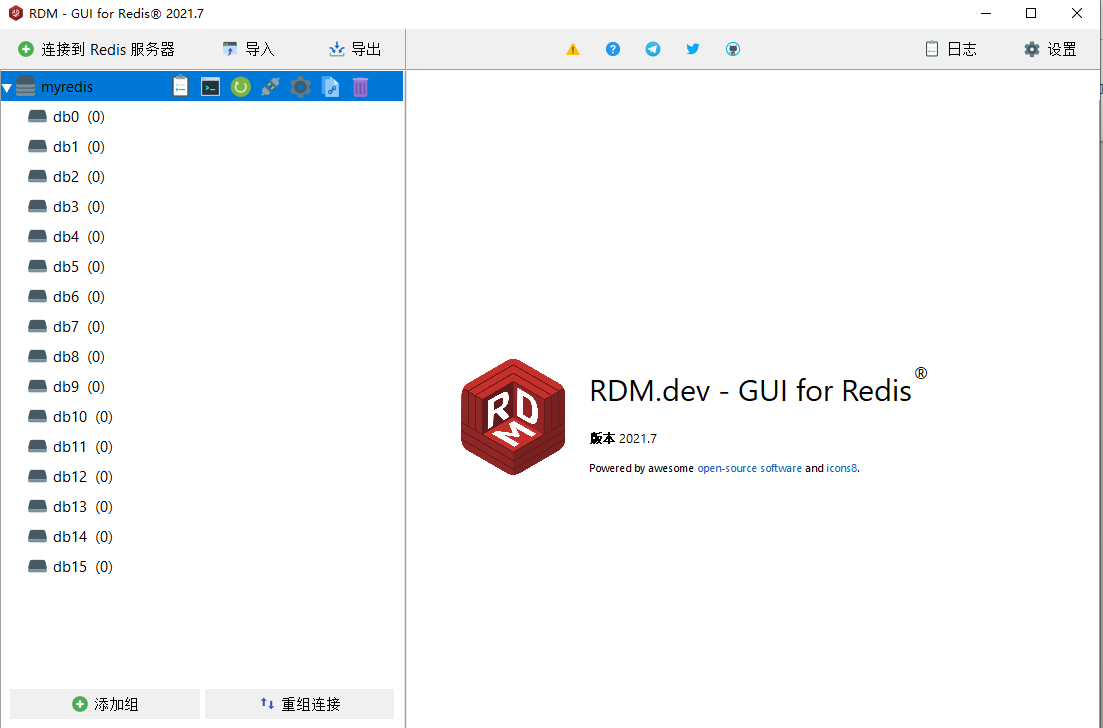




7.连接到Redis服务器

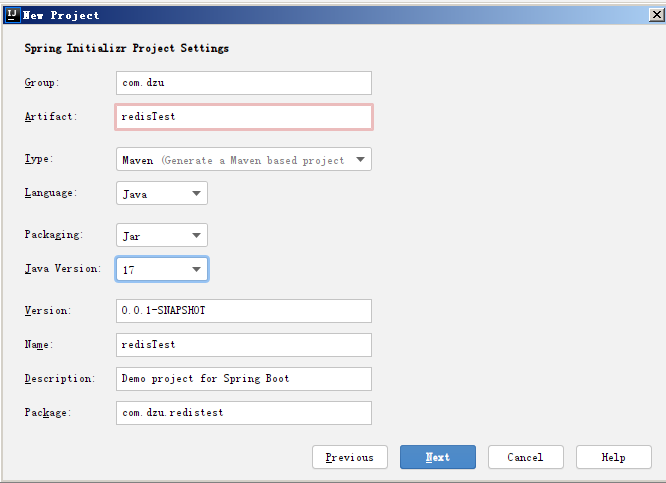






二、SpringBoot整合Redis

1.使用Spring Initializr方式新建项目

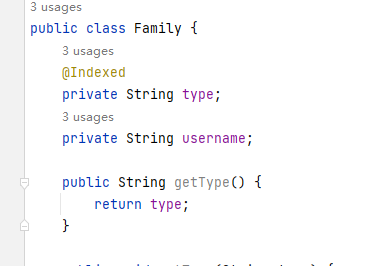


2.在pom.xml中修改Java版本、父工程版本，设置Maven配置文件和本地仓库文件, 添加redis依赖



3.创建实体类,添加注解,自动生成构造方法(有参、无参)，getter/setter方法，toString方法



4.创建Repository接口

public interface PersonRepository extends CrudRepository<Person,String> {

List<Person> findByLastname(String lastname);  
Page<Person> findPersonByLastname(String lastname, Pageable pageable);  
List<Person> findByFirstnameAndLastname(String firstname,String lastname);  
List<Person> findByAddress\_City(String city);  
List<Person> findByFamilyList\_Username(String username);

}

5.修改application.properties，进行Redis数据库连接配置

spring.redis.host=127.0.0.1  
spring.redis.port=6379  
spring.redis.password=

7.测试

@SpringBootTest  
class RedisTestApplicationTests {  
 @Autowired  
 private PersonRepository repository;  
 @Test  
 public void savePerson(){  
 Person person = new Person("有才","张");  
 Person person1 = new Person("小妹","王");  
 Address address = new Address("北京","中国");  
 person.setAddress(address);  
 List<Family> families = new ArrayList<>();  
 Family family1 = new Family("父亲","张君");  
 Family family2 = new Family("母亲","马兰花");  
 families.add(family1);  
 families.add(family2);  
 person.setFamilyList(families);  
 Person save1 = repository.save(person);  
 Person save2 = repository.save(person1);  
 System.*out*.println(save1);  
 System.*out*.println(save2);  
 }

@Test  
public void selectPerson() {  
 List<Person> list = repository.findByAddress\_City("北京");  
 System.*out*.println(list);  
}  
@Test  
public void updatePerson() {  
 Person person = repository.findByFirstnameAndLastname("张","有才").get(0);  
 person.setLastname("小明");  
 Person update = repository.save(person);  
 System.*out*.println(update);  
}  
@Test  
public void deletePerson() {  
 Person person = repository.findByFirstnameAndLastname("张","小明").get(0);  
 repository.delete(person);  
}

}