



链滴

SpringBoot 中使用 GridFS

作者: [HuanYuanHe](#)

原文链接: <https://ld246.com/article/1576978252549>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

什么是GridFS

GridFS是MongoDB提供的用于持久化存储文件的模块

在GridFS存储文件是将文件分块存储，文件会按照256KB的大小分割成多个块进行存储，GridFS使用个集合 (collection)存储文件，一个集合是chunks, 用于存储文件的二进制数据;一个集合是files, 用存储文件的元数据信息(文件名称、块大小、上传时间等信息)。

从GridFS中读取文件要对文件的各各块进行组装、合并。详细参考:<https://docs.mongodb.com/manual/core/gridfs/>

在SpringBoot中使用GridFS

存储文件

```
@Autowired
GridFsTemplate gridFsTemplate;

@Test
public void GridFsTest() throws FileNotFoundException {
    //选择要存储的文件
    File file = new File("/Users/xxx/Desktop/xxx.docx");
    InputStream inputStream = new FileInputStream(file);
    //存储文件并起名称
    ObjectId objectId = gridFsTemplate.store(inputStream, "面试宝典");
    String id = objectId.toString();
    //获取到文件的id, 可以从数据库中查找
    System.out.println(id);
}
```

查找文件

创建GridFSBucket对象

```
@Configuration
public class MongoConfig {

    @Value("${spring.data.mongodb.database}")
    String db;

    @Bean
    public GridFSBucket getGridFSBucket(MongoClient mongoClient){
        MongoDBDatabase mongoDatabase = mongoClient.getDatabase(db);
        GridFSBucket bucket = GridFSBuckets.create(mongoDatabase);
        return bucket;
    }
}
```

```
@Autowired
GridFsTemplate gridFsTemplate;
```

```
@Autowired
GridFSBucket gridFSBucket;
```

```
@Test
public void queryFile() throws IOException {
    String id = "5c1b8fac72884e389ae3df82";
    //根据id查找文件
    GridFSFile gridFSFile = gridFsTemplate.findOne(new Query(Criteria.where("_id").is(id)));
    //打开下载流对象
    GridFSDownloadStream gridFS = gridFSBucket.openDownloadStream(gridFSFile.getObjectI
());
    //创建gridFsSource, 用于获取流对象
    GridFsResource gridFsResource = new GridFsResource(gridFSFile,gridFS);
    //获取流中的数据
    String string = IOUtils.toString(gridFsResource.getInputStream(), "UTF-8");
    System.out.println(string);
}
```

删除文件

```
//删除文件
@Test
public void testDelFile() throws IOException {
    //根据文件id删除fs.files和fs.chunks中的记录
    gridFsTemplate.delete(Query.query(Criteria.where("_id").is("5c1b8fac72884e389ae3df82"))
);
}
```