

**IBM SPSS Collaboration and Deployment
Services - Essentials for Python**

버전 8 릴리스 1

개발자 안내서



참고

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 67 페이지의 『주의사항』에 있는 정보를 확인하십시오.

제품 정보

이 개정판은 새 개정판에 별도로 명시하지 않는 한, IBM SPSS Collaboration and Deployment Services 버전 8, 릴리스 1, 수정 레벨 0 및 모든 후속 릴리스와 수정에 적용됩니다.

© Copyright IBM Corporation 2000, 2017.

목차

제 1 장 IBM SPSS Collaboration and Deployment Services - Essentials for Python. 1	프로세스 관리 조작 26
개요 1	제 3 장 PESImpl 모듈. 29
설치 1	컨텐츠 리포지토리 API. 30
더 이상 사용되지 않는 기능 1	메소드 30
제 2 장 명령행 스크립트 3	랩퍼 클래스 55
글로벌 키워드 3	프로세스 관리 API 58
리포지토리 연결 4	메소드 58
컨텐츠 리포지토리 스크립팅 5	랩퍼 클래스 62
키워드 5	예제 스크립트 65
컨텐츠 리포지토리 조작 6	주의사항 67
프로세스 관리 기능 25	개인정보 보호정책 고려사항 69
키워드 25	상표. 69

제 1 장 IBM SPSS Collaboration and Deployment Services - Essentials for Python

개요

IBM® SPSS® Collaboration and Deployment Services는 API 세트를 스크립팅 프레임워크에 제공하며, 고급 사용자 및 관리자가 작업할 루틴 세트와 리포지토리 오브젝트 및 작업을 결합하는 개별적인 루틴 또는 일괄처리 작업을 쓰는 데 이를 사용할 수 있습니다. 이 기능을 통해 다음과 같이 대량의 태스크를 크게 간소화할 수 있습니다.

- 대규모 사용자 그룹에 대한 보안 권한 변경
- 다수의 폴더 또는 파일에서 레이블 제거 또는 레이블링
- 다수의 폴더 또는 파일 업로드 및 다운로드

프레임워크에는 명령행에서 태스크를 수행할 수 있는 기능 및 사용자 소유의 Python 코드 내에서 IBM SPSS Collaboration and Deployment Services Repository와 상호작용하기 위한 리치 API가 포함됩니다.

동적 오브젝트 지향 프로그래밍 언어인 Python에 대한 일반 정보는 Python 사이트를 참조하십시오.

설치

Windows 및 UNIX 플랫폼에 스크립팅 프레임워크를 설치할 수 있습니다. 스크립팅 플랫폼은 스크립팅 기능으로 액세스되는 리포지토리에서 사용한 플랫폼과 관련이 없습니다.

예를 들어, Windows 플랫폼에서 실행되는 리포지토리는 UNIX 플랫폼에서 실행되는 스크립팅 기능으로 호출될 수 있습니다.

설치 지시사항은 IBM SPSS Collaboration and Deployment Services - Essentials for Python 설치 안내서를 참조하십시오.

더 이상 사용되지 않는 기능

IBM SPSS Collaboration and Deployment Services의 이전 릴리스에서 마이그레이션하는 경우 마지막 버전 이후에 더 이상 사용되지 않는 다양한 기능을 알고 있어야 합니다.

기능이 더 이상 사용되지 않으면 IBM Corp.가 이 제품의 후속 릴리스에서 해당 기능을 제거할 수 있습니다. 미래에 대한 투자는 권장되는 마이그레이션 조치 아래에 나열된 전략적 기능에 중점을 둡니다. 일반적으로, 동등한 대체 기능이 제공되지 않으면 기능이 더 이상 사용되지 않습니다.

다음 표는 더 이상 사용되지 않는 기능을 표시합니다. 가능한 경우 권장되는 마이그레이션 조치도 제공합니다.

표 1. 이전 버전에서 더 이상 사용되지 않는 기능

더 이상 사용되지 않음	권장되는 마이그레이션 조치
보안 제공자: 확장된 그룹 및 허용된 사용자를 지원하는 로컬 대체가 포함된 Active Directory	추가된 필수 그룹이 포함된 표준 Active Directory 보안 제공자 사용
IBM SPSS Collaboration and Deployment Services Enterprise View	분석 데이터 보기 기능 사용
IBM SPSS Collaboration and Deployment Services Enterprise View Driver	분석 데이터 보기 기능 사용
시나리오 파일	시나리오 파일(.scn)은 더 이상 지원되지 않습니다. Deployment Manager에서 Enterprise View 소스 노드를 수정할 수 없습니다. IBM SPSS Modeler 클라이언트에서 이전 시나리오 파일을 수정할 수 있으며 스트림 파일로 다시 저장할 수 있습니다. 또한 시나리오 파일을 사용한 스코어링 구성을 삭제하고 스트림 파일에 따라 다시 작성해야 합니다.
IBM SPSS Deployment Manager의 웹 설치	독립형 설치 프로그램 사용
BIRT Report Designer for IBM SPSS	없음
BIRT Report Designer for IBM SPSS 뷰어	없음
IBM SPSS Collaboration and Deployment Services Portlet	IBM SPSS Collaboration and Deployment Services Deployment Portal 직접 사용 또는 웹 서비스 API 사용
IBM SPSS Collaboration and Deployment Services Web Part	IBM SPSS Collaboration and Deployment Services Deployment Portal 직접 사용 또는 웹 서비스 API 사용
스코어링 서비스 V1 API	스코어링 서비스 V2 API
스케줄링 서버 서비스	없음
보고 서비스	없음
인증 서비스 login 조작	인증 서비스 doLogin 조작
검색 서비스 search 조작	검색 서비스 search2.5조작
SPSS AXIS/Castor 웹 서비스 클라이언트 jar	JRE(Java Runtime Environment), IDE(Integrated Development Environment) 또는 Eclipse WTP(Web Tools Platform)에서 제공하는 도구 사용

더 이상 사용되지 않는 기능에 대한 최신 정보는 IBM Knowledge Center를 참조하십시오.

제 2 장 명령행 스크립트

명령행에서 CADSTool.py Python 파일을 사용하여 IBM SPSS Collaboration and Deployment Services Repository 내에 저장된 자원을 조작할 수 있습니다.

명령행에서 IBM SPSS Collaboration and Deployment Services 스크립팅 조작을 호출하는 데 사용되는 일반 구문은 다음과 같습니다.

```
python CADSTool.py <Operation> <Keywords>
```

여기서,

- <Operation>은 호출할 함수를 지정합니다.
- <Keywords>는 함수에 입력 매개변수로 사용된 키워드/값 쌍을 정의합니다.

글로벌 키워드

표 2는 모든 IBM SPSS Collaboration and Deployment Services 스크립팅 기능으로 지원되는 키워드를 나열합니다. 두 번째 열은 키워드의 선택적 축약 버전을 나열합니다. 키워드는 대소문자를 구분합니다.

표 2. 글로벌 키워드.

키워드	선택적 축약 버전	사용법
--user	-u	<p>리포지토리 서버에 연결할 사용자 이름. 사용자가 네이티브 제공자에 있지 않은 경우 값에는 보안 제공자를 표시하는 접두부가 포함되어야 합니다. 유효한 접두부 값은 다음과 같습니다.</p> <ul style="list-style-type: none"> • 시스템에 고유한 네이티브 로컬 보안 제공자의 네이티브. 이는 기본 제공자입니다. • Active Directory의 AD_<name>. 여기서, <name>은 시스템 내의 보안 제공자 이름에 해당합니다. • 로컬 대체가 사용된 Active Directory의 ADL_<name>. 여기서, <name>은 시스템 내의 보안 제공자 이름에 해당합니다. • OpenLDA의 ldap_<name>. 여기서, <name>은 시스템 내의 보안 제공자 이름에 해당합니다. <p>접두부로 슬래시를 사용하여 사용자 이름이 표시됩니다. Active Directory 제공자의 경우에는 접두부에 도메인을 포함하십시오. 예를 들어, AD_SPSSAD Active Directory 인스턴스의 MYDOMAIN 도메인에 있는 icrod 사용자의 경우 값은 AD_SPSSAD/MYDOMAIN/icrod입니다. icrod 사용자가 SPSSLDAP OpenLDAP 제공자에 있는 경우 값은 ldap_SPSSLDAP/icrod입니다.</p>

표 2. 글로벌 키워드 (계속).

키워드	선택적 축약 버전	사용법
--password	-p	리포지토리 서버에 연결할 비밀번호
--host	-q	리포지토리가 설치되는 호스트/서버 이름
--port	-o	리포지토리 서버 포트 번호
--ssl		리포지토리 서버는 SSL(Secure Sockets Layer) 프로토콜을 사용하여 통신을 암호화함을 표시함. 이 키워드를 사용하는 경우 리포지토리 서버가 SSL에 대해 구성되어야 합니다. 자세한 정보는 관리자 문서를 참조하십시오.
--server_url	-S	리포지토리 서버의 전체 URL. 서버의 URL에 사용자 정의 컨텍스트 루트가 포함된 경우 이 키워드를 사용하십시오. 서버 URL을 지정하는 경우 host, port 및 ssl 키워드의 값은 필요하지 않습니다.
--useDefault	-z	Authorization.properties 파일에 정의된 서버 연결 정보를 사용합니다.
--help	-h	스크립팅 모듈 도움말 정보

리포지토리 연결

모든 명령의 끝에 IBM SPSS Collaboration and Deployment Services Repository 사용자 ID, 비밀번호 및 리포지토리 서버 정보를 지정해야 합니다.

다음은 이 연결 정보를 제공하는 데 사용할 수 있는 방법입니다.

- 키워드 사용. 예를 들면, 다음과 같습니다.

```
--user user --password password --host host --port port
--user user --password password --server_url url
```

- Authorization.properties 파일 사용. 여기서, 명령에는 --useDefault 매개변수(또는 축약 버전인 -z)가 포함됩니다. 이 접근법은 *Scripting folder\Lib\site-packages\config\Authorization.properties*에 있는 Authorization.properties 파일에서 연결 정보를 검색합니다. 리포지토리의 설정과 일치시키려면 단순 텍스트 편집기를 사용하여 파일에서 다음 값을 수정하십시오.

```
# Authorization Information
user=admin
password=pwd
host=yourhost
port=80
```

또는 특성 파일에서 server_url 키워드를 사용할 수 있습니다.

```
# Authorization Information
user=admin
password=pwd
server_url=http://yourhost:80/context_root
```

명령행을 통해 전달된 매개변수에는 항상 우선됩니다. 예를 들어, 명령행에 --user 및 --password 매개변수가 제공되고 --useDefault 또는 -z 매개변수도 제공되는 경우 명령행의 사용자와 매개변수가

사용되며 호스트와 포트가 `Authorization.properties` 파일에서 검색됩니다. 또는 사용자, 비밀번호, 호스트 및 포트가 모두 명령행에 제공되지만 `--useDefault` 또는 `-z` 매개변수도 사용되는 경우 `--useDefault`는 무시되며 명령행 정보만 사용됩니다.

여기에 설명된 모든 API의 경우, 최소한의 필수 매개변수를 사용하기 위해 구문 및 예제에서 `-z` 매개변수를 사용합니다.

컨텐츠 리포지토리 스크립팅

컨텐츠 리포지토리 스크립팅은 파일 및 폴더와 같은 리포지토리 자원에 대해 작업할 수 있는 기능을 제공합니다. 이 영역에는 다음 기능이 포함되어 있습니다.

- 폴더 작성 및 삭제
- 파일 업로드 및 다운로드
- 폴더 내보내기 및 가져오기
- 레이블, 보안 및 메타데이터 관리

이 절에서는 리포지토리 기능에 대한 스크립트의 Python 명령행 사용법에 대해 간략하게 설명합니다. 모든 조작에는 상세한 구문 정보, 예제 및 예상되는 메시지가 포함됩니다.

키워드

표 3은 리포지토리 기능에 지원되는 키워드를 나열합니다. 두 번째 열은 키워드의 선택적 축약 버전을 나열합니다.

중요: 키워드는 대소문자를 구별합니다.

표 3. 리포지토리 API에 대한 키워드.

키워드	선택적 축약 버전	사용법
<code>--source</code>	<code>-s</code>	소스 파일 또는 폴더 경로
<code>--target</code>	<code>-t</code>	대상 폴더 경로
<code>--version</code>	<code>-v</code>	파일 버전
<code>--principal</code>	<code>-r</code>	권한이 부여되어야 할 사용자
<code>--permission</code>	<code>-n</code>	권한 유형(예: 읽기, 쓰기, 수정, 삭제)
<code>--label</code>	<code>-l</code>	파일 버전에 지정할 레이블
<code>--criteria</code>	<code>-c</code>	파일 또는 폴더의 메타데이터 속성을 검색하기 위한 검색 기준
<code>--author</code>	<code>-a</code>	파일 또는 폴더에 대한 작성자 이름
<code>--description</code>	<code>-d</code>	파일 또는 폴더에 대한 설명
<code>--title</code>	<code>-i</code>	파일 또는 폴더에 대한 제목
<code>--expirationDate</code>	<code>-q</code>	파일 또는 폴더에 대한 만료 날짜
<code>--expirationStartDate</code>		파일 또는 폴더에 대한 만료 시작 날짜
<code>--expirationEndDate</code>		파일 또는 폴더에 대한 만료 종료 날짜
<code>--keyword</code>	<code>-k</code>	파일 또는 폴더에 대한 키워드

표 3. 리포지토리 API에 대한 키워드 (계속).

키워드	선택적 축약 버전	사용법
--cascade	-x	폴더에 대한 보안 설정이 하위 폴더 및 파일에 전파되어야 함을 표시함
--provider	-f	프린시펄을 검색하는 데 사용된 보안 제공자
--createVersion	-b	작성될 파일의 새 버전을 표시함
--contentLanguage	-g	파일 또는 폴더에 대한 콘텐츠 언어
--topic		파일 또는 폴더에 지정된 토픽. --topic "topic1;topic2"와 같이 다중 값을 입력할 수 있습니다.
--modifiedBy		파일 또는 폴더를 수정한 사용자
--mimeType		파일의 MIME 유형
--createdBy		파일 또는 폴더를 작성한 사용자
--submittedHierarchy		<i>Submitted Jobs</i> 폴더의 검색 여부를 표시함
--propertyName		사용자 정의 특성의 이름
--customProperty		업데이트될 사용자 정의 특성의 이름-값 쌍
--propertyName		유효한 값을 검색할 사용자 정의 특성의 이름

레이블 및 버전 정보를 허용하는 모든 조작의 경우 사용자는 레이블 또는 버전 중 하나(둘 다는 아님)를 지정해야 합니다. 버전 또는 레이블이 제공된 파일에 지정되지 않는 경우 최신 버전이 사용됩니다.

컨텐츠 리포지토리 조작

advanceSearch 조작

여러 매개변수를 기반으로 리포지토리에서 파일 및 폴더를 검색합니다.

구문

```
python CADSTool.py advanceSearch --author <author>
--title <title> --description <description>
--createdBy <createdBy> --modifiedBy <modifiedBy>
--keyword <keyword> --label <label>
--topic <topic>
--uri <uri> --parentURI <parentURI>
--expirationStartDate <expirationStartDate>
--expirationEndDate <expirationEndDate>
--createdStartDate <createdStartDate>
--createdEndDate <createdEndDate>
--objectModifiedStartDate <objectModifiedStartDate>
--objectModifiedEndDate <objectModifiedStartDate>
--versionModifiedStartDate <versionModifiedStartDate>
--versionModifiedEndDate <versionModifiedEndDate>
--submittedHierarchy -z
```

여기서,

- <author>는 작성자의 이름입니다.
- <title>은 파일/폴더의 제목입니다.
- <description>은 파일/폴더의 설명입니다.

모든 매개변수는 선택사항입니다.

예제

```
python CADSTool.py advanceSearch --label "Production" --keyword "Quarterly"  
--useDefault -z
```

```
python CADSTool.py advanceSearch --createdStartDate "2009-12-01T00:00:00+01:00"  
--createdEndDate "2010-12-15T21:33:40+01:00" -z
```

```
python CADSTool.py advanceSearch --uri "spssc:///id=a010a37ba5992bb00000127b0f952f945be" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- API가 완료될 때 검색 기준과 일치하는 모든 파일 및 폴더의 목록이 표시됩니다. 일반적으로 이 목록에는 완전한 경로 및 버전과 함께 파일 이름이 포함됩니다.
- 파일 및 폴더를 검색하는 중에 오류가 발생했습니다.
- 오류: 해당 옵션 없음:<option>

applySecurity 조작

리포지토리에서 파일 또는 폴더에 대한 보안 액세스 제어 목록(ACL)을 설정합니다.

구문

```
python CADSTool.py applySecurity --source "<source>" --principal "<principal>"  
--permission "<permission>" --provider "<provider>" --cascade -z
```

여기서,

- <source>는 보안 ACL을 적용할 파일 또는 폴더의 완전한 IBM SPSS Collaboration and Deployment Services Repository 경로입니다. 이는 필수 매개변수입니다.
- <principal>은 ACL의 일부로 지정된 파일 또는 폴더에 적용할 사용자(예: *admin*)입니다. 이는 필수 매개변수입니다.
- <permission>은 지정된 파일 또는 폴더에 적용할 권한의 유형(예: 읽기, 쓰기, 수정, 삭제 또는 소유자)입니다. 이는 필수 매개변수입니다.
- <provider>는 사용자(프린시펄)에 대한 정보를 검색하는 데 사용하는 보안 제공자입니다. 이는 선택적 매개변수입니다. 유효한 값은 다음과 같습니다.
 - 시스템에 고유한 네이티브 로컬 보안 제공자의 네이티브. 이는 기본 제공자입니다.
 - Active Directory의 AD_<name>. 여기서, <name>은 시스템 내의 보안 제공자 이름에 해당합니다.
 - 로컬 대체가 사용된 Active Directory의 ADL_<name>. 여기서, <name>은 시스템 내의 보안 제공자 이름에 해당합니다.
 - OpenLDA의 ldap_<name>. 여기서, <name>은 시스템 내의 보안 제공자 이름에 해당합니다.

- --cascade는 지정된 폴더 내의 모든 파일 및 하위 폴더에 보안 설정을 전파하기 위해 폴더에 보안을 설정할 때 사용됩니다. 이는 선택적 매개변수입니다.

예제

다음 예제에서는 폴더에 보안을 적용합니다.

```
python CADSTool.py applySecurity --source "/Projects" --principal "icrod"
--permission "READ" --provider "Native" -z
```

다음 예제에서는 폴더 및 모든 해당 파일과 하위 폴더에 보안을 적용합니다.

```
python CADSTool.py applySecurity --source "/Projects/" --principal "icrod"
--permission "READ" --provider "Native" --cascade -z
```

다음 예제에서는 SPSSAD라는 Active Directory 보안 제공자의 프린시펄에 대한 폴더에 보안을 적용합니다.

```
python CADSTool.py applySecurity --source "/Projects" --principal "ICrod (MYDOMAIN)"
--permission "Write" --provider "AD_SPSSAD" -z
```

다음 예제에서는 LDAP라는 OpenLDAP 보안 제공자의 프린시펄에 대한 폴더에 보안을 적용합니다.

```
python CADSTool.py applySecurity --source "/Projects" --principal "ICrod (LDAP)"
--permission "Read" --provider "ldap_LDAP" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <permission> <source>에 대한 권한을 설정합니다.
- <source> 해당 파일 또는 폴더가 없습니다. 다시 시도하십시오.
- <permission> 유효하지 않은 권한 유형입니다. 다시 시도하십시오.
- <source> 보안 ACL을 설정하는 중에 발생한 오류입니다.

cascadeSecurity 조작

폴더 내의 모든 파일 및 하위 폴더에 폴더의 보안 설정을 전파합니다.

구문

```
python CADSTool.py cascadeSecurity --source "<source>" -z
```

<source> 값은 리포지토리에서 폴더의 완전한 경로입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py cascadeSecurity --source "/Projects" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source>에 대한 보안 ACL이 단계적으로 적용되었습니다.

- <source> 해당 폴더가 없습니다. 다시 시도하십시오.
- <source> 보안 ACL을 단계적으로 적용하는 중에 발생한 오류입니다.

copyResource 조작

리포지토리에서 파일 또는 폴더를 다른 폴더에 복사합니다.

이 API에 대한 이름 변경 기능이 제공됩니다. 여기서, 지정된 파일이 복사될 때 이름을 변경할 수 있습니다. 19 페이지의 『moveResource 조작』 시작 시 설명된 케이스가 copyResource API에도 적용될 수 있습니다.

구문

```
python CADSTool.py copyResource --source "<source>" --target "<target>" -z
```

여기서,

- <source>는 복사할 파일/폴더의 완전한 콘텐츠 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- <target>은 파일/폴더를 복사할 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.

예제

다음 예제에서는 파일을 복사합니다.

```
python CADSTool.py copyResource --source "/Demo/Drafts/MyReport.rptdesign" --target "/Projects" -z
```

다음 예제에서는 파일을 복사하고 이름을 변경합니다.

```
python CADSTool.py copyResource --source "/Demo/Drafts/MyReport.rptdesign" --target "/Projects/Report.rptdesign" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source>가 복사되었습니다.
- <source> 해당 파일 또는 폴더가 없습니다. 다시 시도하십시오.
- <target> 해당 폴더가 없습니다. 다시 시도하십시오.
- <source> 파일 또는 폴더를 복사하는 중에 오류가 발생했습니다.

createFolder 조작

리포지토리의 지정된 위치에 새 폴더를 작성합니다.

구문

```
python CADSTool.py createFolder --source "<source>" -z
```

<source> 값은 작성할 새 폴더의 완전한 경로입니다. 이는 필수 매개변수입니다. 제공된 경로를 기반으로 하위 폴더를 포함한 새 폴더가 작성됩니다.

예제

다음 예제에서는 초안을 작성합니다(아직 존재하지 않는 경우).

```
python CADSTool.py createFolder --source "/Demo/Drafts" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source> 폴더가 작성되었습니다.
- <source> 해당 폴더가 없습니다. 다시 시도하십시오.
- <folder> 폴더가 이미 있습니다. 다시 시도하십시오.
- <source> 폴더를 작성하는 중에 오류가 발생했습니다.

deleteFile 조작

리포지토리에서 모든 버전을 포함한 파일을 삭제합니다.

구문

```
python CADSTool.py deleteFile --source "<source>" --submittedHierarchy -z
```

여기서,

- <source>는 삭제할 파일의 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- --submittedHierarchy는 Submitted Jobs 폴더에서 파일을 삭제합니다. 이는 선택적 매개변수입니다.

예제

다음 예제는 리포지토리에서 모든 버전을 포함한 파일을 삭제합니다.

```
python CADSTool.py deleteFile --source "/Demo/Drafts/MyReport.rptdesign" -z
```

다음 예제는 Submitted Jobs 폴더에서 모든 버전을 포함한 파일을 삭제합니다.

```
python CADSTool.py deleteFile --source "Submitted Jobs/admin/  
2007-05-21.14.10.22.422-test.dbq/test.dbq.html" --submittedHierarchy -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source>가 삭제되었습니다.
- <source> 해당 파일이 없습니다. 다시 시도하십시오.
- <source> 파일을 삭제하는 중에 오류가 발생했습니다.

deleteFileVersion 조작

리포지토리에서 파일의 특정 버전을 삭제합니다.

구문

```
python CADSTool.py deleteFileVersion --source "<source>" --version "<version>"  
--label "<label>" --submittedHierarchy -z
```

여기서,

- <source>는 삭제할 파일의 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- <version>은 삭제할 파일의 특정 버전입니다. 이는 선택적 매개변수입니다.
- <label>은 삭제할 파일의 레이블입니다. 이는 선택적 매개변수입니다.
- --submittedHierarchy는 Submitted Jobs 폴더에서 파일의 특정 버전을 삭제합니다. 이는 선택적 매개변수입니다.

예제

다음 예제에서는 파일의 특정 버전을 삭제합니다.

```
python CADSTool.py deleteFileVersion --source "/Demo/Drafts/MyReport.rptdesign" --version  
"0:2006-08-25 21:15:49.453" -z
```

다음 예제에서는 특정 레이블이 있는 파일을 삭제합니다.

```
python CADSTool.py deleteFileVersion --source "/Demo/Drafts/MyReport.rptdesign" --label  
"Test" -z
```

다음 예제에서는 Submitted Jobs 폴더에서 특정 레이블이 있는 파일을 삭제합니다.

```
python CADSTool.py deleteFileVersion --source "Submitted Jobs/admin/  
2007-05-21.14.10.22.422-test.dbq/test.dbq.html" --label "Test" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source>가 삭제되었습니다.
- <source> 해당 파일이 없습니다. 다시 시도하십시오.
- <source> 파일을 삭제하는 중에 오류가 발생했습니다.

deleteFolder 조작

리포지토리에서 모든 콘텐츠를 포함한 폴더를 삭제합니다.

구문

```
python CADSTool.py deleteFolder --source <source> --submittedHierarchy -z
```

여기서,

- <source>는 삭제할 폴더의 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- --submittedHierarchy는 Submitted Jobs 폴더에서 폴더의 특정 버전을 삭제합니다. 이는 선택적 매개변수입니다.

예제

다음 예제에서는 폴더를 삭제합니다.

```
python CADSTool.py deleteFolder --source "/Demo/Drafts" -z
```

다음 예제에서는 Submitted Jobs 폴더에서 폴더를 삭제합니다.

```
python CADSTool.py deleteFolder --source "Submitted Jobs/admin/  
2007-05-21.14.10.22.422-test.dbq/" --submittedHierarchy -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source>가 삭제되었습니다.
- <source> 해당 폴더가 없습니다. 다시 시도하십시오.
- <source> 폴더를 삭제하는 중에 오류가 발생했습니다.

downloadFile 조작

파일의 특정 버전을 리포지토리에서 로컬 파일 시스템으로 다운로드합니다.

구문

```
python CADSTool.py downloadFile --source "<source>" --version "<version>" --label "<label>" --target "<target>" -z
```

여기서,

- <source>는 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- <version>은 다운로드할 파일의 버전입니다. 이는 선택적 매개변수입니다.
- <label>은 다운로드될 파일의 레이블입니다. 이는 선택적 매개변수입니다.
- <target>는 파일이 다운로드될 완전한 경로(로컬 파일 시스템에서)입니다.

예제

다음 예제에서는 파일의 최신 버전을 다운로드합니다.

```
python CADSTool.py downloadFile --source "/Demo/Drafts/MyReport.rptdesign"  
--target "C:/Demo/Shared/" -z
```

다음 예제에서는 버전 마커를 사용하여 파일의 특정 버전을 다운로드합니다.

```
python CADSTool.py downloadFile --source "/Demo/Drafts/MyReport.rptdesign" --version  
"0:2006-08-25 21:15:49.453" --target "C:/Demo/Shared/" -z
```

다음 예제에서는 파일의 레이블된 버전을 다운로드합니다.

```
python CADSTool.py downloadFile --source "/Demo/Drafts/MyReport.rptdesign" --label "Production"  
--target "C:/Demo/Shared/" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source> 파일이 다운로드되었습니다.
- <source> 해당 파일이 없습니다. 다시 시도하십시오.
- <target> 해당 폴더가 없습니다. 다시 시도하십시오.
- <source> 파일을 다운로드하는 중에 오류가 발생했습니다.

export 조작

사용자가 내보낼 파일 및 폴더를 선택하도록 하고 로컬 파일 시스템에 *.pes 내보내기 파일을 저장하여 콘텐츠 리포지토리에서 내보내기를 시작합니다.

구문

```
python CADSTool.py export --source "<source>" --target "<target>" -z
```

여기서,

- <source>는 내보낼 폴더의 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- <target>은 작성할 *.pes 내보내기 파일의 완전한 경로(로컬 파일 시스템에서)입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py export --source "/Projects/" --target "C:\Demo\drafts.pes" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source>가 내보내기되었습니다.
- <source> 해당 폴더가 없습니다. 다시 시도하십시오.
- <source> 폴더를 내보내는 중에 오류가 발생했습니다.

getAccessControlList 조작

콘텐츠 리포지토리에서 선택된 파일/폴더에 대한 보안 액세스 제어 목록(ACL)을 검색합니다.

구문

```
python CADSTool.py getAccessControlList --source "<source>" -z
```

<source> 값은 파일/폴더의 완전한 경로입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py getAccessControlList --source "/Projects/MyReport.rptdesign" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- `<source>` 해당 파일 또는 폴더가 없습니다. 다시 시도하십시오.
- `<source>`에 대한 보안 세부사항을 검색하는 중에 오류가 발생했습니다.

getAllVersions 조작

리포지토리에서 파일의 모든 버전 목록을 검색합니다.

구문

```
python CADSTool.py getAllVersions --source "<source>" --submittedHierarchy -z
```

여기서,

- `<source>`는 버전을 검색할 파일의 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- `--submittedHierarchy`는 Submitted Jobs 폴더에서 버전을 검색합니다. 이는 선택적 매개변수입니다.

예제

다음 예제에서는 지정된 파일의 모든 버전을 검색합니다.

```
python CADSTool.py getAllVersions --source "/Demo/Drafts/MyReport.rptdesign" -z
```

다음 예제에서는 Submitted Jobs 폴더에서 지정된 파일의 모든 버전을 검색합니다.

```
python CADSTool.py getAllVersions --source "Submitted Jobs/admin/2007-05-21.14.10.22.422-test.dbq/test.dbq.html" --submittedHierarchy -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- `<source>` 해당 파일이 없습니다. 다시 시도하십시오.
- `<source>` 파일 버전을 검색하는 중에 오류가 발생했습니다.
- 프로세스가 완료되면 버전 마커 및 레이블 정보를 포함한 모든 파일 버전의 정보가 표시됩니다.

getChildren 조작

리포지토리의 지정된 폴더에서 모든 파일 및 폴더의 목록을 검색합니다.

구문

```
python CADSTool.py getChildren --source "<source>" -z
```

`<source>` 값은 폴더의 완전한 경로입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py getChildren --source "/Demo/Drafts" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- 명령이 완료되면 지정된 폴더의 모든 콘텐츠를 나열합니다.
- `<source>` 해당 폴더가 없습니다. 다시 시도하십시오.
- `<source>` 자원을 가져오는 중에 오류가 발생했습니다.

getCustomPropertyValue 조작

지정된 사용자 정의 특성으로 허용된 유효한 값을 검색합니다.

구문

```
python CADSTool.py getCustomPropertyValue --propertyName "<propertyName>" -z
```

`<propertyName>` 값은 사용자 정의 특성의 이름입니다. 이는 선택적 매개변수입니다.

예제

```
python CADSTool.py getCustomPropertyValue --propertyName "Language" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- `<propertyName>`이 값을 `<valid values>`로 사용합니다.
- `<propertyName>`에 대한 특성 세부사항을 검색하는 중에 오류가 발생했습니다.

getMetadata 조작

리포지토리에서 파일 또는 폴더의 메타데이터 속성을 검색합니다.

구문

```
python CADSTool.py getMetadata --source "<source>" --version "<version>" --label  
"<label>" --submittedHierarchy -z
```

여기서,

- `<source>`는 메타데이터를 검색할 파일 또는 폴더의 완전한 리포지토리 경로입니다. 폴더의 경우, 버전/레이블 속성이 무시됩니다. 이는 필수 매개변수입니다.
- `<version>`은 메타데이터를 검색할 파일의 버전입니다. 이는 선택적 매개변수입니다.
- `<label>`은 메타데이터를 검색할 파일의 메타데이터입니다. 이는 선택적 매개변수입니다.
- `--submittedHierarchy`는 Submitted Jobs 폴더에서 메타데이터를 검색합니다. 이는 선택적 매개변수입니다.

예제

다음 예제에서는 폴더에 대한 메타데이터를 검색합니다.

```
python CADSTool.py getMetadata --source "/Demo/Drafts" -z
```

다음 예제에서는 파일의 레이블된 버전에 대한 메타데이터를 검색합니다.

```
python CADSTool.py getMetadata --source "/Demo/Drafts/MyReport.rptdesign" --label "Test" -z
```

다음 예제에서는 Submitted Jobs 폴더에서 파일의 레이블된 버전에 대한 메타데이터를 검색합니다.

```
python CADSTool.py getMetadata --source "Submitted Jobs/admin/  
2007-05-21.14.10.22.422-test.dbq/test.dbq.html" --label "LATEST" --submittedHierarchy -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- `<source>` 해당 파일이 없습니다. 다시 시도하십시오.
- `<source>` 파일 메타데이터를 검색하는 중에 오류가 발생했습니다.
- 프로세스가 완료되면 사용자 정의 메타데이터 특성을 포함한 지정된 파일 또는 폴더에 대한 모든 메타데이터 정보가 표시됩니다.

import 조작

로컬 파일 시스템에서 리포지토리로 기존 *.pes 내보내기 파일을 가져옵니다.

구문

```
python CADSTool.py import --source "<source>" --target "<target>"  
--resourceType "<type>"  
--resourceConflict "<rconflict>"  
--labelFrom "<label>"  
--lockResolution "<resolution>"  
--invalidVersionConflict "<vconflict>"  
--resourceDef "<rdefinition>"  
--exclude "<exclusion>"  
-z
```

여기서,

- `<source>`는 리포지토리로 가져올 *.pes 내보내기 파일의 완전한 경로(로컬 파일 시스템에서)입니다. 이는 필수 매개변수입니다.
- `<target>`은 *.pes 내보내기 파일을 가져올 완전한 리포지토리 경로(로컬 파일 시스템에서)입니다. 이는 필수 매개변수입니다.
- `<type>`은 가져오는 콘텐츠의 유형을 표시합니다. 다음 값 중 하나를 지정하십시오.
 - **ContentRepository** - 파일 및 폴더와 같은 콘텐츠 오브젝트
 - **ResourceDef** - 자원 정의 가져오기
 - **Credential** - 사용자 신임 정보
 - **DataSource** - 데이터 소스 정의
 - **MessageDomain** - 메시지 도메인
 - **ServerCluster** - 서버 클러스터 정의
 - **Server** - 서버 정의

- **PromotionPolicy** - 프로모션 정책

이는 선택적 매개변수입니다. 이 매개변수가 지정되지 않은 경우 **ContentRepository**의 기본값이 사용됩니다.

- `<rconflict>`는 중복 ID 또는 이름 충돌을 해결하는 방법을 표시합니다. 다음 값 중 하나를 지정하십시오.
 - **keepTarget**. 대상 항목이 유지됩니다. `.pes` 파일에 포함된 중복 ID가 있는 소스 항목이 무시됩니다.
 - **addNewVersion**. 일반적으로 이 옵션은 ID 충돌 또는 이름 충돌을 해결하는 데 사용됩니다. 소스 오브젝트 및 대상 오브젝트 간에 중복 ID 충돌이 발생하는 경우 오브젝트의 새 버전이 대상 위치에 작성됩니다. 이름 충돌이 발생하면 가져온 오브젝트의 이름이 대상 위치에서 변경됩니다. 일반적으로 이름이 변경된 오브젝트에 `_1`, `_2` 등이 추가됩니다. 오브젝트의 두 버전에 동일한 레이블이 있는 경우, 동일한 항목의 두 버전에 동일한 레이블이 있을 수 없으므로 시스템에서 하나의 레이블을 유지하고 중복 레이블을 삭제합니다. 보유한 레이블은 **labelFrom** 매개변수에 종속됩니다.

이는 선택적 매개변수입니다. 이 매개변수가 지정되지 않은 경우 **keepTarget**의 기본값이 사용됩니다.

- `<label>`은 오브젝트의 두 버전에 동일한 레이블이 있는 경우 사용할 레이블을 지정합니다. 다른 버전의 레이블이 삭제됩니다. 다음 값 중 하나를 지정하십시오.
 - **source**
 - **target**

이는 선택적 매개변수입니다. 이 매개변수가 지정되지 않은 경우 **source**의 기본값이 사용됩니다.

- `<resolution>`은 잠긴 자원이 발생하는 경우 진행 방법을 정의합니다. 다음 값 중 하나를 지정하십시오.
 - **continue**. 잠긴 자원을 생략하고 가져오기를 계속합니다.
 - **abort**. 잠긴 자원이 발생하는 경우 가져오기 프로세스를 종료합니다. 오브젝트 잠금으로 인해 충돌이 발생하면 가져오기 프로세스가 종료되고 실패합니다.

이는 선택적 매개변수입니다. 이 매개변수가 지정되지 않은 경우 **abort**의 기본값이 사용됩니다.

- `<vconflict>`는 가져오기 프로세스 중에 올바르지 않은 버전이 발생하는 경우 진행 방법을 정의합니다. 다음 값 중 하나를 지정하십시오.
 - **import**. 올바르지 않은 버전을 가져옵니다.
 - **discard**. 올바르지 않은 버전을 삭제합니다.

이는 선택적 매개변수입니다. 이 매개변수가 지정되지 않은 경우 **import**의 기본값이 사용됩니다.

- `<rdefinition>`이 자원 정의에 대한 처리 동작을 정의합니다. 다음 값 중 하나를 지정하십시오.
 - **recommended**. 식별자 또는 이름이 대상 정의와 충돌하지 않는 경우에만 자원 정의를 가져옵니다. 충돌이 있으면 자원 정의를 가져오지 않습니다.

- **include.** 파일 가져오기에서 모든 자원 정의를 가져옵니다. 해당 선택란을 선택하여 가져오기에서 제외할 하나 이상의 자원 정의 유형을 선택할 수 있습니다.
- **exclude.** 파일 가져오기에서 자원 정의를 가져오지 않습니다. 사용 가능한 자원 정의를 참조하려면 가져온 오브젝트를 수정해야 합니다.

이는 선택적 매개변수입니다. 이 매개변수가 지정되지 않은 경우 **recommended**의 기본값이 사용됩니다.

- **<exclusion>**은 가져오기 중에 제외되는 자원 유형을 정의합니다. 다중 값은 세미콜론으로 구분된 목록으로서 임의의 순서로 조합될 수 있습니다. 다음 값 중 하나 이상을 지정하십시오.

- **credential** - 사용자 신임 정보 제외
- **customproperty** - 자원 오브젝트에 대한 사용자 정의 특성 제외
- **datasource** - 데이터 소스 정의 제외
- **messagedomain** - 메시지 도메인 제외
- **notification** - 알림 정의 제외
- **servercluster** - 서버 클러스터 정의 제외
- **server** - 서버 정의 제외
- **topic** - 토픽 정의 제외

이는 선택적 매개변수입니다. 이 매개변수가 지정되지 않은 경우 모든 유형이 가져오기에 포함됩니다.

예제

```
python CADSTool.py import --source "C:\Demo\drafts.pes" --target "/Demo/Drafts/"
--resourceConflict "addNewVersion" --labelFrom "target" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- **<source>**가 가져오기되었습니다.
- **<source>** 해당 파일이 없습니다. 다시 시도하십시오.
- **<target>** 해당 폴더가 없습니다. 다시 시도하십시오.
- **<source>** 폴더를 가져오는 중에 오류가 발생했습니다.

moveResource 조작

리포지토리에서 파일 또는 폴더를 다른 폴더로 이동합니다.

이 API에 대한 이름 변경 기능이 제공됩니다. 여기서, 지정된 파일/폴더가 이동될 때 이름을 변경할 수 있습니다. 다음 케이스는 이름 변경 기능의 동작에 대해 설명합니다.

소스가 */Temp Folder/Temp.txt* 이고 대상이 */Demo Folder*인 경우:

- **케이스 1:** *Demo Folder* 폴더가 있으면 *Temp.txt*가 *Demo Folder*로 이동됩니다.

- **케이스 2:** *Demo Folder* 폴더가 없으면 *Temp.txt*가 『/』로 이동되고 *Demo Folder*로 이름이 변경됩니다.

소스가 */Temp Folder/Temp.txt* 이고 대상이 */Demo Folder/Abc.dat*인 경우:

- **케이스 1:** *Demo Folder* 폴더가 있으면 *Temp.txt*가 *Demo Folder*로 이동되고 *Abc.dat*로 이름이 변경됩니다.
- **케이스 2:** *Demo Folder* 폴더가 없으면 오류가 표시됩니다.

구문

```
python CADSTool.py moveResource --source "<source>" --target "<target>" -z
```

여기서,

- *<source>*는 이동할 파일/폴더의 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- *<target>*은 파일/폴더를 이동할 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.

예제

다음 예제에서는 파일을 이동합니다.

```
python CADSTool.py moveResource --source "/Demo/Drafts/MyReport.rptdesign" --target "/Approved" -z
```

다음 예제에서는 폴더를 이동합니다.

```
python CADSTool.py moveResource --source "/Demo/Drafts/" --target "/Projects" -z
```

다음 예제에서는 파일을 이동하고 이름을 변경합니다.

```
python CADSTool.py moveResource --source "/Demo/Drafts/MyReport.rptdesign" --target "/Approved/Report.rptdesign" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- *<source>*가 이동되었습니다.
- *<source>* 해당 파일 또는 폴더가 없습니다. 다시 시도하십시오.
- *<target>* 해당 폴더가 없습니다. 다시 시도하십시오.
- *<source>* 파일 또는 폴더를 이동하는 중에 오류가 발생했습니다.

removeLabel 조작

리포지토리의 파일에서 레이블을 제거합니다.

구문

```
python CADSTool.py removeLabel --source "<source>" --label "<label>" -z
```

여기서,

- `<source>`는 리포지토리에서 파일의 완전한 경로입니다. 이는 필수 매개변수입니다.
- `<label>`은 지정된 파일에서 제거할 레이블 이름입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py removeLabel --source "/Demo/Drafts/MyReport.rptdesign"
--label "Draft" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- `<source>`에 대한 레이블을 제거했습니다.
- `<source>` 해당 폴더가 없습니다. 다시 시도하십시오.
- `<source>` 레이블을 삭제하는 중에 오류가 발생했습니다.
- `<label>` 해당 레이블이 없습니다. 다시 시도하십시오.

removeSecurity 조작

리포지토리의 지정된 파일 또는 폴더에서 보안 액세스 제어 목록(ACL)을 제거합니다.

구문

```
python CADSTool.py removeSecurity --source "<source>" --principal "<principal>"
--provider "<provider>" --cascade -z
```

여기서,

- `<source>`는 보안을 제거할 파일/폴더의 완전한 경로입니다. 이는 필수 매개변수입니다.
- `<principal>`은 지정된 파일/폴더에 대한 보안을 제거할 사용자/프린시펄(예: *admin*)입니다. 이는 필수 매개변수입니다.
- `<provider>`는 사용자(프린시펄)에 대한 정보를 검색하는 데 사용하는 보안 제공자입니다. 이는 선택적 매개변수입니다. 유효한 값은 다음과 같습니다.
 - 시스템에 고유한 네이티브 로컬 보안 제공자의 네이티브. 이는 기본 제공자입니다.
 - Active Directory의 AD_`<name>`. 여기서, `<name>`은 시스템 내의 보안 제공자 이름에 해당합니다.
 - 로컬 대체가 사용된 Active Directory의 ADL_`<name>`. 여기서, `<name>`은 시스템 내의 보안 제공자 이름에 해당합니다.
 - OpenLDA의 Idap_`<name>`. 여기서, `<name>`은 시스템 내의 보안 제공자 이름에 해당합니다.
- `--cascade`는 지정된 폴더 내의 모든 파일 및 하위 폴더에서 보안 설정을 제거하기 위해 폴더에서 보안을 제거할 때 사용됩니다. 이는 선택적 매개변수입니다.

예제

```
python CADSTool.py removeSecurity --source "/Projects/MyReport.rptdesign"
--principal "icrod" --provider "Native" --cascade -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source> 모든 보안 ACL이 제거되었습니다.
- <source> 해당 폴더가 없습니다. 다시 시도하십시오.
- <source> 보안 ACL을 삭제하는 중에 발생한 오류입니다.

search 조작

리포지토리에서 파일 및 폴더를 검색합니다. 결과는 검색 기준 및 해당 버전과 일치하는 파일/폴더의 목록입니다.

구문

```
python CADSTool.py search --criteria "<criteria>" -z
```

<criteria> 값은 리포지토리에서 모든 파일 및 폴더에 대한 메타데이터를 검색하는 데 사용된 검색 문자열입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py search --criteria "Quarterly" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- 검색이 완료될 때 검색 기준과 일치하는 모든 파일 및 폴더의 목록이 표시됩니다. 일반적으로 이 목록에는 완전한 경로 및 버전과 함께 파일 이름이 포함됩니다.
- <criteria> 검색 기준과 일치하는 파일 또는 폴더가 없습니다.
- 파일 및 폴더를 검색하는 중에 오류가 발생했습니다.

setLabel 조작

리포지토리에서 파일의 버전에 레이블을 적용합니다. 파일이 이미 레이블된 경우 원래의 레이블이 제거되고 새 레이블로 대체됩니다.

구문

```
python CADSTool.py setLabel --source "<source>" --version "<version>" --label "<label>" -z
```

여기서,

- <source>는 리포지토리에서 파일의 완전한 경로입니다. 이는 필수 매개변수입니다.
- <version>은 레이블에 적용할 파일의 버전입니다. 이는 필수 매개변수입니다.
- <label>은 파일의 지정된 버전에 적용할 레이블 이름입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py setLabel --source "/Demo/Drafts/MyReport.rptdesign" --version "1:2006-08-25 21:15:49.453" --label "Beta" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source>에 대한 레이블이 설정되었습니다.
- <source> 해당 폴더가 없습니다. 다시 시도하십시오.
- <source> 레이블을 설정하는 중에 오류가 발생했습니다.

setMetadata 조작

메타데이터 특성을 리포지토리의 파일 및 폴더에 적용합니다.

표 4는 메타데이터 특성과 이 특성이 파일 및 폴더에 적용될 수 있는지를 나열합니다.

표 4. 메타데이터 특성 및 자원 유형.

메타데이터 특성	자원 유형
작성자	파일
설명	파일 또는 폴더
제목	파일 또는 폴더
만료 날짜	파일 또는 폴더
키워드	파일
토픽	파일
사용자 정의 메타데이터	파일 또는 폴더

구문

```
python CADSTool.py setMetadata --source "<source>" --version "<version>" --label "<label>" --author "<author>" --title "<title>" --description "<description>" --expirationDate "<expirationDate>" --topic "<topic>" --keyword "<keyword>" --customProperty "<customProperty>" -z
```

여기서,

- <source>는 메타데이터를 설정할 파일 또는 폴더의 완전한 리포지토리 경로입니다. 이는 필수 매개변수입니다.
- <author>는 파일 또는 폴더의 작성자입니다. 이는 선택적 매개변수입니다.
- <title>은 파일 또는 폴더의 제목입니다. 이는 선택적 매개변수입니다.
- <description>은 파일/폴더의 설명입니다. 이는 선택적 매개변수입니다.
- <expirationDate>는 파일 또는 폴더의 만료 날짜입니다. 이는 선택적 매개변수입니다. 날짜 형식은 YYYY-MM-DDThh:mm:ssTZD입니다(예: 1997-07-16T19:20:30+01:00). 여기서, 각 의미는 다음과 같습니다.

YYYY = 4자리의 연도

MM = 2자리의 월(01은 1월 등)

DD = 2자리의 일(01 - 31)

hh = 2자리의 시(00 - 23, 오전/오후 없음)

mm = 2자리의 분(00 - 59)

ss = 2자리의 초(00 - 59)

TZD = 시간대 지정자(Z, +hh:mm 또는 -hh:mm)

- `<keyword>`는 파일 또는 폴더의 키워드입니다. 이는 선택적 매개변수입니다.
- `<version>`은 메타데이터를 적용할 파일의 특정 버전입니다. 이는 선택적 매개변수입니다.
- `<label>`은 메타데이터를 적용할 파일의 레이블된 버전입니다. 이는 선택적 매개변수입니다.
- `<topic>`은 파일 또는 폴더에 적용할 토픽입니다. 이는 선택적 매개변수입니다.
- `<customProperty>`는 파일 또는 폴더에 적용할 사용자 정의 특성 값입니다. 이는 선택적 매개변수입니다. 값은 `<customProperty>=<value>`로 지정됩니다. 둘 이상의 사용자 정의 특성을 적용하려면 구분 문자(`<customProperty>=<value>;<customProperty>=<value>`)로 세미콜론(;)을 사용하십시오. | 연산자를 사용하여 다중 선택 특성 값을 구분하십시오 (`<customProperty>=opt1|opt2;<customProperty>=value`).

참고: `setMetadata` API를 사용하려면 최소 하나 이상의 매개변수를 제공해야 합니다.

예제

```
python CADSTool.py setMetadata --source "/Demo/Drafts/MyReport.rptdesign" --version "0:2006-08-25T21:15:49+01:00" --keyword "Quarterly" --customProperty "multi=hi|hello|bye;Complexity Degree=Simple" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- `<source>` 메타데이터가 설정되었습니다.
- `<source>` 해당 파일 또는 폴더가 없습니다. 다시 시도하십시오.
- `<source>` 메타데이터를 설정하는 중에 오류가 발생했습니다.

uploadFile 조작

파일의 새 버전을 작성하는 옵션을 사용하여(이미 존재하는 경우) 로컬 파일 시스템의 콘텐츠 리포지토리에 파일을 저장합니다.

구문

```
python CADSTool.py uploadFile --source "<source>" --target "<target>" --createVersion -z
```

여기서,

- <source>는 업로드할 파일의 완전한 경로(로컬 파일 시스템에서)입니다. 이는 필수 매개변수입니다.
- <target>은 파일이 업로드될 리포지토리에서 폴더의 완전한 경로입니다. 이는 필수 매개변수입니다.
- --createVersion은 지정된 파일이 이미 종료되었으며 새 버전을 작성해야 함을 표시합니다. 이는 선택적 매개변수입니다.

예제

다음 예제에서 대상은 *Drafts*에 대한 완전한 경로입니다.

```
python CADSTool.py uploadFile --source "C:\Demo\MyReport.rptdesign"
--target "/Demo/Drafts" -z
```

*MyReport.rptdesign*이 */Demo/Drafts* 폴더에 이미 존재하는 경우 다음과 같이 --createVersion 매개변수를 사용하십시오.

```
python CADSTool.py uploadFile --source "C:\Demo\MyReport.rptdesign"
--target "/Demo/Drafts" --createVersion -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source> 파일이 업로드되었습니다.
- <source> 해당 파일이 없습니다. 다시 시도하십시오.
- <target> 해당 폴더가 없습니다. 다시 시도하십시오.
- <source> 파일을 업로드하는 중에 오류가 발생했습니다.

프로세스 관리 기능

프로세스 관리 스크립팅은 작업할 수 있는 기능을 제공합니다. 이 영역에는 다음 기능이 포함되어 있습니다.

- 작업 실행
- 작업 히스토리 검색
- 작업 세부사항 검색

이 절에서는 프로세스 관리 기능에 대한 스크립트의 Python 명령행 사용법에 대해 간략하게 설명합니다. 모든 API에는 상세한 구문 정보, 예제 및 예상되는 메시지가 포함됩니다.

키워드

26 페이지의 표 5는 프로세스 관리 API에 대해 지원되는 키워드를 나열합니다. 두 번째 열은 제공된 키워드의 선택적 축약 버전을 나열합니다. 테이블은 프로세스 관리 API에 특정한 키워드만 나열합니다. 프로세스 관리 API 및 리포지토리 API 모두에 적용되는 추가 키워드에 대해서는 3 페이지의 표 2 및 5 페이지의 표 3의 내용을 참조하십시오.

표 5. 프로세스 관리 API에 대한 키워드

키워드	선택적 축약 버전	사용법
--source	-s	경로가 포함된 소스 작업
--target	-t	대상 폴더 경로
--notification	-j	알림과 함께 작업이 실행됨을 표시함
--async	-m	비동기식으로 작업이 실행됨을 표시함
--execId	-y	작업에 대한 실행 ID
--jobStepName	-q	작업 단계 이름
--log		로그가 삭제되어야 함을 표시함. --target과 함께 사용되는 경우 로그는 --target으로 지정된 위치에 저장됩니다. 그렇지 않으면 로그는 인라인으로 표시됩니다.

프로세스 관리 조작

deleteJobExecutions 조작

지정된 작업 실행 오브젝트를 삭제합니다.

구문

```
python CADSTool.py deleteJobExecutions --execIds "<execIDs>" -z
```

<execIDs> 값은 실행에서 삭제할 식별자의 공백으로 구분된 목록입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py deleteJobExecutions --execIds  
"0a58c33d002ce9080000 010e0ccf7b01800e" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- 실행 ID가 지정되지 않았습니다.

executeJob 조작

전달된 매개변수를 기반으로 동기식 또는 비동기식으로 작업을 실행합니다. 동기 실행의 경우 API는 작업이 완료될 때까지 리턴하지 않습니다. 비동기 실행의 경우 API는 작업이 시작된 후 리턴합니다.

구문

```
python CADSTool.py executeJob --source "<source>" --notification --async -z
```

여기서,

- <source>는 리포지토리에서 작업의 완전한 경로입니다. 이는 필수 매개변수입니다.
- --notification은 알림과 함께 작업을 실행하는 데 사용됩니다. 이는 선택적 매개변수입니다.
- --async는 비동기식으로 작업을 실행하는 데 사용됩니다. 이는 선택적 매개변수입니다.

예제

다음 예제에서는 알림 없이 작업을 동기식으로 실행합니다.

```
python CADSTool.py executeJob --source "/Demo/Jobs/Reports" -z
```

다음 예제에서는 알림과 함께 작업을 동기식으로 실행합니다.

```
python CADSTool.py executeJob --source "/Demo/Jobs/Reports" --notification -z
```

다음 예제에서는 알림 없이 작업을 비동기식으로 실행합니다.

```
python CADSTool.py executeJob --source "/Demo/Jobs/Reports" --async -z
```

다음 예제에서는 알림과 함께 작업을 비동기식으로 실행합니다.

```
python CADSTool.py executeJob --source "/Demo/Jobs/Reports" --async --notification -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- <source> 작업이 실행되었습니다. 작업 실행 ID는 <execId>입니다.
- <source> 해당 작업이 없습니다. 다시 시도하십시오.
- <source> 작업을 실행하는 중에 오류가 발생했습니다.

getJobExecutionDetails 조작

작업 단계 및 반복을 포함한 특정 작업에 대한 실행 세부사항을 나열합니다.

구문

```
python CADSTool.py getJobExecutionDetails --execId "<execID>" --log --target "<target>" -z
```

여기서,

- <execId>는 작업의 실행 식별자입니다. 이는 필수 매개변수입니다.
- --log는 작업 로그가 인라인으로 표시되어야 함을 표시합니다. --log 매개변수가 포함되지 않으면 작업 단계 실행으로 생성된 로그가 표시되지 않습니다. 이는 선택적 매개변수입니다.
- <target>은 로그를 저장할 위치(로컬 파일 시스템에서)입니다. 이는 선택적 매개변수이며, --log 매개변수와 함께만 사용됩니다.

예제

다음 예제에서는 특정 작업 실행에 대한 세부사항을 나열합니다.

```
python CADSTool.py getJobExecutionDetails --execId "0a58c3710016a7860000010d1a6a87b48400" -z
```

다음 예제에서는 인라인으로 표시되는 로그 및 특정 작업 실행에 대한 세부사항을 표시합니다.

```
python CADSTool.py getJobExecutionDetails --execId "0a58c3710016a7860000010d1a6a87b48400" --log -z
```

다음 예제에서는 특정 위치에 저장된 작업 로그 및 특정 작업 실행에 대한 세부사항을 나열합니다.

```
python CADSTool.py getJobExecutionDetails --execId "0a58c3710016a7860000010d1a6a87b48400" --log --target "c:\logs" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- 성공적인 실행을 위해 모든 실행 세부사항은 작업, 작업 단계 및 작업 반복에 대해 나열됩니다. 로그는 인라인으로 표시되고 로컬 파일 시스템의 지정된 위치에 저장됩니다.
- <execId> 해당 실행이 없습니다. 다시 시도하십시오.
- <execId> 작업 실행에 대한 세부사항을 표시하는 중에 오류가 발생했습니다.
- --log 매개변수 없이 --target을 사용할 수 없습니다.

getJobExecutionList 조작

모든 작업의 버전에서 특정 작업에 대한 현재 실행 및 완료된 실행을 나열합니다.

구문

```
python CADSTool.py getJobExecutionList --source "<source>" -z
```

<source> 값은 리포지토리에서 작업의 완전한 경로입니다. 이는 필수 매개변수입니다.

예제

```
python CADSTool.py getJobExecutionList --source "/Demo/Jobs/Reports" -z
```

메시지

다음 메시지는 이 API를 사용하는 경우 표시될 수 있습니다.

- 지정된 작업의 성공적인 실행을 위해 모든 실행 세부사항(예: 실행 ID, 작업 이름, 작업 실행 상태, 작업 실행 시작 및 종료 시작)이 나열됩니다.
- <source> 해당 작업이 없습니다. 다시 시도하십시오.
- <source> 작업에 대한 실행 목록을 표시하는 중에 오류가 발생했습니다.

제 3 장 PESImpl 모듈

IBM SPSS Collaboration and Deployment Services - Essentials for Python 기능은 IBM SPSS Collaboration and Deployment Services Repository 오브젝트와의 상호작용을 Python 스크립트에 직접 허용합니다.

Python 코드 내의 `pes.api.PESImpl` 모듈에서 `PESImpl` 클래스를 가져오십시오. 연결할 리포지토리에 대한 연결 정보를 사용하는 `PESImpl` 오브젝트를 작성하십시오.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("user", "password", "host", "port", ssl=True)
```

`PESImpl` 생성자의 매개변수는 다음과 같습니다.

- `user`는 사용자의 이름에 해당합니다. 사용자가 네이티브 제공자에 있지 않은 경우 값에는 보안 제공자를 표시하는 접두부가 포함되어야 합니다. 유효한 접두부 값은 다음과 같습니다.
 - 시스템에 고유한 네이티브 로컬 보안 제공자의 네이티브. 이는 기본 제공자입니다.
 - Active Directory의 `AD_<name>`. 여기서, `<name>`은 시스템 내의 보안 제공자 이름에 해당합니다.
 - 로컬 대체가 사용된 Active Directory의 `ADL_<name>`. 여기서, `<name>`은 시스템 내의 보안 제공자 이름에 해당합니다.
 - OpenLDA의 `ldap_<name>`. 여기서, `<name>`은 시스템 내의 보안 제공자 이름에 해당합니다.접두사로 슬래시를 사용하여 사용자 이름이 표시됩니다. Active Directory 제공자의 경우에는 접두부에 도메인을 포함하십시오. 예를 들어, `AD_SPSSAD` Active Directory 인스턴스의 `MYDOMAIN` 도메인에 있는 `icrod` 사용자의 경우 값은 `AD_SPSSAD/MYDOMAIN/icrod`입니다. `icrod` 사용자가 `SPSSLDAP` OpenLDAP 제공자에 있는 경우 값은 `ldap_SPSSLDAP/icrod`입니다.
- `password`는 지정된 사용자와 연관된 비밀번호에 해당합니다.
- `host`는 리포지토리 서버의 이름을 지정합니다.
- `port`는 리포지토리 서버의 포트 번호를 지정합니다.
- `ssl=True`는 SSL(Secure Sockets Layer) 프로토콜을 사용하여 통신을 암호화함을 표시합니다. `ssl` 매개변수가 `False`로 설정되거나 매개변수가 `PESImpl` 오브젝트를 작성할 때 생략된 경우 서버 통신은 SSL을 사용하지 않습니다. SSL을 사용하는 경우 리포지토리 서버가 SSL에 대해 구성되어야 합니다. 자세한 정보는 관리자 문서를 참조하십시오.

또는 `host`, `port` 및 `ssl` 매개변수 대신 서버 URL을 지정할 수 있습니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("user", "password", server_url="url")
```

`server_url` 매개변수는 리포지토리 서버의 전체 URL을 지정합니다. 서버가 사용자 정의 컨텍스트 경로를 사용한 경우 이 매개변수를 사용하십시오. 예를 들어, 다음 생성자는 포트 443에서 SSL을 사용하고 `/ibm/spss`의 컨텍스트 경로가 있는 `myserver` 서버에 해당합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("myUser", "myPass", server_url="https://myserver:443/ibm/spss")
```

참고: IPv6 주소는 대괄호로 묶어야 합니다(예: `[3ffe:2a00:100:7031::1]`).

특정 메소드는 `pesImpl` 오브젝트를 사용하여 액세스될 수 있습니다.

컨텐츠 리포지토리 API

컨텐츠 리포지토리 스크립팅은 파일 및 폴더와 같은 리포지토리 자원에 대해 작업할 수 있는 기능을 제공합니다. 이 영역에는 다음 기능이 포함되어 있습니다.

- 폴더 작성 및 삭제
- 파일 업로드 및 다운로드
- 폴더 내보내기 및 가져오기
- 레이블, 보안 및 메타데이터 관리

이 절에서는 리포지토리에 저장된 자원에 대해 작업하는 데 사용되는 `PESImpl` API에 대해 간략하게 설명합니다. 모든 메소드에는 상세한 구문 정보, 예제 및 예상되는 메시지가 포함됩니다.

메소드

다음 절에서는 IBM SPSS Collaboration and Deployment Services에 대해 지원되는 모든 컨텐츠 리포지토리 메소드를 나열합니다.

참고:

- 선택적 매개변수 `Label` 및 `Version`이 사용되는 모든 메소드의 경우 `Label` 또는 `Version`(둘 다는 아님) 중 하나를 사용하십시오. `Version` 또는 `Label`이 제공된 파일 또는 폴더에 지정되지 않는 경우 최신 버전이 사용됩니다.
- 리포지토리의 파일 또는 폴더에 경로가 필요한 모든 메소드의 경우 경로 또는 오브젝트 URI를 사용할 수 있습니다. IBM SPSS Deployment Manager에서 오브젝트 특성 보기를 통해 오브젝트 URI를 확인할 수 있습니다.
- 비라틴 유니코드 문자가 포함된 소스 또는 대상 리포지토리 또는 파일 시스템 경로의 입력이 필요한 메소드의 경우 문자열은 유니코드 오브젝트로 지정되어야 합니다. 예를 들면, 다음과 같습니다.

```
identificationSpecifier = pesImpl.uploadFile
(source=u'C:\Analytics\La Peña.txt',
target=u'/La Peña')
```

advanceSearch 메소드

입력으로 전달된 여러 매개변수를 기반으로 리포지토리에서 파일 및 폴더를 검색합니다.

다음 항목을 검색할 수 있습니다.

- 작성자
- 설명
- 제목
- 생성자
- 수정자
- 만기 시작 날짜
- 만기 종료 날짜
- MIME 유형
- 레이블
- 키워드
- 토픽
- 작성된 시작 날짜
- 작성된 종료 날짜
- 버전이 수정된 시작 날짜
- 오브젝트가 수정된 종료 날짜
- 오브젝트가 수정된 시작 날짜
- 버전이 수정된 시작 날짜
- 버전이 수정된 종료 날짜
- 상위 폴더 URI
- 자원 URI

`advanceSearch(criteriaDict,submittedHierarchy)`

표 6. *advanceSearch*에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<i>criteriaDict</i>	필수	사전	수행될 검색에 대한 키/값 쌍이 사전에 포함됩니다. 허용 가능한 키 값은 다음과 같습니다. <ul style="list-style-type: none"> • author • title • description • createdBy • modifiedBy • expirationStartDate • expirationEndDate • mimeType • label • keyword • topic • createdStartDate • createdEndDate • objectModifiedStartDate • objectModifiedEndDate • versionModifiedStartDate • versionModifiedEndDate • parentURI • uri 	{ "author": "admin", "title": "search", "label": "label 1"}
<i>submittedHierarchy</i>	선택사항	부울	<i>Submitted Jobs</i> 폴더의 검색 여부를 표시함	True 또는 False

다른 검색 필드(예: 제목 또는 작성자)와 함께 사용될 때 현재 *expirationStartDate* 및 *expirationEndDate*가 실행되지 않습니다.

표 7. *advanceSearch*에 대한 리턴 값.

유형	설명
PageResult	각 행이 검색 일치에 해당하는 구조. 자세한 정보는 57 페이지의 『PageResult 클래스』 주제를 참조하십시오.

표 8. *advanceSearch*에 대한 예외.

유형	설명
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예: 레이블 및 키워드로 검색

다음 샘플에서는 *Quarterly*의 키워드 값이 있는 *Production*으로 레이블된 모든 파일 버전을 리턴합니다.

```

from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
critDict = {'label': 'Production', 'keyword': 'Quarterly'}
sResults = pesImpl.advanceSearch(critDict)
sRows = sResults.getRows()
for sRow in sRows:
    print "Author: ", sRow.getAuthor()
    print "Title: ", sRow.getTitle()
    for child in sRow.getChildRow():
        print "Version: ", child.getVersionMarker()
        print "Label: ", child.getVersionLabel()
        print "Keywords:", child.getKeyword()
        print "URI:", child.getUri()

```

예: URI 로 검색

다음 샘플에서는 지정된 URI로 파일의 모든 파일 버전을 리턴합니다.

```

from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
critDict = {'uri': 'spsscr:///id=a010a37ba5992bb00000127b0f952f945be'}
sResults = pesImpl.advanceSearch(critDict)
sRows = sResults.getRows()
for sRow in sRows:
    print "Author: ", sRow.getAuthor()
    print "Title: ", sRow.getTitle()
    for child in sRow.getChildRow():
        print "Version: ", child.getVersionMarker()
        print "Label: ", child.getVersionLabel()
        print "Keywords:", child.getKeyword()
        print "URI:", child.getUri()

```

applySecurity 메소드

리포지토리에서 파일 또는 폴더에 대한 보안 액세스 제어 목록(ACL)을 설정합니다.

`applySecurity(source, principal, permission, provider, cascade)`

표 9. `applySecurity`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<i>source</i>	필수	문자열	리포지토리에서 파일 또는 폴더의 완전한 경로 또는 오브젝트 URI	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a7860000 010dcee0eaa28219"
<i>principal</i>	필수	문자열	ACL의 일부로 지정된 파일 또는 폴더에 적용할 사용자(예: <i>admin</i>)	admin
<i>permission</i>	필수	문자열	지정된 파일 또는 폴더에 적용할 권한의 유형	READ, WRITE, DELETE, MODIFY_ACL, OR OWNER

표 9. `applySecurity`에 대한 입력 매개변수 (계속).

필드	사용	유형	설명	예제 값
<code>provider</code>	선택사항	문자열	<p>사용자(예: <code>Native</code>)에게 보안을 적용하는 데 사용하는 보안 제공자. 유효한 값은 다음과 같습니다.</p> <ul style="list-style-type: none"> • 시스템에 고유한 네이티브 로컬 보안 제공자의 네이티브. 이는 기본 제공자입니다. • Active Directory의 <code>AD_<name></code>. 여기서, <code><name></code>은 시스템 내의 보안 제공자 이름에 해당합니다. • 로컬 대체가 사용된 Active Directory의 <code>ADL_<name></code>. 여기서, <code><name></code>은 시스템 내의 보안 제공자 이름에 해당합니다. • OpenLDA의 <code>ldap_<name></code>. 여기서, <code><name></code>은 시스템 내의 보안 제공자 이름에 해당합니다. 	<code>Native</code>
<code>cascade</code>	선택사항	부울	지정된 폴더 내의 모든 파일 및 하위 폴더에 보안 설정을 전파함	<code>True</code> 또는 <code>False</code>

표 10. `applySecurity`에 대한 리턴 값.

유형	설명
Boolean	<code>True</code> 또는 <code>False</code> (메소드의 성공적인 실행 여부를 기반으로 함)

표 11. `applySecurity`에 대한 예외.

유형	설명
<code>ResourceNotFoundException</code>	소스 파일이 없습니다.
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.
<code>IllegalParameterException</code>	지정된 사용자 또는 보안 제공자 이름이 올바르지 않습니다.

예제

다음 샘플에서는 지정된 파일에 대한 `READ` 권한을 사용자에게 지정합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
bSuccess = pesImpl.applySecurity(source="/Projects",principal="icrod",permission="READ",
    provider="Native")
```

`cascadeSecurity` 메소드

폴더 내의 모든 파일 및 하위 폴더에 폴더의 보안 설정을 전파합니다.

```
cascadeSecurity(source)
```

표 12. *cascadeSecurity*에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<i>source</i>	필수	문자열	리포지토리에서 폴더의 완전한 경로 또는 오브젝트 URI	"/Temp Folder" 또는 "0a58c3670016a7860000 010dcee0eaa28219"

표 13. *cascadeSecurity*에 대한 리턴 값.

유형	설명
Boolean	True 또는 False(메소드의 성공적인 실행 여부를 기반으로 함)

표 14. *cascadeSecurity*에 대한 예외.

유형	설명
ResourceNotFoundException	소스 폴더가 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 샘플에서는 *Projects* 폴더에 대한 보안을 폴더의 콘텐츠에 단계적으로 적용합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "sps", "localhost", "8080")
bSuccess = pesImpl.cascadeSecurity(source="/Projects")
```

copyResource 메소드

리포지토리에서 파일 또는 폴더를 다른 폴더에 복사합니다. 지정된 소스 파일 또는 폴더가 복사될 때 이름을 변경할 수 있습니다.

이름 변경에 대한 자세한 정보는 48 페이지의 『*moveResource* 메소드』의 내용을 참조하십시오.

```
copyResource(source, target)
```

표 15. *copyResource*에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<i>source</i>	필수	문자열	리포지토리에서 파일 또는 폴더의 완전한 경로 또는 오브젝트 URI	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a7860000 010dcee0eaa28219"
<i>target</i>	필수	문자열	파일을 복사할 폴더의 완전한 경로 또는 오브젝트 URI. 지정된 파일 또는 폴더가 복사될 때 이름을 변경하기 위해 새 파일 이름도 제공될 수 있습니다.	"/New Folder" 또는 "/New Folder/abc.dat"

표 16. `copyResource`에 대한 리턴 값.

유형	설명
String	복사된 파일 또는 폴더의 URI

표 17. `copyResource`에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일 또는 대상 폴더가 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 샘플에서는 `Drafts` 폴더를 `Projects`라는 폴더로 복사합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
uri = pesImpl.copyResource(source="/Demo/Drafts/MyReport.rptdesign",target="/Projects")
print uri
```

createFolder 메소드

리포지토리의 지정된 위치에 새 폴더를 작성합니다.

```
createFolder(source)
```

표 18. `createFolder`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에 작성할 폴더	/Temp Folder/Sample Folder

표 19. `createFolder`에 대한 리턴 값.

유형	설명
String	작성된 폴더의 URI

표 20. `createFolder`에 대한 예외.

유형	설명
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.
ResourceAlreadyExistsException	지정된 폴더가 이미 리포지토리에 있습니다.

예제

다음 예제에서는 `Demo` 폴더의 하위로 `Drafts`라는 폴더를 작성합니다. 폴더를 작성하는 중에 문제점이 발생하면 예외 메시지가 콘솔에 전송됩니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
try:
    uri = pesImpl.createFolder(source="/Demo/Drafts")
    print "URI for the folder is:", uri
except:
    print "Unhandled exception in createFolder."
```

deleteFile 메소드

리포지토리에서 파일을 삭제합니다. 파일의 모든 버전이 삭제됩니다.

```
deleteFile(source,submittedHierarchy)
```

표 21. deleteFile에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<i>source</i>	필수	문자열	리포지토리에서 파일의 완전한 경로 또는 오브젝트 URI	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a7860000010dcee0eaa28219"
<i>submittedHierarchy</i>	선택사항	부울	파일이 <i>Submitted Jobs</i> 폴더에 있는지를 표시함	True 또는 False

표 22. deleteFile에 대한 리턴 값.

유형	설명
Boolean	True 또는 False(메소드의 성공적인 실행 여부를 기반으로 함)

표 23. deleteFile에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일이 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.
IllegalParameterException	삭제하기 위해 지정된 자원은 폴더입니다.

예제

다음 예제에서는 리포지토리에서 *MyReport.rptdesign* 파일을 삭제합니다.

```
from pes.util.PESExceptions import *
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
try:
    bSuccess = pesImpl.deleteFile(source="/Demo/Drafts/MyReport.rptdesign")
except ResourceNotFoundException:
    print "Specified file does not exist."
except InsufficientParameterException:
    print "No file specified."
except IllegalParameterException:
    print "Item to be deleted is not a file."
```

deleteFileVersion 메소드

리포지토리에서 파일의 특정 버전을 삭제합니다.

```
deleteFileVersion(source,version,label,submittedHierarchy)
```

표 24. `deleteFileVersion`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 파일의 완전한 경로 또는 오브젝트 URI	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a7860000010dcee0eaa28219"
<code>version</code>	선택사항. 그러나 <code>version</code> 또는 <code>label</code> 중 하나를 지정해야 합니다.	문자열	삭제할 파일의 특정 버전	"0:2006-08-25 21:15:49.453"
<code>label</code>	선택사항. 그러나 <code>version</code> 또는 <code>label</code> 중 하나를 지정해야 합니다.	문자열	삭제할 파일의 레이블된 특정 버전	"Version 1"
<code>submittedHierarchy</code>	선택사항	부울	파일이 <code>Submitted Jobs</code> 폴더에 있는지를 표시함	True 또는 False

표 25. `deleteFileVersion`에 대한 리턴 값.

유형	설명
Boolean	True 또는 False(메소드의 성공적인 실행 여부를 기반으로 함)

표 26. `deleteFileVersion`에 대한 예외.

유형	설명
<code>ResourceNotFoundException</code>	소스 파일 또는 대상 폴더가 없습니다.
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.
<code>IllegalParameterException</code>	삭제하기 위해 지정된 자원은 폴더입니다.

예제

다음 예제에서는 리포지토리에서 `Test`로 레이블된 `MyReport.rptdesign` 파일의 버전을 삭제합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
bSuccess = pesImpl.deleteFileVersion(source="/Demo/Drafts/MyReport.rptdesign",label="Test")
```

deleteFolder 메소드

리포지토리에서 폴더 및 해당 콘텐츠를 삭제합니다.

```
deleteFolder(source,submittedHierarchy)
```

표 27. `deleteFolder`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 폴더의 완전한 경로 또는 오브젝트 URI	"/Temp Folder" 또는 "0a58c3670016a78600 00010dcee0eaa28219"
<code>submittedHierarchy</code>	선택사항	부울	폴더가 <code>Submitted Jobs</code> 폴더에 있는지를 표시함	True 또는 False

표 28. `deleteFolder`에 대한 리턴 값.

유형	설명
Boolean	True 또는 False(메소드의 성공적인 실행 여부를 기반으로 함)

표 29. `deleteFolder`에 대한 예외.

유형	설명
<code>ResourceNotFoundException</code>	지정된 폴더가 없습니다.
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.
<code>IllegalParameterException</code>	삭제하기 위해 지정된 자원은 폴더가 아닙니다.

예제

다음 예제에서는 리포지토리에서 `Drafts`라는 폴더를 삭제합니다. 폴더를 삭제하는 중에 문제점이 발생하면 예외 메시지가 콘솔에 전송됩니다.

```
from pes.util.PESExceptions import *
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
try:
    bSuccess = pesImpl.deleteFolder(source="/Demo/Drafts")
except ResourceNotFoundException:
    print "Specified folder does not exist."
except InsufficientParameterException:
    print "No folder specified."
except IllegalParameterException:
    print "Item to be deleted is not a folder."
```

downloadFile 메소드

파일의 특정 버전을 리포지토리에서 로컬 파일 시스템으로 다운로드합니다.

```
downloadFile(source, target, version, label)
```

표 30. `downloadFile`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	다운로드할 파일의 완전한 리포지토리 경로	"/Temp Folder/Temp.txt"
<code>target</code>	필수	문자열	파일을 다운로드할 폴더의 완전한 경로 (로컬 파일 시스템에서)	"C:\Temp"

표 30. `downloadFile`에 대한 입력 매개변수 (계속).

필드	사용	유형	설명	예제 값
<code>version</code>	선택 사항. <code>version</code> 또는 <code>label</code> 중 하나를 지정할 수 있습니다.	문자열	다운로드할 파일의 특정 버전	"0:2006-08-25 21:15:49.453"
<code>label</code>	선택 사항. <code>version</code> 또는 <code>label</code> 중 하나를 지정할 수 있습니다.	문자열	다운로드할 파일의 레이블된 특정 버전	"Version 2"

표 31. `downloadFile`에 대한 리턴 값.

유형	설명
Resource	리포지토리 오브젝트에 대한 정보의 컨테이너. 자세한 정보는 56 페이지의 『Resource 클래스』 주제를 참조하십시오. .

표 32. `downloadFile`에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일 또는 대상 폴더가 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 샘플에서는 `MyReport.rptdesign` 파일의 `Production`으로 레이블된 버전을 로컬 파일 시스템의 `Shared` 디렉토리에 다운로드합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
resource = pesImpl.downloadFile(source="/Demo/Drafts/MyReport.rptdesign",
    target="c:/Demo/Shared",label="Production")
```

exportResource 메소드

지정된 리포지토리 폴더를 로컬 파일 시스템의 지정된 *.pes 내보내기 파일에 내보냅니다.

```
exportResource(source, target)
```

표 33. `exportResource`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	내보낼 폴더의 완전한 리포지토리 경로 또는 오브젝트 URI	/Temp Folder 또는 0a58c3670016a78 60000010dcee0eaa28219
<code>target</code>	필수	문자열	폴더를 내보낼 완전한 경로(로컬 파일 시스템에서) 및 파일 이름	C:\Temp\backup.pes

표 34. `exportResource`에 대한 리턴 값.

유형	설명
Boolean	True 또는 False(메소드의 성공적인 실행 여부를 기반으로 함)

표 35. `exportResource`에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일 또는 대상 폴더가 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.
IllegalParameterException	지정된 대상은 폴더입니다. 대상은 *.pes 파일이어야 합니다.

예제

다음 샘플에서는 `Drafts` 폴더의 콘텐츠를 로컬 파일 시스템의 `backups` 폴더에 있는 내보내기 파일에 내보냅니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
bSuccess = pesImpl.exportResource(source="/Projects",target="C:\Demo\drafts.pes")
```

getAccessControlList 메소드

리포지토리에서 지정된 파일 또는 폴더에 대한 보안 액세스 제어 목록(ACL)을 검색합니다.

```
getAccessControlList(source,submittedHierarchy)
```

표 36. `getAccessControlList`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 파일 또는 폴더의 완전한 경로 또는 오브젝트 URI	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a786000010dcee0eaa28219"
<code>submittedHierarchy</code>	선택사항	부울	파일 또는 폴더가 <code>Submitted Jobs</code> 폴더에 있는지를 표시함	True 또는 False

표 37. `getAccessControlList`에 대한 리턴 값.

유형	설명
Dictionary	사용자 이름 및 연관된 권한이 포함된 사전이 표시됩니다. 예를 들면, 다음과 같습니다. { "admin": "MODIFY_ACL", "Joe": "DELETE" }

표 38. `getAccessControlList`에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일 또는 대상 폴더가 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 예제에서는 `MyReport.rptdesign` 파일에 대한 ACL을 인쇄합니다.

```

from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
aclDic = pesImpl.getAccessControlList(source = "/Projects/MyReport.rptdesign")
print aclDic

```

getAllVersions 메소드

리포지토리에서 파일의 모든 버전 목록을 검색합니다.

```
getAllVersions(source,submittedHierarchy)
```

표 39. *getAllVersions*에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<i>source</i>	필수	문자열	리포지토리에서 파일의 완전한 경로 또는 오브젝트 URI	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a78600 00010dcee0eaa28219"
<i>submittedHierarchy</i>	선택사항	부울	파일이 <i>Submitted Jobs</i> 폴더에 있는지를 표시함	True 또는 False

표 40. *getAllVersions*에 대한 리턴 값.

유형	설명
List	자원 오브젝트의 목록입니다. 56 페이지의 『Resource 클래스』의 내용을 참조하십시오.

표 41. *getAllVersions*에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일이 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.
IllegalParameterException	지정된 소스가 폴더입니다.

예제

이 예제에서는 *MyReport.rptdesign* 파일의 모든 버전, 각 작성자, 버전 마커 및 버전 레이블의 인쇄에 대한 정보를 검색합니다.

```

from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
resourceList = pesImpl.getAllVersions(source="/Demo/Drafts/MyReport.rptdesign")
for resource in resourceList:
    print resource.getAuthor()
    print resource.getVersionMarker()
    print resource.getVersionLabel()

```

getChildren 메소드

지정된 리포지토리 폴더 내에서 모든 파일 및 폴더의 목록을 검색합니다.

```
getChildren(source,submittedHierarchy)
```

표 42. `getChildren`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 폴더의 완전한 경로 또는 오브젝트 URI	"/Temp Folder" 또는 "0a58c3670016a78600 00010dcee0eaa28219"
<code>submittedHierarchy</code>	선택사항	부울	폴더가 Submitted Jobs 폴더에 있는지를 표시함	True 또는 False

표 43. `getChildren`에 대한 리턴 값.

유형	설명
List	자원 오브젝트의 목록입니다. 56 페이지의 『Resource 클래스』의 내용을 참조하십시오.

표 44. `getChildren`에 대한 예외.

유형	설명
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.
<code>ResourceNotFoundException</code>	폴더가 없습니다.

예제

다음 샘플에서는 `/Demo/Drafts`폴더의 콘텐츠, 각 제목, 작성자 및 자원 식별자 인수를 검색합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
resourceList = pesImpl.getChildren(source="/Demo/Drafts")
for resource in resourceList:
    print "Resource title:", resource.getTitle()
    print "Resource author:", resource.getAuthor()
    print "Resource ID:", resource.getResourceID()
```

getCustomPropertyValue 메소드

지정된 사용자 정의 특성으로 허용된 유효한 값을 검색합니다.

```
getCustomPropertyValue(propertyName)
```

표 45. `getCustomPropertyValue`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>propertyName</code>	필수	문자열	사용자 정의 특성의 이름	"FreeForm"

표 46. `getCustomPropertyValue`에 대한 리턴 값.

유형	설명
List	사용자 정의 특성이 허용하는 유효한 값의 목록을 리턴합니다. 특성에 선택사항(예: 단일 선택 또는 다중 선택)이 필요한 경우 목록에는 선택사항에 대한 모든 유효한 값이 포함됩니다. 자유 형식의 특성인 경우 목록에는 특성이 허용하는 데이터의 유형이 포함됩니다(예: String, Date 또는 Number).

표 47. `getCustomPropertyValue`에 대한 예외.

유형	설명
<code>ResourceNotFoundException</code>	지정된 특성이 없습니다.
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 샘플에서는 사용자 정의 특성인 `Language`에 대한 값에 액세스합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
langList = pesImpl.getCustomPropertyValue(propertyName = "Language")
print langList
```

getMetadata 메소드

리포지토리에서 사용자 정의 특성 및 토픽 정보를 포함한 파일 또는 폴더의 메타데이터 속성을 검색합니다.

```
getMetadata(source, version, label, submittedHierarchy)
```

표 48. `getMetadata`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 파일 또는 폴더의 완전한 경로 또는 오브젝트 URI	<code>"/Temp Folder/Temp.txt"</code> 또는 <code>"0a58c3670016a7860000010dcee0eaa28219"</code>
<code>version</code>	선택사항. version 또는 label 중 하나를 지정할 수 있습니다.	문자열	파일 또는 폴더의 특정 버전	<code>"0:2006-08-25 21:15:49.453"</code>
<code>label</code>	선택사항. version 또는 label 중 하나를 지정할 수 있습니다.	문자열	파일 또는 폴더의 레이블된 특정 버전	<code>"Version 1"</code>
<code>submittedHierarchy</code>	선택사항	부울	파일이 <code>Submitted Jobs</code> 폴더에 있는지를 표시함	<code>True</code> 또는 <code>False</code>

표 49. `getMetadata`에 대한 리턴 값.

유형	설명
<code>Resource</code>	리포지토리 오브젝트에 대한 정보의 컨테이너. 자세한 정보는 56 페이지의 『Resource 클래스』 주제를 참조하십시오.

표 50. `getMetadata`에 대한 예외.

유형	설명
<code>ResourceNotFoundException</code>	소스 파일 또는 폴더가 없습니다.

표 50. `getMetadata`에 대한 예외 (계속).

유형	설명
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 예제에서는 `/Demo/Drafts` 폴더의 자원 식별자에 액세스합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
resource = pesImpl.getMetadata(source="/Demo/Drafts")
resourceid = resource.getResourceID()
```

importResource 메소드

로컬 파일 시스템에서 리포지토리로 기존 `*.pes` 내보내기 파일을 가져옵니다.

```
importResource(source, target, resourceType, resourceConflict, invalidVersionConflict,
resourceDef, labelFrom, lockResolution, exclude)
```

표 51. `importResource`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	가져올 파일의 완전한 경로(로컬 파일 시스템에서)	"C:\Temp\New.pes"
<code>target</code>	필수	문자열	가져올 폴더의 완전한 리포지토리 경로 또는 오브젝트 URI	"/Temp Folder" 또는 "0a58c3670016a7860000 010dcee0eaa28219"
<code>resourceType</code>	선택사항	문자열	가져오는 콘텐츠의 유형입니다. 다음 값 중 하나를 지정하십시오. <ul style="list-style-type: none"> ContentRepository - 파일 및 폴더와 같은 콘텐츠 오브젝트 ResourceDef - 자원 정의 가져오기 Credential - 사용자 신임 정보 DataSource - 데이터 소스 정의 MessageDomain - 메시지 도메인 ServerCluster - 서버 클러스터 정의 Server - 서버 정의 PromotionPolicy - 프로모션 정책 이 매개변수가 지정되지 않은 경우 ContentRepository 의 기본값이 사용됩니다.	

표 51. `importResource`에 대한 입력 매개변수 (계속).

필드	사용	유형	설명	예제 값
<code>resourceConflict</code>	선택사항	문자열	<p>중복 ID 또는 이름 충돌을 해결하는 방법을 표시합니다. 다음 값 중 하나를 지정하십시오.</p> <ul style="list-style-type: none"> keepTarget. 대상 항목이 유지됩니다. <code>.pes</code> 파일에 포함된 중복 ID가 있는 소스 항목이 무시됩니다. addNewVersion. 일반적으로 이 옵션은 ID 충돌 또는 이름 충돌을 해결하는 데 사용됩니다. 소스 오브젝트 및 대상 오브젝트 간에 중복 ID 충돌이 발생하는 경우 오브젝트의 새 버전이 대상 위치에 작성됩니다. 이름 충돌이 발생하면 가져온 오브젝트의 이름이 대상 위치에서 변경됩니다. 일반적으로 이름이 변경된 오브젝트에 <code>_1</code>, <code>_2</code> 등이 추가됩니다. 오브젝트의 두 버전에 동일한 레이블이 있는 경우, 동일한 항목의 두 버전에 동일한 레이블이 있을 수 없으므로 시스템에서 하나의 레이블을 유지하고 중복 레이블을 삭제합니다. 보유한 레이블은 labelFrom 매개변수에 종속됩니다. <p>이 매개변수가 지정되지 않은 경우 keepTarget의 기본값이 사용됩니다.</p>	
<code>labelFrom</code>	선택사항	문자열	<p>오브젝트의 두 버전에 동일한 레이블이 있는 경우 사용할 레이블입니다. 다른 버전의 레이블이 삭제됩니다. source 또는 target을 지정하십시오. 이 매개변수가 지정되지 않은 경우 source의 기본값이 사용됩니다.</p>	
<code>lockResolution</code>	선택사항	문자열	<p>잠긴 자원이 발생하는 경우 진행 방법을 정의합니다. 다음 값 중 하나를 지정하십시오.</p> <ul style="list-style-type: none"> continue. 잠긴 자원을 생략하고 가져오기를 계속합니다. abort. 잠긴 자원이 발생하는 경우 가져오기 프로세스를 종료합니다. 오브젝트 잠금으로 인해 충돌이 발생하면 가져오기 프로세스가 종료되고 실패합니다. <p>이 매개변수가 지정되지 않은 경우 abort의 기본값이 사용됩니다.</p>	

표 51. `importResource`에 대한 입력 매개변수 (계속).

필드	사용	유형	설명	예제 값
<code>invalidVersionConflict</code>	선택사항	문자열	<p>가져오기 프로세스 중에 올바르지 않은 버전이 발생하는 경우 진행 방법을 정의합니다. 다음 값 중 하나를 지정하십시오.</p> <ul style="list-style-type: none"> • import. 올바르지 않은 버전을 가져옵니다. • discard. 올바르지 않은 버전을 삭제합니다. <p>이 매개변수가 지정되지 않은 경우 import의 기본값이 사용됩니다.</p>	
<code>resourceDef</code>	선택사항	문자열	<p>자원 정의에 대한 처리 동작을 정의합니다. 다음 값 중 하나를 지정하십시오.</p> <ul style="list-style-type: none"> • recommended. 식별자 또는 이름이 대상 정의와 충돌하지 않는 경우에만 자원 정의를 가져옵니다. 충돌이 있으면 자원 정의를 가져오지 않습니다. • include. 파일 가져오기에서 모든 자원 정의를 가져옵니다. 해당 선택란을 선택하여 가져오기에서 제외할 하나 이상의 자원 정의 유형을 선택할 수 있습니다. • exclude. 파일 가져오기에서 자원 정의를 가져오지 않습니다. 사용 가능한 자원 정의를 참조하려면 가져온 오브젝트를 수정해야 합니다. <p>이 매개변수가 지정되지 않은 경우 recommended의 기본값이 사용됩니다.</p>	

표 51. `importResource`에 대한 입력 매개변수 (계속).

필드	사용	유형	설명	예제 값
<code>exclude</code>	선택사항	문자열	<p>가져오기 중에 제외되는 자원 유형을 정의합니다. 다중 값은 세미콜론으로 구분된 목록으로서 임의의 순서로 조합될 수 있습니다. 다음 값 중 하나 이상을 지정하십시오.</p> <ul style="list-style-type: none"> credential - 사용자 신임 정보 제외 customproperty - 자원 오브젝트에 대한 사용자 정의 특성 제외 datasource - 데이터 소스 정의 제외 messagedomain - 메시지 도메인 제외 notification - 알림 정의 제외 servercluster - 서버 클러스터 정의 제외 server - 서버 정의 제외 topic - 토픽 정의 제외 <p>이 매개변수가 지정되지 않은 경우 모든 유형이 가져오기에 포함됩니다.</p>	

표 52. `importResource`에 대한 리턴 값.

유형	설명
Boolean	True 또는 False(메소드의 성공적인 실행 여부를 기반으로 함)

표 53. `importResource`에 대한 예외.

유형	설명
<code>ResourceNotFoundException</code>	소스 파일 또는 대상 폴더가 없습니다.
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 샘플에서는 `drafts.pes` 내보내기 파일의 콘텐츠를 IBM SPSS Collaboration and Deployment Services Repository의 `/Demo/Drafts`로 가져옵니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
bSuccess = pesImpl.importResource(source="C:\Demo\drafts.pes",target="/Demo/Drafts")
```

moveResource 메소드

리포지토리에서 파일 또는 폴더를 다른 폴더로 이동합니다. 지정된 소스 파일이 이동될 때 대상 유형 및 존재(최종 이름을 결정함)를 사용하여 지정된 소스 파일의 이름을 변경할 수 있습니다.

다음 표에서는 파일을 이동할 때 이름 변경 기능의 동작에 대해 설명합니다.

표 54. 파일 이름 변경.

대상 유형	대상 폴더가 존재함	대상 폴더가 존재하지 않음
폴더	소스 파일이 대상 폴더의 하위가 됩니다.	소스 파일이 지정된 대상의 상위 폴더로 이동하고 대상 폴더 이름으로 이름이 변경됩니다.
파일	소스 파일이 대상 파일을 포함한 폴더로 이동하고 대상 이름으로 이름이 변경됩니다.	보고된 오류입니다.

예를 들어, 소스가 `/Temp Folder/Temp.txt` 파일이고 지정된 대상이 `/Demo Folder` 폴더인 경우 다음 결과가 발생할 수 있습니다.

- `Demo Folder` 폴더가 있으면 `Temp.txt`가 `Demo Folder`로 이동됩니다.
- `Demo Folder` 폴더가 없으면 `Temp.txt`가 『/』로 이동하고 `Demo Folder`로 이름이 변경됩니다.

또는, 소스가 `/Temp Folder/Temp.txt`이고 지정된 대상이 `/Demo Folder/Abc.dat` 파일이면 다음 결과가 발생할 수 있습니다.

- `Demo Folder` 폴더가 있으면 `Temp.txt`가 `Demo Folder`로 이동하고 `Abc.dat`로 이름이 변경됩니다.
- `Demo Folder` 폴더가 없으면 오류가 표시됩니다.

`moveResource(source, target)`

표 55. `moveResource`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 파일 또는 폴더의 완전한 경로 또는 오브젝트 URI	<code>/Temp Folder/Temp.txt</code> 또는 <code>0a58c3670016a7860000010dcee0eaa28219</code>
<code>target</code>	필수	문자열	파일을 이동할 폴더의 완전한 경로 또는 오브젝트 URI. 지정된 파일 또는 폴더가 이동될 때 이름을 변경하기 위해 새 파일 이름도 제공될 수 있습니다.	<code>/New Folder</code> 또는 <code>/New Folder/abc.dat</code>

표 56. `moveResource`에 대한 리턴 값.

유형	설명
Boolean	True 또는 False(메소드의 성공적인 실행 여부를 기반으로 함)

표 57. `moveResource`에 대한 예외.

유형	설명
<code>ResourceNotFoundException</code>	지정된 소스가 없습니다.
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 샘플에서는 /Demo/Drafts 폴더에서 /Approved 폴더로 MyReport.rptdesign 파일을 이동합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
bSuccess = pesImpl.moveResource(source="/Demo/Drafts/MyReport.rptdesign",target="/Approved")
print bSuccess
```

removeLabel 메소드

리포지토리의 파일에서 레이블을 제거합니다.

```
removeLabel(source,label)
```

표 58. removeLabel에 대한 입력 매개변수.

필드	사용	유형	예제 값	설명
source	필수	문자열	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a7860000 010dcee0eaa28219"	리포지토리에서 파일의 완전한 경로 또는 오브젝트 URI
label	필수	문자열	"Version 1"	제거할 레이블 이름

표 59. removeLabel에 대한 리턴 값.

유형	설명
String	업데이트된 파일의 URI

표 60. removeLabel에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일이 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 샘플에서는 MyReport.rptdesign 파일에서 Draft 레이블을 제거합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
uri = pesImpl.removeLabel(source="/Demo/Drafts/MyReport.rptdesign", label="Draft")
```

removeSecurity 메소드

```
removeSecurity(source,principal,provider,cascade)
```

리포지토리의 지정된 파일 또는 폴더에서 보안 액세스 제어 목록(ACL)을 제거합니다.

표 61. `removeSecurity`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 파일 또는 폴더의 완전한 경로 또는 오브젝트 URI	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a7860000010dcee0eaa28219"
<code>principal</code>	필수	문자열	지정된 파일 또는 폴더에서 제거할 사용자(예: <code>admin</code>)	<code>admin</code>
<code>provider</code>	선택사항	문자열	사용자에 대한 정보를 얻는 데 사용할 보안 제공자(예: <code>Native</code>). 유효한 값은 다음과 같습니다. <ul style="list-style-type: none"> • 시스템에 고유한 네이티브 로컬 보안 제공자의 네이티브. 이는 기본 제공자입니다. • Active Directory의 <code>AD_<name></code>. 여기서, <code><name></code>은 시스템 내의 보안 제공자 이름에 해당합니다. • 로컬 대체가 사용된 Active Directory의 <code>ADL_<name></code>. 여기서, <code><name></code>은 시스템 내의 보안 제공자 이름에 해당합니다. • OpenLDA의 <code>ldap_<name></code>. 여기서, <code><name></code>은 시스템 내의 보안 제공자 이름에 해당합니다. 	<code>Native</code>
<code>cascade</code>	선택사항	부울	지정된 폴더 내의 모든 파일 및 하위 폴더에 보안 설정을 전파함	<code>True</code> 또는 <code>False</code>

표 62. `removeSecurity`에 대한 리턴 값.

유형	설명
Boolean	<code>True</code> 또는 <code>False</code> (메소드의 성공적인 실행 여부를 기반으로 함)

표 63. `removeSecurity`에 대한 예외.

유형	설명
<code>ResourceNotFoundException</code>	소스 파일 또는 대상 폴더가 없습니다.
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.
<code>IllegalParameterException</code>	지정된 사용자 또는 보안 제공자 이름이 올바르지 않습니다.

예제

다음 샘플에서는 `MyReport.rptdesign` 파일에서 프린시펄의 ACL을 제거합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
bSuccess = pesImpl.removeSecurity(source="/Projects/MyReport.rptdesign",principal="icrod")
```

search 메소드

리포지토리에서 파일을 검색하며 검색 기준과 일치하는 메타데이터 콘텐츠가 포함된 파일 버전의 목록을 리턴합니다.

`search(criteria)`

표 64. `search`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>criteria</code>	필수	문자열	파일 메타데이터를 검색하는 데 사용된 값	"Age"

표 65. `search`에 대한 리턴 값.

유형	설명
PageResult	각 행이 검색 일치에 해당하는 구조. 자세한 정보는 57 페이지의 『PageResult 클래스』 주제를 참조하십시오.

표 66. `search`에 대한 예외.

유형	설명
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 예제에서는 메타데이터 필드에 *Quarterly* 텍스트가 포함된 파일 버전을 검색합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
sResults = pesImpl.search(criteria="Quarterly")
sRows = sResults.getRows()
for sRow in sRows:
    print "Author: ", sRow.getAuthor()
    print "Title: ", sRow.getTitle()
    for child in sRow.getChildRow():
        print "Version: ", child.getVersionMarker()
        print "Label: ", child.getVersionLabel()
        print "Keywords:", child.getKeyword()
        print "URI:", child.getUri()
```

setLabel 메소드

리포지토리에서 파일의 버전에 레이블을 적용합니다. 파일이 이미 레이블된 경우 원래의 레이블이 새 레이블로 대체됩니다.

`setLabel(source, version, label)`

표 67. `setLabel`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 파일의 완전한 경로 또는 오브젝트 URI	"/Temp Folder/Temp.txt" 또는 "0a58c3670016a7860000 010dcee0eaa28219"
<code>version</code>	필수	문자열	파일의 특정 버전	"0:2006-08-25 21:15:49.453"

표 67. `setLabel`에 대한 입력 매개변수 (계속).

필드	사용	유형	설명	예제 값
<code>label</code>	필수	문자열	파일에 적용할 레이블	"Version 1"

표 68. `setLabel`에 대한 리턴 값.

유형	설명
String	업데이트된 파일의 URI

표 69. `setLabel`에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일 또는 버전이 없습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예제

다음 샘플에서는 *Beta* 레이블을 `MyReport.rptdesign` 파일의 두 번째 버전에 지정합니다. Resource 오브젝트에 대한 `getVersionMarker` 메소드는 레이블될 버전의 마커를 리턴합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
betaVersion = \
    pesImpl.getAllVersions(source="/Demo/Drafts/MyReport.rptdesign")[1].getVersionMarker()
print "Marker for the beta version is:", betaVersion
uri = pesImpl.setLabel(source="/Demo/Drafts/MyReport.rptdesign", version=betaVersion,
    label="Beta")
```

setMetadata 메소드

메타데이터 특성을 리포지토리의 파일 및 폴더에 적용합니다.

다음 표에서는 메타데이터 특성과 이 특성이 파일 및 폴더에 적용될 수 있는지를 식별합니다.

표 70. 리포지토리 오브젝트 특성.

메타데이터 특성	자원 유형
작성자	파일
설명	파일 또는 폴더
제목	파일 또는 폴더
만료 날짜	파일 또는 폴더
키워드	파일
토픽	파일
사용자 정의 메타데이터	파일 또는 폴더

```
setMetadata(source,version,label,props)
```

표 71. `setMetadata`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 파일 또는 폴더의 완전한 경로 또는 오브젝트 URI	<code>"/Temp Folder/Temp.txt"</code> 또는 <code>"0a58c3670016a7860000 010dcee0eaa28219"</code>
<code>version</code>	선택 사항. version 또는 label 중 하나를 지정할 수 있습니다.	문자열	다운로드할 파일의 특정 버전	<code>"0:2006-08-25 21:15:49.453"</code>
<code>label</code>	선택 사항. version 또는 label 중 하나를 지정할 수 있습니다.	문자열	특정 버전의 레이블	<code>"Label 1"</code>
<code>props</code>	필수	사전	키로 메타데이터 이름을 사용한 사전에는 설정될 모든 메타데이터가 포함됩니다. 예제 값 열에 표시된 대로, 목록을 topic 및 customProperty의 사전에서의 값으로 사용합니다. 나머지 메타데이터에는 문자열을 사용합니다.	<pre>{ 'author': 'admin', 'title': 'newTitle', 'description': 'desc', 'topic': [a,b], 'customProperty': { 'language': 'hindi english', 'FreeForm': 'abcd' } }</pre>

표 72. `setMetadata`에 대한 리턴 값.

유형	설명
String	메타데이터가 설정되는 파일 또는 폴더의 URI

표 73. `setMetadata`에 대한 예외.

유형	설명
<code>InsufficientParameterException</code>	필수 매개변수가 지정되어 있지 않습니다.
<code>ResourceNotFoundException</code>	소스 파일 또는 폴더가 없습니다.

예제

다음 샘플에서는 `Quarterly` 키워드를 `MyReport.rptdesign` 파일의 `Production` 버전에 지정합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
pDict = {'keyword': 'Quarterly'}
uri = pesImpl.setMetadata(source="/Demo/Drafts/MyReport.rptdesign", version=prodVersion,
    props=pDict)
print uri
```

uploadFile 메소드

파일의 새 버전을 작성하는 옵션을 사용하여(이미 존재하는 경우) 로컬 파일 시스템의 리포지토리에 파일을 저장합니다.

`uploadFile(source, target, versionFlag)`

표 74. `uploadFile`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	업로드할 파일의 완전한 경로(로컬 파일 시스템에서)	"C:\Temp\Temp.txt"
<code>target</code>	필수	문자열	파일이 업로드될 리포지토리에서 대상 폴더의 완전한 경로입니다.	"/Temp Folder"
<code>versionFlag</code>	선택사항	부울	지정된 파일이 이미 존재하는 경우 파일의 새 버전이 작성됩니다.	True 또는 False

표 75. `uploadFile`에 대한 리턴 값.

유형	설명
String	업로드된 파일의 URI

표 76. `uploadFile`에 대한 예외.

유형	설명
ResourceNotFoundException	소스 파일 또는 대상 폴더가 없습니다.
ResourceAlreadyExistsException	소스 파일과 이름이 동일한 파일 또는 폴더가 대상 폴더에 존재하고 <code>versionFlag</code> 매개변수가 지정되지 않습니다.
InsufficientParameterException	필수 매개변수가 지정되어 있지 않습니다.

예제

이 예제에서는 리포지토리에서 `MyReport.rptdesign` 파일을 `/Demo/Drafts` 폴더에 업로드합니다. 파일이 이미 존재하는 경우 파일의 새 버전이 `versionFlag` 매개변수를 사용하여 업로드됩니다.

```
from pes.util.PESExceptions import *
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
try:
    uri = pesImpl.uploadFile(source="C:\Demo\MyReport.rptdesign",target="/Demo/Drafts")
    print "URI for the uploaded file is: ", uri
except ResourceAlreadyExistsException:
    uri = pesImpl.uploadFile(source="C:\Demo\MyReport.rptdesign",target="/Demo/Drafts",
        versionFlag=True)
    print "URI for the uploaded file is: ", uri
```

랩퍼 클래스

PESImpl API에는 콘텐츠 리포지토리 메소드가 호출한 웹 서비스에서 리턴된 오브젝트에 대해 랩퍼의 역할을 수행하는 클래스가 포함됩니다. 이 랩퍼 클래스는 메소드에서 리턴된 정보를 표시하는 인터페이스를 제공합니다.

Resource 클래스

Resource 클래스는 오브젝트별 데이터에 대한 액세스를 제공하는 ResourceSpecifier.Resource 리포지토리 오브젝트에 단순화된 래퍼의 역할을 수행합니다.

이 클래스에는 리포지토리 오브젝트와 연관된 표준 메타데이터를 비롯하여 리포지토리의 오브젝트에 대해 정의된 사용자 정의 메타데이터 정보가 포함됩니다. 표 77은 Resource 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 77. Resource 클래스에 대한 메소드.

메소드 이름	설명
getAccessControlList	오브젝트의 보안 권한에 대한 사전을 리턴함. 키와 사용자에게 제공되는 가장 높은 수준의 권한만으로 사용되는 사용자 이름이 포함됩니다. 예를 들면, 다음과 같습니다. Joe라는 사용자가 자원 X에 대해 삭제 권한이 있는 경우, X를 표시하는 자원 오브젝트의 getAccessControlList가 웹 서비스 호출에서 {'Joe':'DELETE'}를 리턴하고 세 개의 권한(읽기, 쓰기, 삭제)을 모두 리턴하지는 않습니다.
getOwner	문자열로 오브젝트의 소유자 이름을 리턴함
getAuthor	문자열로 오브젝트의 작성자 이름을 리턴함
getContentSize	오브젝트의 크기를 리턴함
getCreatedBy	문자열로 오브젝트를 작성한 사용자의 이름을 리턴함
getCreationDate	datetime 오브젝트로 오브젝트의 작성 날짜를 리턴함
getDescription	목록으로 오브젝트의 설명을 리턴함
getDescriptionLanguage	목록으로 오브젝트의 언어를 리턴함
getExpirationDate	datetime 오브젝트로 오브젝트의 만료 날짜를 리턴함
isExpired	지정된 오브젝트가 만료되었는지를 표시함
getMimeType	문자열로 오브젝트의 MIME 유형을 리턴함
getModificationDate	datetime 오브젝트로 오브젝트의 마지막으로 수정된 날짜를 리턴함
getObjectCreationDate	datetime 오브젝트로 오브젝트의 오브젝트 작성 날짜를 리턴함
getObjectLastModifiedBy	문자열로 오브젝트를 마지막으로 수정한 사용자를 리턴함
getObjectLastModifiedDate	datetime 오브젝트로 오브젝트의 마지막으로 수정된 날짜를 리턴함
getResourceID	문자열로 오브젝트의 자원 식별자를 리턴함
getResourcePath	문자열로 지정된 오브젝트의 경로를 리턴함
getTitle	문자열로 오브젝트의 제목을 리턴함
getTopicList	오브젝트에 대한 토픽의 목록을 리턴함
getVersionMarker	문자열로 오브젝트의 버전을 리턴함
getVersionLabel	문자열로 오브젝트의 레이블을 리턴함
getCustomMetadata	사전으로 오브젝트와 연관된 사용자 정의 특성을 리턴함
getKeywordList	오브젝트와 연관된 키워드의 목록을 리턴함

IdentificationSpecifier 클래스

이 클래스는 오브젝트의 ID별 데이터에 대한 액세스를 허용하여 IdentificationSpecifier 리포지토리 오브젝트에 단순화된 래퍼의 역할을 수행합니다.

표 78은 IdentificationSpecifier 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 78. IdentificationSpecifier 클래스에 대한 메소드.

메소드 이름	설명
getIdentifier	문자열로 오브젝트의 식별자 값을 리턴함
getVersionMarker	문자열로 오브젝트의 버전을 리턴함
getVersionLabel	문자열로 오브젝트 버전에 적용된 레이블을 리턴함

PageResult 클래스

이 PageResult 클래스는 검색 결과에 대한 컨테이너 역할을 수행합니다. 결과에서 개별 히트는 PageResult 오브젝트의 행에 해당합니다.

예를 들어, 네 개의 자원을 리턴하는 검색으로 네 개의 행이 포함된 PageResult 오브젝트가 생성됩니다. 표 79는 PageResult 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 79. PageResult 클래스에 대한 메소드.

메소드 이름	설명
getRows	SearchRow 오브젝트의 목록을 리턴함. 자세한 정보는 『SearchRow 클래스』 주제를 참조하십시오.

SearchRow 클래스

SearchRow 클래스는 개별 검색 결과에 대한 오브젝트 레벨 정보의 컨테이너 역할을 수행합니다. 이 클래스의 메소드를 사용하여 오브젝트에 대한 메타데이터에 액세스할 수 있습니다.

표 80은 SearchRow 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 80. SearchRow 클래스에 대한 메소드.

메소드 이름	설명
getTitle	파일 또는 폴더의 이름을 리턴함
getAuthor	파일 또는 폴더의 작성자를 리턴함
getMimeType	파일의 MIME 유형을 리턴함
getObjectLastModifiedBy	파일 또는 폴더를 마지막으로 수정한 사용자를 리턴함
getModified	파일 또는 폴더가 마지막으로 수정된 날짜 및 시간을 리턴함
getFolderPath	파일 또는 폴더의 위치를 리턴함
getFolder	파일 또는 폴더의 상위 폴더 이름을 리턴함
getParentURI	상위의 오브젝트 URI를 리턴함
getTopic	파일 또는 폴더와 연관된 토픽을 리턴함
getChildRow	SearchChildRow 오브젝트의 목록을 리턴함(자세한 정보는 다음 절 참조).

오브젝트에 대한 버전 레벨의 정보에 액세스하려면 getChildRow 메소드를 사용하여 오브젝트 버전에 해당하는 하위 행을 리턴하십시오.

SearchChildRow 클래스

SearchChildRow 클래스는 개별 검색 결과에 대한 버전 레벨 정보의 컨테이너 역할을 수행합니다. 이 클래스의 메소드를 사용하여 오브젝트 버전에 대한 메타데이터에 액세스할 수 있습니다.

표 81은 SearchChildRow 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 81. SearchChildRow 클래스에 대한 메소드.

메소드 이름	설명
getExpirationDate	파일 또는 폴더의 만료 날짜를 리턴함
getKeyword	파일 또는 폴더의 버전과 연관된 키워드를 리턴함
getVersionLabel	파일 또는 폴더의 버전 레이블을 리턴함
getDescription	파일 또는 폴더의 설명을 리턴함
getLanguage	언어를 리턴함
getVersionCreationDate	파일 또는 폴더가 작성된 날짜 및 시간을 리턴함
getVersionMarker	파일 또는 폴더의 버전 마커를 리턴함
getUri	파일 또는 폴더의 오브젝트 URI를 리턴함

프로세스 관리 API

프로세스 관리 스크립팅은 작업할 수 있는 기능을 제공합니다. 이 영역에는 다음 기능이 포함되어 있습니다.

- 작업 실행
- 작업 히스토리 검색
- 작업 세부사항 검색

이 절에서는 리포지토리에 저장된 작업에 대해 작업하는 데 사용되는 PESImpl 메소드에 대해 간략하게 설명합니다. 모든 메소드에는 상세한 구문 정보, 예제 및 예상되는 메시지가 포함됩니다.

메소드

다음 절에서는 IBM SPSS Collaboration and Deployment Services에 대해 지원되는 모든 프로세스 관리 스크립팅 메소드를 나열합니다.

참조: 리포지토리의 파일/폴더에 경로가 필요한 모든 메소드의 경우 경로 또는 오브젝트 URI를 사용할 수 있습니다. IBM SPSS Deployment Manager에서 오브젝트 특성 보기를 통해 오브젝트 URI를 확인할 수 있습니다.

cancelJob 메소드

실행 중인 작업을 취소합니다.

```
cancelJob(executionId)
```

표 82. `cancelJob`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>executionId</code>	필수	문자열	작업에 대한 실행 ID	0a58c33d002ce90800 00010e0ccf7b01800e

표 83. `cancelJob`에 대한 리턴 값.

유형	설명
Boolean	작업이 취소될 때 성공 메시지를 리턴함

예제

이 예제에서는 `Reports` 작업의 실행을 종료합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
execId = pesImpl.executeJob(source='/Demo/Jobs/Reports', notification = True,
    asynchronous=True)
print "Execution ID: ", execId
status = pesImpl.cancelJob(execId)
print "Successful cancellation: ", status
```

`deleteJobExecutions` 메소드

하나 이상의 작업 실행을 삭제합니다.

`deleteJobExecutions(executionId)`

표 84. `deleteJobExecutions`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>executionId</code>	필수	목록	작업 실행 식별자의 목록	[0a58c33d002ce9080000010e0ccf7b01800e, 0a59c33d002ce9080060010e0cdf7b01802r]

표 85. `deleteJobExecutions`에 대한 리턴 값.

유형	설명
Boolean	True 또는 False(메소드의 성공적인 실행 여부를 기반으로 함)

예제

이 예제에서는 `Reports` 작업의 실행을 삭제합니다.

```
from pes.util.PESExceptions import *
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
executions = pesImpl.getJobExecutionList(source="/Demo/Jobs/Reports")
execRows = executions.getRows()

# Get the execution ID from the execution history
deleteList = []
for exrow in execRows :
    uuid = exrow.getEventObjId()
    deleteList.append(uuid)
```

```
if len(deleteList) != 0:
    print 'Deleting ',len(deleteList) ,' histories'
    pesImpl.deleteJobExecutions(deleteList)
```

executeJob 메소드

전달된 매개변수에 기반하여 동기식 또는 비동기식으로 작업을 실행합니다. 동기 실행의 경우 메소드는 작업이 완료될 때까지 리턴하지 않습니다. 비동기 실행의 경우 메소드는 작업이 시작된 후 리턴합니다.

`executeJob(source,notification,asynchronous)`

표 86. `executeJob`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	업로드할 파일의 완전한 경로(로컬 파일 시스템에서)	"C:\Temp\Temp.txt"
<code>notification</code>	선택사항	부울	작업이 알림과 함께 실행되는지 또는 알림 없이 실행되는지를 표시합니다. 기본값은 False입니다.	True 또는 False
<code>asynchronous</code>	선택사항	부울	작업이 비동기로 실행되는지를 표시합니다. 기본값은 False입니다.	True 또는 False

표 87. `executeJob`에 대한 리턴 값.

유형	설명
String	작업의 실행에 대한 고유 식별자입니다. 이 식별자는 특정 작업 실행을 참조하는 데 사용됩니다.

예제

이 예제에서는 알림 없이 비동기식으로 `Reports` 작업의 실행을 시작합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
execId = pesImpl.executeJob(source='/Demo/Jobs/Reports', notification = True,
    asynchronous=True)
print "Execution ID: ", execId
```

getJobExecutionDetails 메소드

작업 단계 및 반복을 포함한 특정 작업에 대한 실행 세부사항을 나열합니다.

`getJobExecutionDetails(executionId,log,target)`

표 88. `getJobExecutionDetails`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>executionId</code>	필수	문자열	작업의 실행 ID	0a58c33d002ce9080000 010e0ccf7b01800e
<code>log</code>	선택사항	부울	작업 로그가 인라인으로 표시되는지를 나타냅니다.	True 또는 False

표 88. `getJobExecutionDetails`에 대한 입력 매개변수 (계속).

필드	사용	유형	설명	예제 값
<code>target</code>	선택사항	문자열	로그를 저장할 위치(로컬 파일 시스템에서). <code>--log</code> 매개변수와 함께만 사용됩니다.	"c:\logs"

표 89. `getJobExecutionDetails`에 대한 리턴 값.

유형	설명
<code>jobExecutionDetails</code>	작업 실행에 대한 세부사항. 자세한 정보는 63 페이지의 『 <code>jobExecutionDetails</code> 클래스』 주제를 참조하십시오.

예제

이 예제에서는 작업 실행을 위해 `execId` ID를 사용하여 작업 단계 실행에 대한 정보를 검색하고, 각 단계에 대한 결과를 콘솔에 전송합니다.

```
from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
execDetails = pesImpl.getJobExecutionDetails(executionId=execId)
print "Job ID: ", execDetails.getUUID()
print "Event ID: ", execDetails.getEventUUID()
print "Started: ", execDetails.getStartDateTime()
print "Ended: ", execDetails.getEndDateTime()
for step in execDetails.getJobStepDetails():
    print "Step ID: ", step.getEventUUID()
    print "Step Name: ", step.getEventName()
    print "Started: ", step.getStartDateTime()
    print "Ended: ", step.getEndDateTime()
    print "Success: ", step.getExecutionSuccess()
```

`getJobExecutionList` 메소드

모든 버전의 작업에서 현재 실행 중인 작업 및 완료된 작업을 포함하여 특정 작업에 대한 실행을 나열합니다.

```
getJobExecutionList(source)
```

표 90. `getJobExecutionList`에 대한 입력 매개변수.

필드	사용	유형	설명	예제 값
<code>source</code>	필수	문자열	리포지토리에서 작업의 완전한 경로	"/testJob"

표 91. `getJobExecutionList`에 대한 리턴 값.

유형	설명
<code>PageResult</code>	작업에 대한 실행 목록의 컨테이너. 자세한 정보는 62 페이지의 『 <code>PageResult</code> 클래스』 주제를 참조하십시오. .

예제

이 예제에서는 `Reports` 작업에 대한 실행을 검색하고, 각 실행에 대한 정보를 콘솔에 전송합니다.

```

from pes.api.PESImpl import PESImpl
pesImpl = PESImpl("admin", "spss", "localhost", "8080")
executions = pesImpl.getJobExecutionList(source="/Demo/Jobs/Reports")
execRows = executions.getRows()
if execRows:
    for exrow in execRows:
        print "Job Path: ", exrow.getPath()
        print "Object ID: ", exrow.getObjId()
        print "Event ID: ", exrow.getEventObjId()
        print "Version ", exrow.getVersionMarker()
        print "Started: ", exrow.getEventStartDateTime()
        print "Ended: ", exrow.getEventEndDateTime()

```

랩퍼 클래스

PESImpl API에는 프로세스 관리 메소드가 호출한 웹 서비스에서 리턴된 오브젝트에 대해 랩퍼의 역할을 수행하는 클래스가 포함됩니다. 이 랩퍼 클래스는 메소드에서 리턴된 정보를 표시하는 인터페이스를 제공합니다.

PageResult 클래스

이 PageResult 클래스는 작업 실행 특정 데이터의 검색을 허용하는 작업 실행 결과에 대한 컨테이너 역할을 수행합니다.

개별 작업 실행은 PageResult 오브젝트의 행에 해당합니다. 예를 들어, 네 번 실행한 작업은 네 개의 행이 포함된 PageResult 오브젝트에 해당합니다. 표 92는 PageResult 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 92. PageResult 클래스에 대한 메소드.

메소드 이름	설명
getRows	작업의 실행을 각각 표시하는 Row 오브젝트의 목록을 리턴합니다. 자세한 정보는 『Row 클래스』 주제를 참조하십시오.

Row 클래스

Row 클래스는 작업 실행에 대한 작업 레벨 정보의 컨테이너 역할을 수행합니다. 이 클래스의 메소드를 사용하여 작업 실행에 대한 메타데이터에 액세스할 수 있습니다.

표 93은 Row 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 93. Row 클래스에 대한 메소드.

메소드 이름	설명
getObjId	작업의 실행 ID를 리턴함
getPath	작업의 경로를 리턴함
getVersionMarker	실행된 작업의 버전 마커를 리턴함
getVersionLabel	실행된 작업의 버전 레이블을 리턴함
getEventObjId	실행된 작업의 이벤트 ID를 리턴함
getEventState	실행 중인 작업의 상태를 리턴함
getEventCompletionCode	작업의 완료 코드를 리턴함

표 93. Row 클래스에 대한 메소드 (계속).

메소드 이름	설명
getEventStartDateTime	작업의 시작 날짜 및 시간을 리턴함
getEventEndDateTime	작업의 종료 날짜 및 시간을 리턴함
getQueuedDateTime	작업의 큐된 날짜 및 시간을 리턴함

jobExecutionDetails 클래스

이 클래스는 getJobExecutionDetails 메소드에서 리턴됩니다. 작업의 실행 세부사항을 저장하고 작업의 각 단계에 대한 정보를 제공하는 jobStepExecution 오브젝트의 목록을 포함합니다.

표 94는 jobExecutionDetails 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 94. jobExecutionDetails 클래스에 대한 메소드.

메소드 이름	설명
getJobStepDetails	jobStepExecutionDetails 오브젝트의 목록을 리턴함. 자세한 정보는 『jobStepExecutionDetails 클래스』 주제를 참조하십시오.
getArtifactLocation	작업 아티팩트 위치의 목록을 리턴함
getCompletionCode	작업 실행의 완료 코드를 리턴함
getEndTime	작업 실행의 종료 날짜 및 시간을 리턴함
getEventName	작업 실행의 이벤트 이름을 리턴함
getEventUUID	작업 실행의 이벤트 ID를 리턴함
getExecutionState	작업 실행의 실행 상태를 리턴함
getExecutionSuccess	작업 실행의 성공 또는 실패 상태를 리턴함
getExecutionWarning	경고 발생 여부를 표시함
getLog	생성된 로그(문자열로)를 리턴함
getNotificationEnabled	이메일 알림의 사용 가능 여부를 표시함
getQueuedDateTime	작업 실행의 큐된 날짜 및 시간을 리턴함
getStartDateTime	작업 실행의 시작 날짜 및 시간을 리턴함
getUserName	작업을 실행한 사용자의 이름을 리턴함
getUUID	작업의 실행 ID를 리턴함

jobStepExecutionDetails 클래스

이 클래스는 작업 단계의 실행 세부사항을 저장하고 jobStepChildExecutionDetails 오브젝트의 목록을 저장합니다. 이 클래스에는 모든 메소드 호출을 위임하는 ExecutionDetails 오브젝트가 포함됩니다.

64 페이지의 표 95는 jobStepExecutionDetails 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 95. *jobStepExecutionDetails* 클래스에 대한 메소드.

메소드 이름	설명
<code>getJobStepChildExecutionList</code>	<code>jobStepChildExecutionDetails</code> 오브젝트의 목록을 리턴함. 자세한 정보는 『 <code>jobStepChildExecutionDetails</code> 클래스』 주제를 참조하십시오.
<code>getArtifactLocation</code>	작업 단계 아티팩트 위치의 목록을 리턴함
<code>getCompletionCode</code>	작업 단계의 완료 코드를 리턴함
<code>getEndDateTime</code>	작업 단계의 종료 날짜 및 시간을 리턴함
<code>getEventName</code>	작업 단계의 이벤트 이름을 리턴함
<code>getEventUUID</code>	작업 단계의 이벤트 ID를 리턴함
<code>getExecutionState</code>	작업 단계의 실행 상태를 리턴함
<code>getExecutionSuccess</code>	작업 단계의 성공 또는 실패 상태를 리턴함
<code>getExecutionWarning</code>	경고 발생 여부를 표시함
<code>getLog</code>	생성된 로그(문자열로)를 리턴함
<code>getNotificationEnabled</code>	이메일 알림의 사용 가능 여부를 표시함
<code>getQueuedDateTime</code>	작업 단계의 큐된 날짜 및 시간을 리턴함
<code>getStartDateTime</code>	작업 단계의 시작 날짜 및 시간을 리턴함
<code>getUserName</code>	작업 단계를 실행한 사용자의 이름을 리턴함
<code>getUUID</code>	작업 단계의 실행 ID를 리턴함

jobStepChildExecutionDetails 클래스

`jobStepChildExecutionDetails` 클래스는 개별 작업 단계의 하위 실행을 위한 컨테이너 역할을 수행합니다. 예를 들어, 반복 보고서 작업 단계에서 단계의 각 반복에 대한 하위 실행을 생성합니다. 이 클래스의 메소드를 사용하여 하위 실행에 대한 메타데이터에 액세스할 수 있습니다.

표 96은 `jobStepChildExecutionDetails` 클래스에 사용 가능한 모든 메소드를 나열합니다.

표 96. *jobStepChildExecutionDetails* 클래스에 대한 메소드.

메소드 이름	설명
<code>getArtifactLocation</code>	하위 실행 아티팩트 위치의 목록을 리턴함
<code>getCompletionCode</code>	하위 실행의 완료 코드를 리턴함
<code>getEndDateTime</code>	하위 실행의 종료 날짜 및 시간을 리턴함
<code>getEventName</code>	하위 실행의 이벤트 이름을 리턴함
<code>getEventUUID</code>	하위 실행의 이벤트 ID를 리턴함
<code>getExecutionState</code>	하위 실행의 실행 상태를 리턴함
<code>getExecutionSuccess</code>	하위 실행의 성공 또는 실패 상태를 리턴함
<code>getExecutionWarning</code>	경고 발생 여부를 표시함
<code>getLog</code>	생성된 로그(문자열로)를 리턴함
<code>getNotificationEnabled</code>	이메일 알림이 사용 가능한지를 표시함
<code>getQueuedDateTime</code>	하위 실행의 큐된 날짜 및 시간을 리턴함
<code>getStartDateTime</code>	하위 실행의 시작 날짜 및 시간을 리턴함
<code>getUserName</code>	하위 실행을 실행한 사용자의 이름을 리턴함

표 96. *jobStepChildExecutionDetails* 클래스에 대한 메소드 (계속).

메소드 이름	설명
getUUID	하위 실행의 실행 ID를 리턴함

예제 스크립트

PESImpl 클래스의 사용을 설명하는 예제 스크립트는 다음 디렉토리에 설치됩니다.

<설치 위치>/samples

이 스크립트는 다음을 포함한 다양한 태스크를 수행합니다.

- IBM SPSS Collaboration and Deployment Services Repository에서 만료된 항목 삭제
- 만료된 제출 아티팩트 삭제
- 작업 히스토리 삭제

IBM SPSS Deployment Manager의 일반 작업 단계에서 스크립트를 호출하여 리포지토리 유지보수 태스크를 수행할 수 있습니다.

주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다. 본 자료는 다른 언어로도 제공될 수 있습니다. 그러나 자료에 접근하기 위해서는 해당 언어로 된 제품 또는 제품 버전의 사본이 필요할 수 있습니다.

IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 라이선스까지 부여하는 것은 아닙니다. 라이선스에 대한 의문사항은 다음으로 문의하십시오.

07326

서울특별시 영등포구

국제금융로 10, 3IFC

한국 아이.비.엠 주식회사

대표전화서비스: 02-3781-7114

2바이트(DBCS) 정보에 관한 라이선스 문의는 한국 IBM에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan Ltd.

19-21, Nihonbashi-Hakozakicho, Chuo-ku

Tokyo 103-8510, Japan

IBM은 타인의 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여(단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증 없이 이 책을 "현상태대로" 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 변경된 사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통지 없이 언제든지 개선 및/또는 변경할 수 있습니다.

이 정보에서 언급되는 비IBM 웹 사이트는 단지 편의상 제공된 것으로, 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트 사용으로 인한 위험은 사용자 본인이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

(i) 독립적으로 작성된 프로그램과 기타 프로그램(본 프로그램 포함) 간의 정보 교환 및 (ii) 교환된 정보의 상호 이용을 목적으로 본 프로그램에 관한 정보를 얻고자 하는 라이선스 사용자는 다음 주소로 문의하십시오.

07326

서울특별시 영등포구

국제금융로 10, 31FC

한국 아이.비.엠 주식회사

대표전화서비스: 02-3781-7114

이러한 정보는 해당 조건(예를 들면, 사용료 지불 등)하에서 사용될 수 있습니다.

이 정보에 기술된 라이선스가 부여된 프로그램 및 프로그램에 대해 사용 가능한 모든 라이선스가 부여된 자료는 IBM이 IBM 기본 계약, IBM 프로그램 라이선스 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

인용된 성능 데이터와 고객 예제는 예시 용도로만 제공됩니다. 실제 성능 결과는 특정 구성과 운영 조건에 따라 다를 수 있습니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 제품들을 테스트하지 않았으므로, 비IBM 제품과 관련된 성능의 정확성, 호환성 또는 기타 청구에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM이 제시하는 방향 또는 의도에 관한 모든 언급은 특별한 통지 없이 변경될 수 있습니다.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이들 예제에는 개념을 가능한 완벽하게 설명하기 위하여 개인, 회사, 상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 인물 또는 기업의 이름과 유사하더라도 이는 전적으로 우연입니다.

저작권 라이선스:

이 정보에는 여러 운영 플랫폼에서의 프로그래밍 기법을 보여주는 원어로 된 샘플 응용프로그램이 들어 있습니다. 귀하는 이러한 샘플 프로그램의 작성 기준이 된 운영 플랫폼의 애플리케이션 프로그래밍 인터페이스(API)에 부합하는 애플리케이션을 개발, 사용, 판매 또는 배포할 목적으로 IBM에 추가 비용을 지불하지 않고 이들 샘플 프로그램을 어떠한 형태로든 복사, 수정 및 배포할 수 있습니다. 이러한

샘플 프로그램은 모든 조건하에서 완전히 테스트된 것은 아닙니다. 따라서 IBM은 이러한 프로그램의 신뢰성, 서비스 가능성 또는 기능을 보증하거나 진술하지 않습니다. 본 샘플 프로그램은 일체의 보증 없이 "현상태대로" 제공됩니다. IBM은 귀하의 샘플 프로그램 사용과 관련되는 손해에 대해 책임을 지지 않습니다.

개인정보 보호정책 고려사항

SaaS(Software as a Service) 솔루션을 포함한 IBM 소프트웨어 제품(이하 "소프트웨어 오퍼링")은 제품 사용 정보를 수집하거나 최종 사용자의 경험을 개선하는 데 도움을 주거나 최종 사용자와의 상호 작용을 조정하거나 그 외의 용도로 쿠키나 기타 다른 기술을 사용할 수 있습니다. 많은 경우에 있어서, 소프트웨어 오퍼링은 개인 식별 정보를 수집하지 않습니다. IBM의 일부 소프트웨어 오퍼링은 귀하가 개인 식별 정보를 수집하도록 도울 수 있습니다. 본 소프트웨어 오퍼링이 쿠키를 사용하여 개인 식별 정보를 수집할 경우, 본 오퍼링의 쿠키 사용에 대한 특정 정보가 다음에 규정되어 있습니다.

본 소프트웨어 오퍼링은 개인 식별 정보를 수집하기 위해 쿠키 및 기타 다른 기술을 사용하지 않습니다.

본 소프트웨어 오퍼링에 배치된 구성이 쿠키 및 기타 기술을 통해 최종 사용자의 개인 식별 정보 수집 기능을 고객인 귀하에게 제공하는 경우, 귀하는 통지와 동의를 위한 요건을 포함하여 이러한 정보 수집과 관련된 법률 자문을 스스로 구해야 합니다.

이러한 목적의 쿠키를 포함한 다양한 기술의 사용에 대한 자세한 정보는 IBM 개인정보 보호정책(<http://www.ibm.com/privacy/kr/ko>), IBM 온라인 개인정보 보호정책(<http://www.ibm.com/privacy/details/kr/ko>), "쿠키, 웹 비콘 및 기타 기술" 및 "IBM 소프트웨어 제품 및 SaaS(Software-as-a Service) 개인정보 보호정책(<http://www.ibm.com/software/info/product-privacy>)" 부분을 참조하십시오.

상표

IBM, IBM 로고 및 [ibm.com](http://www.ibm.com)은 전세계 여러 국가에 등록된 International Business Machines Corp.의 상표 또는 등록상표입니다. 기타 제품 및 서비스 이름은 IBM 또는 타사의 상표입니다. IBM 상표 목록은 웹 "저작권 및 상표 정보"(www.ibm.com/legal/copytrade.shtml)에 있습니다.

Adobe, Adobe 로고, PostScript 및 PostScript 로고는 미국 및/또는 기타 국가에서 사용되는 Adobe Systems Incorporated의 등록상표 또는 상표입니다.

Intel, Intel 로고, Intel Inside, Intel Inside 로고, Intel Centrino, Intel Centrino 로고, Celeron, Intel Xeon, Intel SpeedStep, Itanium 및 Pentium은 미국 또는 기타 국가에서 사용되는 Intel Corporation 또는 그 계열사의 상표 또는 등록상표입니다.

Linux는 미국 또는 기타 국가에서 사용되는 Linus Torvalds의 등록상표입니다.

Microsoft, Windows, Windows NT 및 Windows 로고는 미국 또는 기타 국가에서 사용되는 Microsoft Corporation의 상표입니다.

UNIX는 미국 및 기타 국가에서 사용되는 The Open Group의 등록상표입니다.

Java 및 모든 Java 기반 상표와 로고는 Oracle 및/또는 그 계열사의 상표 또는 등록상표입니다.

기타 제품 및 서비스 이름은 IBM 또는 타사의 상표입니다.

