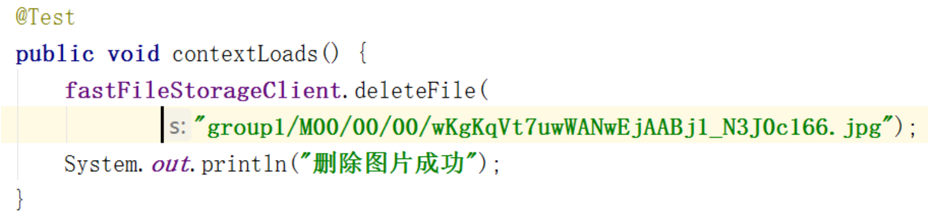
11_2_SpringBoot整合FastDFS

- 在我们的后台系统整合FastDFS,具体操作步骤如下：
- 1，添加依赖
  -
    ```
    <dependency>
                            <groupId>com.github.tobato</groupId>
                            <artifactId>fastdfs-client</artifactId>
                            <version>1.26.1-RELEASE</version>
                    </dependency>
    ```
- 2，导入配置
  -
    ```
    @Import(FdfsClientConfig.class)
    @SpringBootApplication
    public class TestApplication {
            ...
    }
    ```
- 3，配置application.yml
  -
    ```
    fdfs:
      so-timeout: 1500
      connect-timeout: 600
      pool:
        jmx-enabled: false
      thumb-image:
        width: 100
        height: 100
      tracker-list: 192.168.10.168:22122
    ```
- 4，使用客户端操作类
  -
    ```
    @Autowired
        private FastFileStorageClient fastFileStorageClient;
     @RequestMapping("/upload")
        @ResponseBody
        public String upload(){
            File file = new File("");
            try {
                StorePath storePath =
    fastFileStorageClient.uploadImageAndCrtThumbImage(new FileInputStream(file),
    file.length(), "JPG", null);
                String fullPath = storePath.getFullPath();
                String path = storePath.getPath();
            } catch (FileNotFoundException e) {
                e.printStackTrace();
            }
            return "OK";
        }
    ```
    -
- 5，设置上传的资源大小限制

    -

      ```
      spring:
        servlet:
          multipart:
            max-file-size: 10MB
      ```

- 6，资源的维护（重要）
    - 需要维护的
    - 当我们修改图片，或者删除记录这样的操作时，这个图片就再也没有被使用的机会
    - 所以这个时候我们除了维护数据库本身的记录，还需要去删除相关的图片资源
    - 怎么删除？路径（groupName+path）
    -

      ```
      @Test
      public void contextLoads() {
          fastFileStorageClient.deleteFile(
                  s: "group1/M00/00/00/wKgKqVt7uwWANwEjAABj1_N3J0c166.jpg");
          System.out.println("删除图片成功");
      }
      ```

-