1. S3에 파일 업로드를 하기위해 AWSServiceImpl에서 upload메소드를 작성하였다.
   1. fileName을 multipartFile에서 받고, data도 getInputStream메소드로 받는다.
   2. try catch문으로 에러를 처리했다.
   3. s3의 putObject메소드를 사용하기위해
      1. bucketname : 고정으로 사용
      2. fileName : a에서 얻음
      3. data : a에서 얻음
      4. metadata : 데이터의 크기, multipartFile에서 .getSize()메소드 사용
   4. 하여 putObject메소드로 S3에 파일을 업로드 했다.
2. S3에 있는 버켓안의 객체 리스트를 가저오기 위해 s3.jsp에서 form으로 버켓이름을 받아 controller에게 전달했다.
   1. iAWSServ에 getFileList라는 메소드를 새로 만들고, 반환값은 List<S3ObjectSummary>로 만든다.
   2. s3를 받아오고,
   3. ListObjectsV2Result result = s3.listObjectsV2(bucketName);
   4. return result.getObjectSummaries();
3. 파일 읽어오기 테스트 진행
   1. iAWSServiceImpl에서 getImageURL(bucketName, fileName) 메소드를 만들어준다. (bucketName과 fileName은 빠른 진행을 위해 고정한다.)
   2. 반환형은 URL 로 만들어준다.
   3. s3를 얻고 s3.getUrl(bucketName, fileName)을 리턴해준다.
   4. a 태그와 image태그를 사용하여 보여주지만, 접근 권한이 설정되어 있지 않기 때문에 접근할 수 없다.
4. 객체 액세스 제어 목록 읽기
   1. form태그로 text형식으로 버킷 이름과 파일 이름을 controller로 보내준다.
   2. @RequestParam("bucketName") String bucketName
   3. b의 형식으로 버켓네임과 파일네임을 받아준다.
   4. model.addAttribute로 s3.jsp로 값을 보내주면서
   5. iAWSServ.getAccessCtrlLst(bucketName, fileName)메소드를 만들어준다.
   6. s3를 얻고,
   7. AccessControlList acl = s3.getObjectAcl(bucketName, fileName);
   8. 현재 bucketName에 있는 fileName의 AccessContolList정보를 얻는다.
   9. return acl.getGrantsAsList()로 리턴해준다.
   10. <ul>
   11. <c:forEach var="grant" items="${grants }">
   12. <li>Grantee : ${grant.getGrantee() }</li>
   13. <li>Grantee : ${grant.getGrantee().getIdentifier() }</li>
   14. <li>Permission : ${grant.getPermission() }</li>
   15. </c:forEach>
   16. </ul>
   17. Grants리스트는 위 형식으로 한정보씩 읽어 출력한다.
5. 객체 액세스 제어 목록 설정
   1. 목록을 읽어올때처럼 bucketName과 fileName을 controller로 보내준다.
   2. iAWSServ.setAccess(bucketName, fileName) 메소드를 작성해준다.
   3. s3를 얻어온후 목록을 얻을때와 마찬가지로

AccessControlList acl = s3.getObjectAcl(bucketName, fileName);로 AccessControlList를 불러와준다.

* 1. acl.grantPermission(GroupGrantee.AllUsers, Permission.Read);
  2. 위와같이 액세스제어 설정을 저장해준다.
  3. s3.setObjectAcl(bucketName, fileName, acl)로 객체에 대한 액세스 제어설정을 세팅해준다.

1. Quiz\_업로드한 파일 s3.jsp에서 바로 띄어주기
   1. 이미지 업로드때와 같이

<form action="uploadImageV2" method="post" enctype="multipart/form-data">

<input type="file" name="uploadFile">

<input type="submit" val="클릭">

</form>

해서 controller에 파일을 보내준다

* 1. 먼저 파일을 업로드하면서 파일 이름을 반환해줄 수 있게
  2. iAWSServ.uploadImageV2(buckName(“johyeonki-bucket”으로 고정), fileName)메소드를 만들어준다.
  3. uploadImageV2메소드는 기존의 uploadImage와 똑같지만 파일 이름을 리턴해준다.
  4. 업로드하고 바로 이미지를 띄우게하면 Access Denied가 출력되며 이미지가 나오지 않는다.
  5. 5번에서 만들었던 iAWSServ.setAccessCtrlLst(“johyeonki-bucket”, fileName))으로 권한을 모든 유저그룹 (GroupGrantee.AllUsers)에게 읽기 권한을 부여해준다. (Permission.Read)
  6. 그리고, 3번에서 만든 파일을 읽어오는 메소드로 이미지URL을 s3.jsp에 보내준다.

model.addAttribute("imageUrl\_2", iAWSServ.getImageURL("johyeonki-bucket", fileName));

* 1. fileName은 c의 메소드에서 리턴된 값을 이용한다.
  2. s3.jsp에서 <img src="${imageUrl\_2 }">태그로 이미지를 띄워준다.