
OZ .NET Server

API Developer's Guide



Table of Contents

Part I. OZ .NET Server API	1
1. Class	2
Cache	2
ConnectionPool	3
ConnectionPoolInfo	5
ConnectionPoolStatus	7
DataBind	8
Log	9
Mail	10
NameValueCollection	13
MemoryStatus	14
Module	15
Monitor	20
OZAPIException	21
OZDataSetInfo	21
OZODItem	22
OZParameter	23
OZReportParameter	24
OZServerInfo	24
Service	25
2. Interface	27
IReportInfo	27
Part II. OZ Scheduler API	29
1. Class	30
DirectExportResult	30
DirectPrintResult	31
DirectTaskResult	31
FileInfo	32
Program	33
Publisher	35
Scheduler	37
NameValueCollection	46
Task Option	47
EMail Option	49

Parameter Option	50
Scheduler Server Option	51
Scheduling Time Option	52
etc.	55
ServerInfo	56
TaskHolidayGroupInfo	57
TaskHolidayInfo	58
TaskResult	59
 Part III. OZ Repository API	 61
1. Class	62
RepositoryEX	62
Configuring the repository	63
Creating groups and users	63
Editing groups and users	64
Deleting groups and users	66
Getting groups and users	66
Setting up group and user permissions	69
Getting group and user permissions	70
Creating categories and items	71
Editing categories and items	72
Deleting categories and items	75
Getting categories and items	76
Setting up category and item permissions	79
Getting category and item permissions	80
Using item version control	83
Distribute items	87
Getting error	88
RepositoryException	89
2. Interface	90
ICategoryInfo	90
IGroupInfo	92
IHistoryInfo	93
IItemInfo	94
ILoginInfo	95
IRepository	97
IRepositoryCategory	99
IRepositoryExternalLogin	104

IRepositoryGroup	105
IRepositoryGroupAuthorityToCategory	110
IRepositoryGroupAuthorityToItem	112
IRepositoryHistory	114
IRepositoryItem	116
IRepositoryMultiLoginUser	121
IRepositoryUser	122
IRepositoryUserAuthorityToCategory	126
IRepositoryUserAuthorityToItem	129
IUserInfo	130
 Part IV. User Defined Log API	 133
1. Interface	134
IClearQuestLogTarget	134
IDMLogTarget	135
IFileStoreLogTarget	136
IHTTPStoreLogTarget	136
IMonitorLogTarget	137
IProcedureLogTarget	139
IRDBLogTarget	140
ISAPLogTarget	141
ISOAPLogTarget	142
IUDSLogTarget	143
IUserDefinedLogger	143
IUserDefinedLogTarget	144
IXMLLogTarget	145
 Part V. User Defined Resource API	 146
1. Class	147
OZUDRInfo	147
OZUDRResult	148
Resource	149
2. Interface	151
IOZUserDefinedResource	151

OZ .NET Server API

OZ .NET Server API는 다양한 [Class](#)와 [Interface](#)를 제공합니다.

Class

OZ .NET Server API에서 제공하는 클래스는 아래와 같습니다.

- [Cache](#)
- [ConnectionPool](#)
- [ConnectionPoolInfo](#)
- [ConnectionPoolStatus](#)
- [DataBind](#)
- [Log](#)
- [Mail](#)
- [MemoryStatus](#)
- [Module](#)
- [Monitor](#)
- [OZAPIException](#)
- [OZDataSetInfo](#)
- [OZODItem](#)
- [OZParameter](#)
- [OZReportParameter](#)
- [OZServerInfo](#)
- [Service](#)

Cache

오즈 서버의 캐시 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Cache

```
public Cache(string url, string id, string pw, bool autoLogin, bool useUSL)
```

오즈 서버 캐시 매니저와 관련된 처리를 수행합니다. 캐시 매니저 환경 설정 및 캐시 항목 삭제 등을 처리합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

autoLogin - 자동 로그인 여부

useUSL - USL 사용 여부

Method Detail

GetCacheConfiguration

`public OZAttributeList GetCacheConfiguration() throws OZAPIException`

오즈 서버의 캐시 설정을 가져옵니다. 조회 내용은 캐시 키 사용 여부, 캐시 저장 경로, 메모리 캐시 타임 아웃, 디스크 캐시 타임 아웃, 프리 메모리 퍼센트입니다.

SetCacheConfiguration

`public void SetCacheConfiguration(OZAttributeList attrs) throws OZAPIException`

캐시 설정을 변경합니다. 캐시 설정 변경은 해당 변경된 값을 캐시 키 사용여부, 메모리 캐시 타임 아웃, 디스크 캐시 타임 아웃, 프리 메모리 퍼센트, 캐시 저장 경로명 등의 요소로 변경합니다.

Parameters:

attrs - 캐시 설정 속성 값(Active: 캐시키 사용 여부, CACHE_FILE_PATH: 캐시 저장 경로,

DM_CACHE_FILE_PATH: Data Module 캐시 저장 경로, memoryCacheValidTime: 메모리 캐시 타임 아웃, diskCacheValidTime: 디스크 캐시 타임 아웃, FreeMemoryPercentage: 프리 메모리 퍼센트)

ConnectionPool

데이터베이스 연결 Pool 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

ConnectionPool

```
public ConnectionPool(string url, string id, string pw, bool autoLogin, bool useUSL)
```

오즈 서버의 데이터베이스 연결 Pool에 대한 인터페이스로 해당 데이터베이스의 ADO .NET/ODBC 정보의 등록, 삭제 등의 관리 기능을 수행합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

autoLogin - 자동 로그인 여부

useUSL - USL 사용 여부

Method Detail

AddPool

```
public void AddPool(ConnectionPoolInfo pool) throws OZAPIException
```

커넥션 풀 인포에 들어있는 정보에 맞춰서 커넥션 풀을 만들어 추가합니다. 커넥션 풀 인포에는 상태, 이름, 타입, 주소, 포트, SID, DB드라이버 종류, DB 서버이름, 사용자명, 암호, 최대접속, 초기 접속, 타임 아웃, URL, 프로퍼티, 사용자정의 드라이버가 있습니다.

Parameters:

pool - 추가할 ConnectionPoolInfo의 내용

RemovePool

```
public void RemovePool(string alias) throws OZAPIException
```

해당 앨리어스 이름의 ConnectionPool을 삭제합니다.

Parameters:

alias - ConnectionPool 앨리어스 이름

GetPoolInfos

```
public ConnectionPoolInfo[] GetPoolInfos() throws OZAPIException
```

모든 ConnectionPool에 대한 정보를 가져옵니다.

GetPoolStatuses

```
public ConnectionPooStatus[] GetPoolStatuses() throws OZAPIException
```

모든 ConnectionPool에 대한 상태 정보를 가져옵니다.

GetPoolInfo

```
public ConnectionPoolInfo GetPoolInfo(string alias) throws OZAPIException
```

해당 ConnectionPool에 대한 ConnectionPoolInfo 리스트를 가져옵니다.

Parameters:

alias - ConnectionPool의 앨리어스 이름

Save

```
public void Save() throws OZAPIException throws OZAPIException
```

서버에 모든 ConnectionPool의 정보를 저장합니다.

ConnectionPoolInfo

데이터베이스 연결에 대한 각종 정보를 제공하는 클래스입니다.

Property Detail

Alias

```
public string Alias
```

해당 ConnectionPool 정보의 alias명

Vendor

```
public string Vendor
```

해당 ConnectionPool 정보의 Vendor명

InitialConnections

```
public int InitialConnections
```

해당 ConnectionPool 초기 커넥션 개수

MaxConnections

```
public int MaxConnections
```

해당 ConnectionPool 최대 커넥션 개수

TimeOut

```
public int TimeOut
```

해당 커넥션을 가져올 때 timeout 시간

InitSQLs

```
public string InitSQLs
```

ConnectionPool에서 사용할 커넥션을 가져온 뒤 바로 실행하는 쿼리문

CloseSQLs

```
public string CloseSQLs
```

커넥션이 close또는 free되기 전 바로 실행하는 쿼리문

SessionQuery

```
public string SessionQuery
```

커넥션의 session 쿼리문

DoConnectioncheck

```
public bool DoConnectionCheck
```

커넥션 가져올 당시 커넥션이 유효한지 여부

TestQuery

```
public string TestQuery
```

커넥션이 유효한지 체크하기 위한 쿼리문

Items

```
public IDictionary Items
```

dbconfig.xml 파일의 Vendor마다 있는 Items의 key, Value

Decode

```
public string Decode
```

해당 ConnectionPool에 대한 디코딩셋

Encode

```
public string Encode
```

해당 ConnectionPool에 대한 인코딩셋

FetchRow

```
public int FetchRow
```

JDBC 전송 ROW 측, 결과 셋 반환 시 한꺼번에 가져올 수 있는 행 수

ConnectionPoolStatus

ConnectionPool의 상태 정보를 멤버 변수로 제공하는 클래스입니다.

Member Variables

Status

```
public Status Status
```

현재 상태

StatusString

```
public string StatusString
```

해당 상태 메시지

FreeConnectionCount

```
public int FreeConnectionCount
```

프리 상태의 커넥션 수

CheckedOutConnectionCount

```
public int CheckedOutConnectionCount
```

Checked out된 커넥션 수

ErrorMessage

```
public string ErrorMessage
```

ConnectionPool 생성 시 발생한 Error 메시지

Error

`public Exception Error`

ConnectionPool 생성 시 발생한 Error

DataBind

데이터 바인딩 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

DataBind

`public DataBind(string url, string id, string pw, bool autoLogin, bool useUSL)`

데이터 바인딩 모듈 관련 기능을 수행합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

autoLogin - 자동 로그인 여부

useUSL - USL 사용 여부

Method Detail

GetConfiguration

`public OZAttributeList GetConfiguration() throws OZAPIException`

DataBind의 설정 값 즉, "databind.properties"에 설정되어 있는 값을 가져옵니다.

SetConfiguration

`public void SetConfiguration(OZAttributeList config) throws OZAPIException`

DataBind 설정 즉, "databind.properties"에 설정되어 있는 값을 변경할 수 있습니다.

Parameters:

config - DataBind 설정 속성(ConcurrentFetchSize: 클라이언트로 데이터를 전송할 때 Stream 버퍼의 크기, ConcurrentFirstRow: 클라이언트로 데이터를 전송할 때 맨 처음 전송되는 첫 데이터의 행 개수)

Log

오즈 서버 로그 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Log

```
public Log(string url, string id, string pw, bool autoLogin, bool useUSL)
```

오즈 서버 로그 매니저와 관련된 처리를 수행하며 서버 운용시 발생하는 각종 메시지, 정보, 에러 등을 기록하기 위한 환경 설정을 변경합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

autoLogin - 자동 로그인 여부

useUSL - USL 사용 여부

Method Detail

GetConfiguration

```
public string GetConfiguration() throws OZAPIException
```

로그 설정을 서버로부터 가져옵니다.

DownloadLog

```
public Stream DownloadLog() throws OZAPIException
```

서버로부터 로그 파일을 다운로드합니다.

DownloadLog

```
public Stream DownloadLog(string fileName) throws OZAPIException
```

서버로부터 다운로드한 로그 파일을 로컬에 저장합니다.

Parameters:

fileName - 저장할 파일명

SetConfiguration

public void SetConfiguration(string logs) throws OZAPIException

로그 환경 설정 값을 변경합니다.

Parameters:

logs - 변경할 로그 환경 설정 값, "key=value" 형식으로 설정

GetFileNames

public string[] GetFileNames() throws OZAPIException

생성된 로그 파일명을 가져옵니다.

Mail

메일 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Mail

public Mail(string url, string id, string pw, bool autoLogin, bool useUSL)

메일 관련 정보를 설정하고 메일을 전송하는 기능을 수행합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

autoLogin - 자동 로그인 여부

useUSL - USL 사용 여부

Method Detail

AddAlias

```
public void AddAlias(string aliasName, NameValueCollection configMap)
```

오즈 서버에 설정한 앨리어스 이름으로 메일 설정 정보를 추가합니다.

Parameters:

aliasName - 앨리어스 이름

configMap - 메일 설정 속성 값([NameValueCollection](#) 참고)

GetAliasConfig

```
public NameValueCollection GetAliasConfig(string aliasName)
```

해당 앨리어스의 메일 설정 정보를 가져옵니다.

Parameters:

aliasName - 앨리어스 이름

GetMailAliasNames

```
public string[] GetMailAliasNames()
```

오즈 서버의 mail.properties에 설정된 메일 앨리어스를 모두 가져옵니다.

각 앨리어스의 active 여부와 관계없이 설정된 모든 앨리어스를 가져옵니다.

ModifyAlias

```
public void ModifyAlias(string aliasName, string newAliasName, NameValueCollection configMap)
```

해당 앨리어스의 앨리어스 이름 및 메일 설정 정보를 변경합니다.

Parameters:

aliasName - 앨리어스 이름

newAliasName - 변경할 앨리어스 이름

configMap - 메일 설정 속성 값([NameValueCollection](#) 참고)

RemoveAlias

```
public void RemoveAlias(string aliasName)
```

해당 앨리어스의 메일 설정 정보를 삭제합니다.

Parameters:

aliasName - 앨리어스 이름

Send

```
public void Send(string aliasName, string from, string fromUserName, string to,
string cc, string bcc, string subject, string context, bool isHTML, string
localFileFullPath, string fileName)
```

해당 앨리어스에 설정된 메일 정보로 메일을 비동기 방식으로 전송합니다.

Parameters:

aliasName - 앨리어스 이름

from - 발신자 메일 주소

fromUserName - 발신자 이름. 값을 null 또는 ""로 설정할 경우 발신자 메일 주소로 적용됨

to - 수신자 메일 주소

cc - 참조 메일 주소. 값을 null 또는 ""로 설정할 경우 참조 메일 주소로 적용되며, 여러 값을 설정할 경우
", " 또는 ";"를 구분자로 설정

bcc - 숨은 참조 메일 주소. 값을 null 또는 ""로 설정할 경우 숨은 참조 메일 주소로 적용되며, 여러 값을 설
정할 경우 ", " 또는 ";"를 구분자로 설정

subject - 메일 제목

context - 메일 내용

isHTML - 메시지 형식을 HTML로 설정할지 여부

localFileFullPath - 파일을 첨부할 경우 첨부할 파일의 전체 경로. 여러 값을 설정할 경우 "|"를 구분자로 설
정

fileName - 파일을 첨부할 경우 첨부 파일명. 첨부 파일의 구분자인 "|" 문자열은 파일명에 포함될 수 없습
니다.

SendSync

```
public bool SendSync(string aliasName, string from, string fromUserName, string
to, string cc, string bcc, string subject, string context, bool isHTML, string
localFileFullPath, string fileName)
```

해당 앨리어스에 설정된 메일 정보로 메일을 동기 방식으로 전송하고 메일 전송 성공 여부를 가져옵니다.

Parameters:

aliasName - 앨리어스 이름

from - 발신자 메일 주소

fromUserName - 발신자 이름. 값을 null 또는 ""로 설정할 경우 발신자 메일 주소로 적용됨

to - 수신자 메일 주소

cc - 참조 메일 주소. 값을 null 또는 ""로 설정할 경우 참조 메일 주소로 적용되며, 여러 값을 설정할 경우 "," 또는 ";"를 구분자로 설정

bcc - 숨은 참조 메일 주소. 값을 null 또는 ""로 설정할 경우 숨은 참조 메일 주소로 적용되며, 여러 값을 설정할 경우 "," 또는 ";"를 구분자로 설정

subject - 메일 제목

context - 메일 내용

isHTML - 메시지 형식을 HTML로 설정할지 여부

localFileFullPath - 파일을 첨부할 경우 첨부할 파일의 전체 경로. 여러 값을 설정할 경우 "|"를 구분자로 설정

fileName - 파일을 첨부할 경우 첨부 파일명. 첨부 파일의 구분자인 "|" 문자열은 파일명에 포함될 수 없습니다.

Mail > NameValueCollection

NameValueCollection에서 지원하는 Key에 대한 설명은 아래와 같습니다.

Key List

Key	설명
active	메일 전송 기능 사용 여부(true/false)
fromSend	발신자 메일 주소
toSend	수신자 메일 주소
SMTPServer	SMTP 서버 URL
SMTPServerPort	SMTP 서버 포트 번호
SMTPUserID	SMTP 인증이 필요한 메일 서버를 이용할 때 인증할 사용자 ID
SMTPUserPassword	SMTP 인증이 필요한 메일 서버를 이용할 때 인증할 사용자 ID의 패스워드

Key	설명
SMTPUserID_encrypted	SMTP 인증이 필요한 메일 서버를 이용할 때 인증할 암호화된 사용자 ID
SMTPUserPassword_encrypted	SMTP 인증이 필요한 메일 서버를 이용할 때 인증할 암호화된 사용자 패스워드
EnableSSL	SSL 사용 여부(true/false)
SendRetryCount	메일 전송 실패 시 재전송 횟수
SendRetryPeriodTime	메일 전송 실패 시 재전송 주기(단위: 초)
PrefixSubjectMessage	메일 제목 앞에 사용할 접두어

MemoryStatus

오즈 서버가 실행되고 있는 System의 메모리 상태를 제공하는 클래스입니다.

Member Variables

TotalMemory

```
public long TotalMemory
```

오즈 서버가 실행되고 있는 System의 VM의 토탈 메모리

FreeMemory

```
public long FreeMemory
```

오즈 서버가 실행되고 있는 System의 VM의 Free 메모리

FreeMemoryPercentage

```
public long FreeMemoryPercentage
```

오즈 서버가 실행되고 있는 System의 VM의 FreeMemory/TotalMemory

UseMemory

```
public long UseMemory
```

오즈 서버가 실행되고 있는 System의 VM의 사용 메모리

Module

오즈 데이터 모듈과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Module

```
public Module(string url, string id, string pw, bool autoLogin, bool useUSL)
```

오즈 데이터 모듈 관련 기능을 수행합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

autoLogin - 자동 로그인 여부

useUSL - USL 사용 여부

Property Detail

MemoAllowed

```
public bool MemoAllowed
```

메모 허용 여부

Password

```
public string Password
```

OZD 저장 시 password

Method Detail

AddODIParameter

`public void AddODIParameter(string odiName, string key, string value) throws OZAPIException`

SDM 파일을 만들기 위해 ODI에 지정되어 있는 ODI 파라미터 값을 설정합니다. ODI 파라미터 값을 설정하지 않을 경우에는 ODI 기본 파라미터의 기본 값이 사용됩니다.

Parameters:

odiName - ODI 이름

key - ODI 파라미터 이름

value - ODI 파라미터 값

AddODIParameter

`public void AddODIParameter(string odiName, string item, string category, IDictionary parameters) throws OZAPIException`

SDM 파일을 만들기 위해 ODI에 지정되어 있는 ODI 파라미터 값을 설정합니다. ODI 파라미터 값을 설정하지 않을 경우에는 ODI 기본 파라미터 기본 값이 사용됩니다.

ODI 파라미터 값에 따라 각각 다른 SDM 파일이 필요할 경우 각각의 파라미터값 별로 SDM 생성 단위를 만들고자 할 때 사용합니다.

Parameters:

odiName - 디자이너에서 지정한 ODI 이름

item - ODI 파일명

category - ODI 파일 경로

parameters - 파라미터 Key, Value 가 설정되어 있는 Dictionary

AddParameter

`public void AddParameter(string key, string value) throws OZAPIException`

SDM 파일을 만들기 위해 폼에 지정되어 있는 폼 파라미터 값을 설정합니다. 폼 파라미터 값을 설정하지 않을 경우에는 폼 파라미터 기본 값이 사용됩니다.

Parameters:

key - 폼 파라미터 이름

value - 폼 파라미터 값

AddReport

```
public IReportInfo AddReport(string itemName, string categoryName)
```

다중 보고서를 하나의 OZD 파일로 만들기 위하여 Report를 추가합니다.

Parameters:

itemName - Report를 추가할 아이템 이름

categoryName - Report를 추가할 아이템의 카테고리 이름

AddReport

```
public IReportInfo AddReport(string itemName, string categoryName, string  
displayName)
```

다중 보고서를 하나의 OZD 파일로 만들기 위하여 Report를 추가합니다.

Parameters:

itemName - Report를 추가할 아이템 이름

categoryName - Report를 추가할 아이템의 카테고리 이름

displayName - 뷰어의 보고서 트리에 표시할 보고서 이름

GetODIParameter

```
public OZParameter[] GetODIParameter(string category, string odiName)
```

ODI 파일에 추가된 ODI 파라미터를 가져옵니다.

Parameters:

category - 카테고리 이름

odiName - ODI 이름

GetOZD

```
public Stream GetOZD()
```

다중 보고서를 갖는 OZD를 생성한 후 입력 스트림을 가져옵니다.

GetOZD

```
public Stream GetOZD(string item, string category, string[] urls) throws  
OZAPIException
```

서버로부터 보고서 파일과 SDM 파일을 가져와 OZD 파일을 생성하여 가져옵니다. OZD 파일을 생성할 때 urls에 지정된 이미지 파일도 함께 저장합니다.

API 함수를 이용하여 데이터를 바인딩할 경우에는 항상 DM_TYPE="Memory", FetchType="Batch"로 바인딩됩니다. 대용량 데이터를 바인딩할 경우에는 메모리를 많이 사용하거나 결과물을 얻는데 시간이 걸릴 수 있습니다.

Parameters:

item - 아이템 이름(보고서 파일인 OZR 파일 이름)

category - 카테고리 이름

urls - OZD에 함께 저장할 이미지 파일의 URL

GetOZD

```
public Stream GetOZD(string itemName, string categoryName)
```

다중 보고서를 갖는 OZD를 생성한 후 입력 스트림을 가져옵니다.

Parameters:

itemName - 아이템 이름(보고서 파일인 OZR 파일 이름)

categoryName - 카테고리 이름

GetOZQReportParameter

```
public OZODIItem[] GetOZQReportParameter(string category, string ozqItemName)
```

OZQR 파일에 추가된 ODI 파라미터를 가져옵니다.

Parameters:

category - 카테고리 이름

ozqItemName - OZQR 이름

GetOZReportParameter

```
public OZReportParameter GetOZReportParameter(string category, string reportName)
```

보고서의 ODI 파라미터와 폼 파라미터를 가져옵니다.

FX 데이터에 추가된 ODI 파라미터는 가져오지 않습니다.

Parameters:

category - 카테고리 이름

reportName - 보고서 이름

GetOZReportDataSetInfo

```
public OZDataSetInfo[] GetOZReportDataSetInfo(string category, string reportName)  
throws OZCPEException
```

보고서에 추가된 데이터 셋 정보를 가져옵니다. 데이터 트리에 추가된 순서대로 리턴됩니다.

Parameters:

category - 카테고리 이름

reportName - 보고서 이름

RedirectODIPath

```
public void RedirectODIPath(string odiName, string path)
```

OZD 생성시 동적으로 변경할 ODI파일을 등록합니다.

Parameters:

odiName - ODI 이름

path - ODI 파일 경로

SaveOZD

```
public void saveOZD(String fileName) throws OZCPEException
```

다중 보고서를 갖는 OZD를 생성한 후 특정 경로에 OZD 파일을 저장합니다.

Parameters:

fileName - 저장할 카테고리 경로 및 파일 이름

SaveOZD

```
public void saveOZD(String fileName, String item, String category) throws  
OZCPEException
```

다중 보고서를 갖는 OZD를 생성한 후 특정 경로에 OZD 파일을 저장합니다.

Parameters:

fileName - 저장할 카테고리 경로 및 파일 이름

item - 보고서 파일(.ozr) 이름

category - 보고서 파일(.ozr)의 카테고리 이름

SaveOZD

```
public void saveOZD(String fileName, String item, String category, String[] urls)
throws OZCPEException
```

보고서 파일을 OZD 파일로 저장합니다.

API 함수를 이용하여 데이터를 바인딩할 경우에는 항상 DM_TYPE="Memory", FetchType="Batch"으로 바인딩됩니다. 대용량 데이터를 바인딩할 경우에는 메모리를 많이 사용하거나 결과물을 얻는데 시간이 걸릴 수 있습니다.

Parameters:

fileName - 저장 경로를 포함한 OZD 파일명. 설정한 저장 폴더가 없을 경우에는 자동으로 폴더 생성함

item - 보고서 파일(.ozr) 이름

category - 보고서 파일(.ozr)의 카테고리 이름

urls - OZD에 함께 저장할 이미지 파일의 URL

Monitor

오즈 서버의 모니터링 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Monitor

```
public Monitor(string url, string id, string pw, bool autoLogin, bool useUSL)
```

오즈 서버의 실시간 모니터링 관련 기능을 수행합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

autoLogin - 자동 로그인 여부

useUSL - USL 사용 여부

Method Detail

GetServerInfo

```
public OZServerInfo GetServerInfo() throws OZPAIException
```

서버의 버전 정보와 각 모듈별 버전 정보를 가져옵니다.

GetMemoryInfo

```
public MemoryStatus GetMemoryInfo() throws OZAPIException
```

현재의 메모리 사용량(전체 메모리, 사용 메모리, 비사용 메모리) 정보를 가져옵니다.

DownloadLog

```
public Stream DownloadLog() throws OZAPIException
```

서버에 저장된 모니터 로그 파일을 가져옵니다.

OZAPIException

API를 사용할 때 생성되는 Exception입니다. 모든 API에서 OZAPIException을 발생시킵니다.

Property Detail

Message

```
public string Message
```

에러 내용

OZDataSetInfo

보고서에 추가된 데이터 셋 정보를 제공하는 클래스입니다.

Method Detail

GetDataSetServiceType

```
public string GetDataSetServiceType()
```

데이터 셋의 서비스 타입을 가져옵니다.

ODI, DataService, FXData 중 하나의 값으로 리턴됩니다.

GetDataSetName

```
public string GetDataSetName()
```

데이터 셋의 이름을 가져옵니다.

GetDataSetFields

```
public OZAttributeList GetDataSetFields()
```

데이터 셋의 필드 정보를 가져옵니다.

OZAttributeList는 fieldName과 fieldType으로 구성되어 있으며, fieldType은 JDBC Types과 동일합니다.
. 예를 들어, VARCHAR 타입인 경우 12가 리턴됩니다.

GetMaxRowCount

```
public int GetMaxRowCount()
```

데이터 셋의 "최대 행 수" 속성 값을 가져옵니다.

CSVDataSet, FxDataSet은 무조건 0으로 리턴됩니다.

GetUse

```
public bool GetUse()
```

데이터 셋의 "사용" 속성 값을 가져옵니다.

CSVDataSet, FxDataSet은 무조건 true로 리턴됩니다.

GetExportable

```
public bool GetExportable()
```

데이터 셋의 "셋 저장 가능" 속성 값을 가져옵니다.

OZODItem

ODI 정보를 제공하는 클래스입니다.

Method Detail

GetCategory

```
public string GetCategory()
```

카테고리 이름을 가져옵니다.

GetODIName

```
public string GetODIName()
```

ODI 이름을 가져옵니다.

GetODIFileName

```
public string GetODIFileName()
```

ODI 파일 이름을 가져옵니다.

GetODIParameters

```
public OZParameter[] GetODIParameters()
```

해당 ODI 파라미터를 가져옵니다.([OZParameter](#) 참고)

OZParameter

파라미터 정보를 제공하는 클래스입니다.

Method Detail

GetName

```
public string GetName()
```

파라미터 이름을 가져옵니다.

GetType

```
public int GetType()
```

파라미터 타입을 가져옵니다.

파라미터 타입은 JDBC Types과 동일합니다. 예를 들어, VARCHAR 타입인 경우 12가 리턴됩니다.

GetValue

```
public string GetValue()
```

파라미터 값을 가져옵니다.

GetDescription

```
public string GetDescription()
```

파라미터 설명을 가져옵니다.

OZReportParameter

보고서에 추가된 ODI 파라미터와 폼 파라미터 정보를 제공하는 클래스입니다.

Method Detail

GetODIList

```
public OZODIItem[] GetODIList()
```

ODI 정보 객체를 가져옵니다.([OZODIItem](#) 참고)

GetFormParameters

```
public OZParameter[] GetFormParameters()
```

폼 파라미터 정보 객체 가져옵니다.([OZParameter](#) 참고)

OZServerInfo

오즈 서버와 오즈 서버가 실행되고 있는 System의 버전 정보를 제공하는 클래스입니다.

Member Variables

OSName

```
public string OSName
```

오즈 서버가 실행되고 있는 System의 운영체제 이름

OSVersion

```
public string OSVersion
```

운영체제의 버전

FrameworkVersion

```
public string FrameworkVersion
```

.NET Framework 버전

FrameworkVendor

public string FrameworkVendor

.NET Framework 공급자

ServerVersion

public string ServerVersion

오즈 서버 버전

Service

오즈 서버의 서비스 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Service

public Service(string url, string id, string pw, bool autoLogin, bool useUSL)

서버 정보, 상태, 처리, 메모리 수집 등의 처리 기능을 수행합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

autoLogin - 자동 로그인 여부

useUSL - USL 사용 여부

Method Detail

GrabageCollect

public void GrabageCollect() throw OZCPEException

사용되지 않는 메모리를 수거해 재사용 가능하게 해주는 garbage collection을 실행합니다.

Ping

```
public bool Ping() throw OZCPEException
```

서버의 구동 여부를 반환합니다.

GetHandlerCount

```
public int GetHandlerCount() throw OZCPEException
```

서버의 동시 실행 개수를 가져옵니다.

Interface

OZ .NET Server API에서 제공하는 인터페이스는 아래와 같습니다.

- [IReportInfo](#)

IReportInfo

다중 보고서를 갖는 OZD 생성 시 보고서에 대한 부가 정보를 설정합니다.

Property Detail

ItemName

`string ItemName()`

아이템 이름

CategoryName

`string CategoryName()`

아이템의 카테고리 이름

Method Detail

AddURL

`void AddURL(params string[] urls)`

보고서에 포함된 이미지 파일의 URL을 배열로 설정합니다.

Parameters:

urls - 이미지 파일 URL

AddParameter

`void AddParameter(string key, string value)`

파라미터 값을 설정합니다.

Parameters:

key - 파라미터 이름

value - 파라미터 값

AddODIParameter

```
public void AddODIParameter(string odiName, string key, string value)
```

ODI 파라미터 값을 설정합니다.

Parameters:

odiName - ODI 이름

key - ODI 파라미터 이름

value - ODI 파라미터 값

AddProperties

```
public void AddProperties(string key, string value)
```

보고서 저장 옵션을 설정합니다.

Parameters:

key - 옵션 이름

value - 옵션 값([NameValueCollection](#) 참고)

AddFrameworkURL

```
void AddFrameworkURL(string fxName, string url, string charset)
```

FX 데이터의 프레임워크 URL을 설정합니다.

Parameters:

fxName - FX 데이터 이름

url - 프레임워크 URL

charset - 프레임워크 URL의 파라미터를 처리할 문자 셋. 문자 셋을 null 또는 공백으로 설정한 경우 시스템의 기본 문자 셋으로 적용됩니다. 문자 셋에 base64(대소문자 구분 안 함)를 포함하여 설정한 경우 예를 들어, base64/ksc5601 또는 base64/utf-8 등의 형태로 설정한 경우 ksc5601 또는 utf-8로 인코딩한 후 base64로 한 번 더 인코딩합니다.

OZ Scheduler API

OZ Scheduler API는 다양한 [Class](#)를 제공합니다.

Class

OZ Scheduler API에서 제공하는 클래스는 아래와 같습니다.

- [DirectExportResult](#)
- [DirectPrintResult](#)
- [DirectTaskResult](#)
- [FileInfo](#)
- [Program](#)
- [Publisher](#)
- [Scheduler](#)
- [ServerInfo](#)
- [TaskHolidayGroupInfo](#)
- [TaskHolidayInfo](#)
- [TaskResult](#)

DirectExportResult

익스포트 태스크 실행 결과를 나타냅니다.

Member Variables

ExportFileList

```
public string ExportFileList
```

익스포트된 파일 이름

PageCount

```
public int PageCount
```

보고서의 익스포트된 페이지 수

한 보고서를 여러 개의 포맷으로 동시에 익스포트할 경우 보고서의 익스포트된 페이지 수 합계를 나타냅니다. 예를 들어, 페이지 수가 4 페이지인 보고서를 xls, doc 포맷으로 익스포트 시 PageCount 값으로 8이 리턴됩니다.

보고서 익스포트 시 "viewer.largebundle=true"일 경우 PageCount 값으로 1이 리턴됩니다.

DirectPrintResult

프린트 태스크 실행 결과를 나타냅니다.

Member Variables

PageCount

```
public int PageCount
```

페이지 수

PageCopy

```
public int PageCopy
```

인쇄 매수

PageRange

```
public string PageRange
```

인쇄 범위

PrinterName

```
public string PrinterName
```

프린터 이름

PrinterDriverName

```
public string PrinterDriverName
```

프린터 드라이버 이름

DirectTaskResult

태스크 실행 결과를 나타냅니다.

Member Variables

TaskID

```
public string TaskID
```

태스크 ID

CompletedTime

```
public string CompletedTime
```

태스크 완료 시간(단위: 초)

ExecuteTime

```
public string ExecuteTime
```

태스크 수행 시간(단위: 초)

IsSuccessful

```
public bool IsSuccessful
```

태스크 성공 여부

FormName

```
public string FormName
```

보고서 이름

ErrorMessage

```
public string ErrorMessage
```

에러 메시지

FileInfo

파일 또는 폴더에 대한 기본 정보 값(파일/폴더, 이름, 크기, 최종 수정 시각)들을 가지고 있습니다.

Property Detail

IsDirectory

```
public bool IsDirectory
```

디렉토리인지 파일인지 구분

Name

```
public string Name
```

파일명 또는 디렉토리명

Size

```
public long Size
```

파일 크기 또는 디렉토리 크기

LastModified

```
public DateTime LastModified
```

파일 또는 디렉토리의 최종 수정 시각, 1970년 1월 1일 00:00:00 GMT를 기준으로 일시초로 시간을 계산한 값

Program

외부 프로그램 연동 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Program

```
public Program(string ip, int port)
```

오즈 스케줄러에서 외부 프로그램 등록 및 관리 기능을 수행합니다.

Parameters:

ip - 오즈 스케줄러 서버가 설치되어 있는 호스트 컴퓨터의 IP

port - 스케줄러 포트(기본 값: 9521)

Method Detail

CreateFolder

```
public void CreateFolder(ServerInfo s, string folder)
```

스케줄러에서 외부 프로그램의 폴더를 생성합니다.

Parameters:

s - 오즈 서버 정보

folder - 생성할 폴더명

DownloadFile

```
public Stream DownloadFile(ServerInfo s, string file)
```

등록된 외부 프로그램 파일을 다운로드합니다.

Parameters:

s - 오즈 서버 정보

file - 다운로드할 파일명

GetExternalProgramInfos

```
public FileInfo[] GetExternalProgramInfos(ServerInfo s, string folder)
```

스케줄러에서 외부 프로그램 목록을 가져옵니다.

Parameters:

s - 오즈 서버 정보

folder - 폴더명

RemoveFiles

```
public void RemoveFiles(ServerInfo s, string folder, string[] files)
```

등록된 외부 프로그램 파일을 삭제합니다.

Parameters:

s - 오즈 서버 정보

folder - 폴더명

files - 삭제할 프로그램 파일명

RemoveFolder

```
public void RemoveFolder(ServerInfo s, string folder, bool forciblyRemove)
```

스케줄러에서 외부 프로그램의 폴더를 삭제합니다.

Parameters:

s - 오즈 서버 정보

folder - 삭제할 폴더명

forciblyRemove - 하위 폴더가 존재할 경우 폴더 삭제 여부(true : 폴더를 삭제함. false : 폴더를 삭제하지 않음)

UploadFile

```
public void UploadFile(ServerInfo s, string file, Stream inputStream)
```

스케줄러 서버에 외부 프로그램을 등록합니다.

Parameters:

s - 오즈 서버 정보

file - 등록할 파일명

inputStream - 등록할 파일의 입력 스트림

Publisher

익스포트된 파일을 관리하는 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Publisher

```
public Publisher(string ip, int port)
```

오즈 스케줄러에 의해 생성되는 보고서 결과 파일을 관리합니다.

Parameters:

ip - 오즈 스케줄러 서버가 설치되어 있는 호스트 컴퓨터의 IP

port - 스케줄러 포트(기본 값: 9521)

Method Detail

CreateFolder

```
public void CreateFolder(ServerInfo s, string folder)
```

스케줄러 익스포트 파일을 저장할 폴더를 생성합니다.

Parameters:

s - 오즈 서버 정보
folder - 생성할 폴더명

DownloadFile

```
public Stream DownloadFile(ServerInfo s, string file)
```

등록된 스케줄러 익스포트 파일을 다운로드합니다.

Parameters:

s - 오즈 서버 정보
file - 폴더명

GetPublishedInfos

```
public FileInfo[] GetPublishedInfos(ServerInfo s, string folder)
```

스케줄러에서 익스포트 파일 목록을 가져옵니다.

Parameters:

s - 오즈 서버 정보
folder - 폴더명

RemoveFiles

```
public void RemoveFiles(ServerInfo s, string folder, string[] files)
```

등록된 스케줄러 익스포트 파일을 삭제합니다.

Parameters:

s - 오즈 서버 정보
folder - 삭제할 폴더명
files - 삭제할 파일명

RemoveFolder

```
public void RemoveFolder(ServerInfo s, string folder, bool forciblyRemove)
```

스케줄러 익스포트 파일이 저장되는 폴더를 삭제합니다.

Parameters:

s - 오즈 서버 정보
folder - 삭제할 폴더명

forciblyRemove - 하위 폴더가 존재할 경우 폴더 삭제 여부(true : 폴더를 삭제함. false : 폴더를 삭제하지 않음)

Scheduler

태스크 설정 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

Scheduler

```
public Scheduler(string ip, int port)
```

보고서 스케줄 작업을 생성하고 결과 파일 및 스케줄러 관리 기능을 수행합니다.

Parameters:

ip - 오즈 스케줄러 서버가 설치되어 있는 호스트 컴퓨터의 IP

port - 스케줄러 포트(기본 값: 9521)

Method Detail

CreateTask

```
public string CreateTask(ServerInfo s, NameValueCollection configMap,  
NameValueCollection exportMap)
```

폼 파일을 기반으로 태스크를 생성한 후 생성된 태스크 ID를 가져옵니다. CreateTask 함수로 여러 개의 태스크 생성 시 뷰어의 대기 큐와 상관없이 실행되는 뷰어의 개수만큼 Thread와 param 파일이 생성됩니다.

Parameters:

s - 오즈 서버 정보

configMap - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

exportMap - 익스포트 정보([NameValueCollection](#) 참고)

GetTaskInfos

```
public ScheduledTask[] GetTaskInfos(ServerInfo s)
```

현재 생성된 태스크의 결과물을 가져옵니다.

Parameters:

s - 오즈 서버 정보

GetTaskResult

```
public TaskResult[] GetTaskResult(ServerInfo s, string from, string to, string taskId, int rollingIndex)
```

롤링된 파일을 검색하여 태스크 실행 결과를 가져옵니다.

Parameters:

s - 오즈 서버 정보

from - 태스크 결과를 가져올 시작 시간

to - 태스크 결과를 가져올 종료 시간

taskId - 가져올 태스크 아이디

rollingIndex - 파일 인덱스

GetTaskResults

```
public TaskResult[] GetTaskResults(ServerInfo s, string from, string to, string taskId)
```

태스크 결과를 가져옵니다.

Parameters:

s - 오즈 서버 정보

from - 태스크 결과를 가져올 시작 시간

to - 태스크 결과를 가져올 종료 시간

taskId - 가져올 태스크 아이디

GetTaskWaitTime

```
public int GetTaskWaitTime(ServerInfo s, String taskId)
```

태스크가 대기한 시간을 가져옵니다.(단위: 초)

Parameters:

s - 오즈 서버 정보

taskId - 가져올 태스크 아이디

GetTaskWaitCount

```
public int GetTaskWaitCount()
```

실행 대기 중인 태스크 개수를 가져옵니다.

RemoveTask

```
public void RemoveTask(ServerInfo s, string taskId)
```

태스크를 삭제합니다.

Parameters:

s - 오즈 서버 정보

taskId - 삭제할 태스크 아이디

ModifyTask

```
public string ModifyTask(ServerInfo s, string taskId, NameValueCollection  
configMap, NameValueCollection exportMap)
```

태스크 속성을 변경합니다.

Parameters:

s - 오즈 서버 정보

taskId - 속성을 변경할 태스크 아이디

configMap - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

exportMap - 익스포트 정보([NameValueCollection](#) 참고)

PauseTask

```
public void PauseTask(ServerInfo s, string taskId)
```

태스크를 일시 중지시킵니다.

Parameters:

s - 오즈 서버 정보

taskId - 중지할 태스크 아이디

ResumeTask

```
public void ResumeTask(ServerInfo s, string taskId)
```

중지된 태스크를 다시 실행시킵니다.

Parameters:

s - 오즈 서버 정보

taskId - 다시 실행시킬 태스크 아이디

Stop

```
public void Stop(ServerInfo s, bool forciblyStop)
```

스케줄러를 중지시킵니다.

Parameters:

s - 오즈 서버 정보

forciblyStop - 스케줄러를 강제로 중지시킬지 여부

Export

```
public bool Export(ServerInfo s, NameValueCollection configMap,  
NameValueCollection exportMap)
```

뷰어 파라미터를 그대로 사용하여 익스포트한 후 익스포트 성공 여부를 반환합니다.

서버에 설정된 스케줄러의 환경 설정 중 "ViewerType=None"이 아닌 경우에만 사용할 수 있습니다.

Parameters:

s - 오즈 서버 정보

configMap - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

exportMap - 익스포트 정보([NameValueCollection](#) 참고)

MakePDF

```
public bool MakePDF(ServerInfo s, NameValueCollection configMap,  
NameValueCollection exportMap)
```

스케줄러의 작업 목록에 등록하지 않고 PDF 익스포트만 수행한 후 익스포트 성공 여부를 반환합니다. 서버에 설정된 스케줄러의 환경 설정 중 "ViewerType=None"이 아닌 경우에만 사용할 수 있습니다.

Parameters:

s - 오즈 서버 정보

configMap - 익스포트를 위한 폼 정보([NameValueCollection](#) 참고)

exportMap - PDF 익스포트 정보([NameValueCollection](#) 참고)

Print

```
public bool Print(ServerInfo s, NameValueCollection configMap, NameValueCollection  
printMap)
```

뷰어 파라미터를 적용시켜 프린트하고 성공 여부를 반환합니다.

실제 프린트 작업의 성공 여부가 아니라 뷰어에 프린트 호출이 성공했는지 여부를 반환합니다.

"task_type=viewerTag"인 경우에만 사용될 수 있으며 이외의 값이 들어오면 무시합니다. 또한 서버에 설정된 스케줄러의 환경 설정 중 "ViewerType=None"이 아닌 경우에만 사용할 수 있습니다.

태스크 작업 주기 옵션과 관계없이 즉시 한 번만 프린트 작업을 호출하고 태스크는 생성되지 않으며, 메일 보내기 등의 파라미터를 설정하여도 메일 관련 작업은 일어나지 않고 오직 프린트만 합니다.

뷰어 파라미터에서 설정한 파라미터 중 아래의 파라미터는 항상 해당 값으로 고정되어 동작합니다.

- viewer.allowmultiframe=true
- viewer.mode=print
- viewer.printcommand=true
- viewer.showerrorMessage=false
- viewer.useprogressbar=false
- print.ignoreerror=false
- print.mode=silent
- export.confirmsave=false
- export.format=""
- information.debug=debug

Parameters:

s - 오즈 서버 정보

configMap - 스케줄러 설정 정보([NameValueCollection](#) 참고)

printMap - 출력 정보([NameValueCollection](#) 참고)

GetCurrentStatus

```
string GetCurrentStatus()
```

스케줄러의 사용 메모리, 전체 메모리, 실행 중인 뷰어 개수, 대기 중인 뷰어 개수를 가져옵니다.

GetConfiguration

```
public NameValueCollection GetConfiguration(ServerInfo s)
```

스케줄러 설정 값을 가져옵니다.

Parameters:

s - 오즈 서버 정보

ModifyConfiguration

```
public void ModifyConfiguration(ServerInfo s, NameValueCollection configMap)
```

스케줄러의 설정 값을 변경합니다.

Parameters:

s - 오즈 서버 정보

configMap - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

SetReceiveTimeout

```
public void SetReceiveTimeout(int timeOut)
```

소켓 타임 아웃을 설정합니다.(단위: 초)

Parameters:

timeOut - 소켓 타임 아웃

Ping

```
public bool Ping()
```

서버의 구동 여부를 반환합니다.

GetSchedulingInfos

```
public string[] GetSchedulingInfos(string path)
```

설정한 경로에 있는 OZS 파일 목록을 가져옵니다.

path를 파일명으로 지정하면 해당 OZS 파일 정보만 가져오며, path를 폴더명으로 지정하면 해당 폴더에 있는 모든 OZS 파일 정보를 가져옵니다. 만일 해당 경로에 파일이 존재하지 않거나 존재하지 않는 폴더를 지정하면 빈 값을 리턴합니다.

Parameters:

path - OZS 파일 목록을 가져올 경로. OZS 파일 목록을 가져올 경로는 /%SCH_HOME%/[path]/로 지정됩니다. 예를 들어 path를 sample로 설정할 경우에는 /%SCH_HOME%/sample/에 있는 OZS 파일 목록을 가져옵니다.

ConvertSchedulingInfos

```
public void ConvertSchedulingInfos(string oldPath, string newPath)
```

2.5 버전의 OZS 파일을 최신 버전으로 변경하여 저장합니다.

Parameters:

oldPath - 변환할 OZS 파일명 또는 경로. 해당 경로는 /%SCH_HOME%/[oldPath]/입니다.

newPath - 변환된 OZS 파일 저장 경로. 해당 경로는 /%SCH_HOME%/[newPath]/이며 newPath는 oldPath와 다르게 설정하여야 합니다.

AddTaskHoliday

```
public bool AddTaskHoliday(TaskHolidayInfo value)
```

태스크 휴일 정보를 추가합니다.

Parameters:

value - 태스크 휴일 정보

ModifyTaskHoliday

```
public bool ModifyTaskHoliday(string old_key, TaskHolidayInfo new_value)
```

태스크 휴일 정보를 수정합니다.

Parameters:

old_key - 수정할 태스크 휴일 이름

new_value - 새로운 태스크 휴일 정보

DeleteTaskHoliday

```
public bool DeleteTaskHoliday(string[] keys)
```

태스크 휴일 정보를 삭제합니다.

Parameters:

keys - 삭제할 태스크 휴일 이름 배열

AddTaskHolidayGroup

```
public bool AddTaskHolidayGroup(TaskHolidayGroupInfo value)
```

태스크 휴일 그룹 정보를 추가합니다.

Parameters:

value - 태스크 휴일 그룹 정보

ModifyTaskHolidayGroup

```
public bool ModifyTaskHolidayGroup(string old_key, TaskHolidayGroupInfo new_value)
```

태스크 휴일 그룹 정보를 수정합니다.

Parameters:

old_key - 수정할 태스크 휴일 그룹 이름

new_value - 새로운 태스크 휴일 그룹 정보

DeleteTaskHolidayGroup

```
public bool DeleteTaskHolidayGroup(string key)
```

태스크 휴일 그룹 정보를 삭제합니다.

Parameters:

key - 삭제할 태스크 휴일 그룹 이름

GetTaskHolidayInfos

```
public TaskHolidayInfos GetTaskHolidayInfos()
```

태스크 휴일 정보 목록을 가져옵니다.

GetTaskHolidayGroupInfos

```
public TaskHolidayGroupInfos GetTaskHolidayGroupInfos()
```

태스크 휴일 그룹 정보 목록을 가져옵니다.

SaveTaskHoliday

```
public void SaveTaskHoliday()
```

태스크 휴일 정보를 xml 파일로 저장합니다.

DirectExport

```
public DirectExportResult DirectExport(ServerInfo s, NameValueCollection  
configMap, NameValueCollection exportMap)
```

뷰어 파라미터를 적용시켜 익스포트한 후 결과 정보를 반환합니다.

서버에 설정된 스케줄러의 환경 설정 중 "ViewerType=None"이 아닌 경우에만 사용할 수 있습니다.

Parameters:

s - 오즈 서버 정보

configMap - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

exportMap - 익스포트 정보([NameValueCollection](#) 참고)

DirectExportFiles

```
public Hashtable[] DirectExportFiles(ServerInfo s, NameValueCollection properties,  
NameValueCollection exportProperties, string localTempPath)
```

뷰어 파라미터를 적용시켜 익스포트한 결과를 로컬 임시 경로에 저장한 후 파일 정보를 반환합니다.
서버에 설정된 스케줄러의 환경 설정 중 "ViewerType=None"이 아닌 경우에만 사용할 수 있습니다.

Parameters:

s - 오즈 서버 정보

properties - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

exportProperties - 익스포트 정보([NameValueCollection](#) 참고)

localTempPath - 익스포트한 결과를 저장할 로컬 임시 경로

DirectExportResult

```
public DirectExportResult DirectExportResult(ServerInfo s, NameValueCollection  
properties, NameValueCollection exportProperties)
```

뷰어 파라미터를 적용시켜 익스포트한 후 결과 정보를 반환합니다.

서버에 설정된 스케줄러의 환경 설정 중 task_type이 viewerTag이고, ViewerType이 ActiveX 또는 Applet인 경우에만 사용할 수 있습니다.

Parameters:

s - 오즈 서버 정보

properties - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

exportProperties - 익스포트 정보([NameValueCollection](#) 참고)

DirectExportByteArray

```
public Hashtable DirectExportByteArray(ServerInfo s, NameValueCollection  
configProperties, NameValueCollection exportProperties)
```

태스크를 실행한 후 실행 결과 정보를 반환합니다.

태스크 실행 결과를 파일로 저장하지 않고, 메모리로 익스포트하려면 ude.classname을 설정한 후 ViewerType=Applet, export_file=false로 설정하시기 바랍니다.

서버에 설정된 스케줄러의 환경 설정 중 "ViewerType=None"이 아닌 경우에만 사용할 수 있습니다.

Parameters:

s - 오즈 서버 정보

configProperties - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

exportProperties - 익스포트 정보([NameValueCollection](#) 참고)

DirectPrint

```
public DirectPrintResult DirectPrint(ServerInfo s, NameValueCollection configMap,
NameValueCollection printMap)
```

뷰어 파라미터를 적용시켜 프린트하고 결과 정보를 반환합니다.

"task_type=viewerTag"인 경우에만 사용될 수 있으며 이외의 값이 들어오면 무시합니다. 또한 서버에 설정된 스케줄러의 환경 설정 중 "ViewerType=None"이 아닌 경우에만 사용할 수 있습니다.

Parameters:

s - 오즈 서버 정보

configMap - 스케줄러 설정 옵션([NameValueCollection](#) 참고)

printMap - 출력 정보. print.mode = silent로만 실행됩니다. ([NameValueCollection](#) 참고)

GetFile

```
public void GetFile(ServerInfo s, string fileName, string targetFile)
```

파일을 가져옵니다.

Parameters:

s - 오즈 서버 정보

fileName - 가져올 파일명

targetFile - 저장할 파일명

GetTaskProperties

```
public void GetTaskProperties(ServerInfo s, string TaskID, NameValueCollection
configMap, NameValueCollection exportMap)
```

태스크의 속성 즉 스케줄러에서 태스크를 생성할 때 사용되는 설정 옵션(configMap)과 스케줄러에서 익스포트되는 파일의 설정 옵션을 가져옵니다.

Parameters:

s - 오즈 서버 정보

taskID - 속성을 가져올 태스크 아이디

configMap - 익스포트를 위한 폼 정보([NameValueCollection](#) 참고)

exportMap - PDF 익스포트 정보([NameValueCollection](#) 참고)

Scheduler > NameValueCollection

서버에 설정된 스케줄러의 환경 설정 중 "ViewerType"을 "None"으로 설정한 경우에는 익스포트 시에 뷰어를 사용하지 않고, 서버 API를 이용하여 익스포트하므로, OZD 파일로만 익스포트할 수 있습니다.

아래 파라미터는 항상 고정된 값으로만 동작됩니다.

- viewer.mode=export
- viewer.useprogressbar=false
- viewer.allowmultiframe=true
- export.mode=silent
- export.confirmsave=false
- information.debug=debug
- viewer.showerrormessage=false

-
- [Task Option](#)
 - [EMail Option](#)
 - [Parameter Option](#)
 - [Scheduler Server Option](#)
 - [Scheduling Time Option](#)
 - etc.

Scheduler > NameValueCollection > Task Option

리포트 이름 설정

Key	설명
report_name	태스크 생성할 리포트명
category_name	태스크를 생성할 리포트의 카테고리명

서버 데이터 모듈

Key	설명
dm_server_check	서버 데이터 모듈 선택 check, null 중 하나의 값으로 설정

Key	설명
dm_server_name	서버 데이터 모듈 파일명 Repository의 odi가 있는 category에 저장된 SDM 파일명
odi_name	서버 데이터 모듈을 만들 ODI명
odi_category_name	서버 데이터 모듈을 만들 ODI가 저장된 카테고리명 루트인 경우 "/" 로 입력

뷰어 파라미터 설정

Key	설명
task_type	뷰어 파라미터 그대로 사용 여부 설정 viewerTag로만 설정 이 옵션은 뷰어에서 사용하는 파라미터를 그대로 사용할 수 있도록 설정해 주는 옵션으로 , 주로 멀티 폼을 사용하는 보고서를 익스포트할 때 사용됩니다.

태스크 타임 아웃 설정

Key	설명
taskwait_timeout	태스크 실행 대기 타임아웃(단위: 초) 주기적으로 실행되는 태스크는 동작 안 함 scheduler_server.properties 파일의 ViewerConcurrentCount 옵션 값을 n으로 설정한 경우 동작 안 함

외부 프로그램 설정

Key	설명
external_program_check	파라미터 생성과 파일 경로의 동적 생성을 위한 외부 프로그램 체크 여부 check, null 중 하나의 값으로 설정
external_program_command	외부 프로그램명

Key	설명
	SCH_HOME/External"에서의 상대 경로

Scheduler > NameValueCollection > EMail Option

Key	설명
mail_check	전자 메일 발송 여부 check, null 중 하나의 값으로 설정
mail_notify_error_check	수신자에게 에러 통보 여부 선택 check, null 중 하나의 값으로 설정
mail_recipient_to	전자 메일 수신자
mail_recipient_cc	참조할 사람 메일
mail_recipient_bcc	숨은 참조할 사람 메일
mail_subject	전자 메일 제목
mail_text_message	전자 메일 내용
mail_html_comment	전자 메일 내용에 HTML 사용 여부 check, null 중 하나의 값으로 설정
html_mail_content	전자 메일 내용에 생성한 HTML 파일로 대체 check, null 중 하나의 값으로 설정
mail_attach_list	전자 메일에 첨부할 스케줄러의 생성된 파일 리스트 다수 선택이 가능하며 '/'로 구분
mail_attach_zipcheck	전자 메일에 첨부할 파일을 하나의 파일로 압축하여 전송

Key	설명
	check, null 중 하나의 값으로 설정
mail_attach_zipfilename	전자 메일에 첨부할 파일을 하나의 파일로 압축하여 전송 시 파일 이름
mail_attach_zippassword	전자 메일에 첨부할 파일을 하나의 파일로 압축하여 전송할 때 파일 암호

Scheduler > NameValueCollection > Parameter Option

Key	설명
odi_parameter_count	ODI 파라미터 수
odi_parameter_name_1 ... odi_parameter_name_n	ODI 파라미터명
odi_parameter_value_1 ... odi_parameter_value_n	ODI 파라미터 값
parameter_count	폼 파라미터 수 파라미터의 수는 폼과 각각의 ODI에서 사용하는 파라미터 수의 합을 입력 파라미터 수로 설정하고, 그 개수만큼 파라미터 이름과 해당 값을 설정해 주어야 합니다.
parameter_name_1 ... parameter_name_n	폼 파라미터명 리포트 파라미터명은 "[FORM]." 으로 시작합니다. 예를 들어 empNo라는 파라미터는 "[FORM].empNo"로 입력하여야 합니다. ODI 파라미터명은 ODI명으로 시작합니다. 예를 들어 ODI명이 "testodi"이고 파라미터명이 "id"일 경우에는 "testodi.id"로 입력하여야 합니다.

Key	설명
parameter_value_1 ... parameter_value_n	폼 파라미터 값
use_ozd_parameter	OZD 바인딩 시 OZD에 저장된 파라미터 값을 적용할지 여부 true, false 중 하나의 값으로 설정

Scheduler > NameValueCollection > Scheduler Server Option

Key	설명
SchedulerPort	스케줄러 포트 번호(기본 값: "9521")
SchedulingInfoFilePath	스케줄링 정보 파일 저장 경로명
SMTPServer	SMTP 서버 주소
SMTPServerProt	SMTP 서버 포트 번호
MailFrom	전자 메일 발송 주소
TempRepositoryFilePath	임시 파일 저장 경로
RepositoryFileRootPath	익스포트 파일 루트 저장 경로 설정
ExternalProgramFilePath	외부 프로그램 저장 경로 설정
ErrorNotifyToSender	관리자에게 에러 통보 여부 설정 true, false 중 하나의 값으로 설정

Scheduler > NameValueCollection > Scheduling Time Option

실행 타입

Key	설명
launch_type	실행 타입 once, immediately, periodically 중 하나의 값으로 설정

스케줄 시간(launch_type = once인 경우)

Key	설명
execution_year	실행 시간 - 년
execution_month	실행 시간 - 월
execution_day	실행 시간 - 일
execution_hour	실행 시간 - 시
execution_min	실행 시간 - 분

스케줄 시간(launch_type = periodically인 경우)

Key	설명
start_year	시작 시간 - 년
start_month	시작 시간 - 월
start_day	시작 시간 - 일
end_year	종료 시간 - 년

Key	설명
end_month	종료 시간 - 월
end_day	종료 시간 - 일
periodically_execution_day_type	주기 타입 daily, weekly, monthly 중 하나의 값으로 설정

스케줄 시간(periodically_execution_day_type = daily인 경우)

Key	설명
daily_type	평일
daily_every_days	설정된 일 간격

스케줄 시간(periodically_execution_day_type = weekly인 경우)

Key	설명
weekly_every_weeks	매주 몇번째 주
weekly_monday_check	매주 월요일 check, null 중 하나의 값으로 설정
weekly_tuesday_check	매주 화요일 check, null 중 하나의 값으로 설정
weekly_wednesday_check	매주 수요일 check, null 중 하나의 값으로 설정
weekly_thursday_check	매주 목요일 check, null 중 하나의 값으로 설정
weekly_friday_check	매주 금요일 check, null 중 하나의 값으로 설정

Key	설명
weekly_saturday_check	매주 토요일 check, null 중 하나의 값으로 설정
weekly_sunday_check	매주 일요일 check, null 중 하나의 값으로 설정

스케줄 시간((periodically_execution_day_type = monthly인 경우))

Key	설명
monthly_every_months	설정된 달 간격(몇 달)
monthly_type	특정 일이나 주 specific_day, day_of_week, user_defined 중 하나의 값으로 설정
monthly_days	1 ~ 31 : 매달 특정 일 LAST : 매달 마지막 일
monthly_which_week	매달 몇 번째 주 T1 : 매달 첫 번째 주 T2 : 매달 두 번째 주 T3 : 매달 세 번째 주 T4 : 매달 네 번째 주 T5 : 매달 다섯 번째 주 TL : 매달 마지막 주
monthly_which_week_day	매달 특정 요일 sunday, monday, tuesday, wednesday, thursday, friday, saturday 중 하나의 값으로 설정
monthly_user_defined_days	설정할 일을 오름차순으로 콤마(,)로 연결하여 설정

실행 횟수

Key	설명
periodically_execution_time_type	실행 횟수 once, repeat, user_defined 중 하나의 값으로 설정

실행 횟수(periodically_execution_time_type = once인 경우)

Key	설명
once_hour	실행시킬 시간 - 시
once_min	실행시킬 시간 - 분

실행 횟수(periodically_execution_time_type = repeat인 경우)

Key	설명
repeat_every_hours	주기적으로 실행 간격 - 시
repeat_every_minutes	주기적으로 실행 간격 - 분
repeat_start_hour	주기적 실행 시작 시간 - 시
repeat_start_minute	주기적 실행 시작 시간 - 분
repeat_end_hour	주기적 실행 종료 시간 - 시
repeat_end_minute	주기적 실행 종료 시간 - 분

실행 횟수(periodically_execution_time_type = user_defined인 경우)

Key	설명
user_defined_time	설정할 시간(시:분)을 오름차순으로 콤마(,)로 연결하여 설정

[Scheduler](#) > [NameValueCollection](#) > etc.

Key	설명
export_file	DirectExportByteArray 함수 실행 시 태스크 실행 결과를 파일로 저장할지 여부 true, false 중 하나의 값으로 설정 ActiveX 뷰어에서는 항상 true로 동작함
file_export_list	변환 파일 리스트 다수 선택이 가능하며 '/'로 구분

ServerInfo

오즈 서버에 대한 기본적인 정보를 가지고 있는 클래스입니다.

Property Detail

URL

`public string URL`

오즈 서버 URL. Server가 HTTP 방식일 경우에만 설정합니다.

ID

`public string ID`

사용자의 ID

Password

`public string Password`

사용자의 패스워드

DirectoryName

`public string DirectoryName`

경로

UserType

`public UserType UserType`

사용자 타입(Administrator, Normal, Everyone)

TaskHolidayGroupInfo

태스크 휴일 그룹 설정 기능과 관련된 속성과 메소드를 제공하는 클래스입니다.

Constructor Detail

TaskHolidayGroupInfo

```
public TaskHolidayGroupInfo(string name)
```

태스크에 지정할 휴일 그룹을 정의합니다.

Parameters:

name - 이름. 이름에 ";" 문자가 포함될 수 없습니다.

Property Detail

Name

```
public string Name
```

태스크 휴일 그룹 이름

References

```
public string[] References
```

태스크 휴일 그룹에서 참조할 Reference 이름의 배열

Method Detail

AddReference

```
public void AddReference(string[] names)
```

태스크 휴일 그룹이 참조하는 태스크 휴일 이름을 등록합니다.

Parameters:

names - 태스크 휴일 이름. 동일한 이름이 있을 경우 해당 값 무시함

RemoveReferences

```
public void RemoveReferences(string[] names)
```

태스크 휴일 그룹에서 참조하는 태스크 휴일 이름을 삭제합니다.

Parameters:

names - 태스크 휴일 그룹에서 삭제할 태스크 휴일 이름의 배열

ClearReferences

```
public void ClearReferences()
```

태스크 휴일 그룹에서 참조하는 태스크 휴일 이름을 모두 삭제합니다.

TaskHolidayInfo

태스크 휴일 설정 기능과 관련된 속성을 제공하는 클래스입니다.

Constructor Detail

TaskHolidayInfo

```
public TaskHolidayInfo(string name)
```

태스크를 실행하지 않을 휴일을 정의합니다.

Parameters:

name - 이름. 이름에 ";" 문자가 포함될 수 없습니다.

Property Detail

Name

```
public string Name
```

태스크 휴일 이름

BaseDay

`public string BaseDay(string pattern)`

태스크 휴일로 지정할 기준일

Parameters:

pattern - 기준일의 패턴.

Type

`public string Type`

태스크 휴일 등록 시 지정한 타입

solar, lunar 중 하나의 값으로 설정

Start

`public int Start`

"baseday"를 기준으로 며칠 전부터 휴일로 지정할지에 대한 시작 값

End

`public int End`

"baseday"를 기준으로 며칠 후까지 휴일로 지정할지에 대한 값

TaskResult

태스크 실행 결과를 나타냅니다.

Property Detail

TaskID

`public string TaskID`

태스크 ID

CompletedTime

`public string CompletedTime`

태스크 완료 시간

IsSuccessful

`public bool IsSuccessful`

태스크 성공 여부

FormFileName

`public string FormFileName`

보고서 이름

ParamInfo

`public string ParamInfo`

파라미터

SchedulingType

`public string SchedulingType`

스케줄 타입

ExportedFiles

`public string ExportedFiles`

익스포트된 파일 이름

ErrorMessage

`public string ErrorMessage`

에러 발생 시 에러 메시지

OZ Repository API

OZ Repository API는 다양한 [Class](#)와 [Interface](#)를 제공합니다.

Class

OZ Repository API에서 제공하는 클래스는 아래와 같습니다.

- [RepositoryEX](#)
- [RepositoryException](#)

RepositoryEX

리포지토리 기능과 관련된 메소드를 제공하는 클래스입니다.

Constructor Detail

RepositoryEx

`public RepositoryEx(string url, string id, string pw, bool useUSL) throws OZCPEException`

리포지토리 기능을 수행합니다.

Parameters:

url - HTTP 방식 오즈 서버의 URL

id - 사용자 아이디

pw - 사용자 패스워드

useUSL - USL 사용 여부

Method Detail

RepositoryEX 클래스에서 제공하는 메소드는 아래와 같습니다.

- [Configuring the repository](#)
- [Creating groups and users](#)
- [Editing groups and users](#)
- [Deleting groups and users](#)
- [Getting groups and users](#)

- [Setting up group and user permissions](#)
- [Getting group and user permissions](#)
- [Creating categories and items](#)
- [Editing categories and items](#)
- [Deleting categories and items](#)
- [Getting categories and items](#)
- [Setting up category and item permissions](#)
- [Getting category and item permissions](#)
- [Using item version control](#)
- [Distribute items](#)
- [Getting error](#)

Configuring the repository

GetConfiguration

```
public NameValueCollection GetConfiguration()
```

리포지토리 설정을 가져옵니다.

SetConfiguration

```
public void SetConfiguration(NameValueCollection conf)
```

리포지토리 설정을 변경합니다.

Parameters:

conf - 변경할 설정

Reload

```
public void Reload()
```

리포지토리를 재시작합니다.

Creating groups and users

CreateGroup

```
public string CreateGroup(string groupName, string parentGroupID, string description)
```

새로운 그룹을 생성하고 생성된 그룹의 ID를 반환합니다.

Parameters:

groupName - 새로 생성할 그룹의 이름

parentGroupID - 새로 생성할 그룹의 상위 그룹 ID

description - 그룹에 대한 설명 내용

CreateUser

```
public string CreateUser(string userName, string password, string description)
```

새로운 사용자를 생성하고 생성된 사용자의 ID를 반환합니다.

Parameters:

userName - 사용자 이름

password - 패스워드

description - 사용자에게 대한 설명 내용

CreateUserInGroup

```
public string CreateUserInGroup(string uName, string pwd, string gID, string desc)  
throws OZCPEException
```

지정한 그룹에 새로운 사용자를 생성하고 생성된 사용자의 ID를 반환합니다.

Parameters:

uName - 새로 생성할 사용자의 이름

Pwd - 패스워드

gID - 사용자를 생성할 그룹 ID

desc - 사용자에게 대한 설명 내용

Editing groups and users

ModifyGroupName

```
public string ModifyGroupName(string groupID, string groupName)
```

지정한 사용자 ID에 해당하는 사용자의 이름을 변경하고 변경 성공 여부를 반환합니다.

Parameters:

groupID - 그룹의 이름을 변경할 그룹 ID

groupName - 변경할 그룹의 이름

ModifyGroupDescription

```
public bool ModifyGroupDescription(string groupID, string description)
```

지정한 사용자 ID에 해당하는 사용자의 설명을 변경하고 변경 성공 여부를 반환합니다.

Parameters:

groupID - 그룹의 설명을 변경할 그룹 ID

description - 변경할 그룹의 설명

ModifyUserName

```
public string ModifyUserName(string userID, string userName)
```

지정한 사용자 ID에 해당하는 사용자의 이름을 변경하고 변경 성공 여부를 반환합니다.

Parameters:

userID - 사용자 ID

userName - 변경할 사용자 이름

ModifyUserPassword

```
public bool ModifyUserPassword(string userID, string oldPwd, string newPwd)
```

지정한 사용자 ID에 해당하는 사용자의 패스워드를 변경하고 변경 성공 여부를 반환합니다.

Parameters:

userID - 사용자 ID

oldPwd - 변경 전 패스워드

newPwd - 변경 후 패스워드

ModifyUserDescription

```
public bool ModifyUserDescription(string userID, string description)
```

지정한 사용자 ID에 해당하는 사용자의 설명 내용을 변경하고 변경 성공 여부를 반환합니다.

Parameters:

userID - 사용자 ID

description - 변경할 설명 내용

TransferGroup

```
public bool TransferGroup(string groupID, string targetGroupID)
```

지정한 그룹을 다른 그룹으로 이동하고 그룹 이동 성공 여부를 가져옵니다.

Parameters:

groupID - 그룹을 이동할 그룹 ID

targetGroupID - 이동할 그룹 ID

TransferUser

```
public bool TransferUser(string userID, string newGroupID)
```

지정한 사용자의 그룹을 이동하고 그룹 이동 성공 여부를 가져옵니다.

Parameters:

userID - 그룹을 이동할 사용자 ID

newGroupID - 이동할 그룹 ID

Deleting groups and users

DeleteGroup

```
public bool DeleteGroup(string groupID)
```

지정한 그룹 ID에 해당하는 그룹을 삭제하고 삭제 성공 여부를 반환합니다.

Parameters:

groupID - 삭제할 그룹 ID

DeleteUser

```
public bool DeleteUser(string userID)
```

지정한 사용자 ID에 해당하는 사용자의 모든 정보를 삭제하고 삭제 성공 여부를 반환합니다.

Parameters:

userID - 삭제할 사용자의 ID

Getting groups and users

GetGroupID

```
public string GetGroupID(string userID)
```

지정한 사용자가 속해 있는 그룹의 정보를 가져옵니다.

Parameters:

userID - 그룹 정보를 가져올 사용자 ID

GetGroupInfo

```
public IGroupInfo GetGroupInfo(string groupID)
```

지정한 그룹 ID에 해당하는 그룹의 정보를 가져옵니다.

Parameters:

groupID - 그룹 정보를 가져올 그룹 ID

GetGroupInfoListInGroup

```
public IGroupInfo[] GetGroupInfoListInGroup(string groupID, bool isRecursive)
```

그룹에 추가된 그룹 정보를 가져옵니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

groupID - 그룹 ID(Full Path로 설정)

isRecursive - 하위 그룹 포함 여부

GetSubGroupInfos

```
public IGroupInfo[] GetSubGroupInfos(string groupID)
```

지정한 그룹의 하위 그룹 정보를 가져옵니다.

Parameters:

groupID - 하위 그룹의 정보를 가져올 그룹 ID

GetParentGroupInfo

```
public IGroupInfo GetParentGroupInfo(string groupID)
```

지정한 그룹의 상위 그룹의 정보를 가져옵니다.

Parameters:

groupID - 상위 그룹의 정보를 가져올 그룹 ID

GetUserInfo

```
public IUserInfo GetUserInfo(string userID)
```

지정한 사용자 ID에 해당하는 사용자의 사용자 정보를 가져옵니다.

Parameters:

userID - 사용자 정보를 가져올 사용자 ID

GetUserInfos

```
public IUserInfo[] GetUserInfos()
```

모든 사용자의 정보를 가져옵니다.

GetUserInfos

```
public IUserInfo[] GetUserInfos(string groupID)
```

지정한 그룹 ID에 등록되어 있는 모든 사용자의 정보를 가져옵니다.

Parameters:

groupID - 사용자의 정보를 가져올 그룹 ID

GetUserInfoListInGroup

```
public IUserInfo[] GetUserInfoListInGroup(string groupID, bool isRecursive)
```

그룹에 추가된 사용자 정보를 가져옵니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

groupID - 사용자의 정보를 가져올 그룹 ID

isRecursive - 하위 그룹 사용자 포함 여부

CheckPassword

```
public bool CheckPassword(string userID, string password)
```

사용자의 비밀번호가 맞는지 여부를 체크합니다.

Parameters:

userID - 비밀번호를 체크할 사용자 ID

password - 체크할 비밀번호

Setting up group and user permissions

AddGroupAdministrator

```
public bool AddGroupAdministrator(string userID, string groupID)
```

지정한 그룹의 그룹 관리자를 추가하고 추가 성공 여부를 반환합니다.

Parameters:

userID - 그룹 관리자로 추가할 사용자 ID

groupID - 그룹 관리자를 추가할 그룹 ID

RemoveGroupAdministrator

```
public bool RemoveGroupAdministrator(string userID, string groupID)
```

지정한 그룹의 그룹 관리자를 해제하고 해제 성공 여부를 반환합니다.

Parameters:

userID - 그룹 관리자 권한을 해제할 사용자 ID

groupID - 그룹 관리자를 해제할 그룹 ID

AddAdministrator

```
public bool AddAdministrator(string userID)
```

지정한 사용자 ID를 관리자로 설정합니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

userID - 관리자로 설정할 사용자 ID

RemoveAdministrator

```
public bool RemoveAdministrator(string userID)
```

지정한 사용자 ID에게 설정된 관리자 권한을 삭제합니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

userID - 관리자 권한을 삭제할 사용자 ID

EnableLogin

```
public void EnableLogin(string userID)
```

지정한 사용자 ID의 사용자를 로그인 가능하도록 설정합니다.

Parameters:

userID - 로그인 가능하도록 할 사용자 ID

DisableLogin

```
public void DisableLogin(string userID)
```

지정한 사용자 ID의 사용자를 로그인하지 못하도록 설정합니다.

Parameters:

userID - 로그인하지 못하도록 할 사용자 ID

Getting group and user permissions

GetGroupAdministrators

```
public IUserInfo[] GetGroupAdministrators(string groupID)
```

지정한 그룹의 그룹 관리자 정보를 가져옵니다.

Parameters:

groupID - 그룹 관리자 정보를 가져올 그룹 ID

IsGroupAdministrator

```
public bool IsGroupAdministrator(string userID, string groupID)
```

지정한 사용자 ID에 해당하는 사용자가 해당 그룹의 관리자인지 여부를 확인합니다.

Parameters:

userID - 관리자인지 체크할 사용자 ID

groupID - 그룹 ID

IsAdministrator

```
public bool IsAdministrator(string userID)
```

지정한 사용자 ID가 관리자인지 여부를 확인합니다.

Parameters:

userID - 관리자인지 체크할 사용자 ID

IsLoginEnabled

```
public bool IsLoginEnabled(string userID)
```

지정한 사용자 ID의 사용자가 로그인 가능한지 여부를 가져옵니다.

Parameters:

userID - 로그인 가능 여부를 체크할 사용자 ID

Creating categories and items

CreateCategories

```
public string[] CreateCategories(string[] categoryNames, string[]  
parentCategoryIDs, string comment)
```

카테고리를 새로 생성하고 생성된 카테고리의 ID를 반환합니다.

Parameters:

categoryName - 생성할 카테고리의 이름

parentCategoryIDs - 생성할 카테고리의 상위 카테고리 ID

comment - 주석문

CreateItem

```
public string[] CreateItem(string[] itemNames, string[] descriptions, string[]  
categoryIDs, bool[] isCompressed, Stream[] items, string comment, long[]  
createdTimes)
```

시간을 지정하여 아이템을 생성하고 생성된 아이템 ID를 반환합니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

itemNames - 새로 생성할 아이템의 이름

descriptions - 새로 생성할 아이템의 설명 내용

categoryIDs - 새로 생성할 아이템이 카테고리 이름

isCompressed - 압축 여부

items - 새로 생성할 아이템의 입력 스트림

comment - 주석문

createdTimes - 설정할 아이템 생성시간. 설정할 시간이 0, 음수이면 현재시간으로 설정

CreateItems

```
public string[] CreateItems(string[] itemNames, string[] descriptions, string[] categoryIDs, bool[] isCompressed, Stream[] items, string comment)
```

아이템을 생성하고 생성된 아이템 ID를 반환합니다.

Parameters:

itemNames - 새로 생성할 아이템의 이름
descriptions - 새로 생성할 아이템의 설명 내용
categoryIDs - 새로 생성할 아이템이 카테고리 이름
isCompressed - 압축 여부
items - 새로 생성할 아이템의 입력 스트림
comment - 주석문

Editing categories and items

ModifyCategoryName

```
public string ModifyCategoryName(string categoryID, string newCategoryName, string comment)
```

지정한 카테고리 ID에 해당하는 카테고리의 이름을 변경합니다.

Parameters:

categoryID - 이름을 변경할 카테고리의 ID
newCategoryName - 변경할 카테고리 이름
comment - 주석문

ModifyCategoryDesc

```
public bool ModifyCategoryDesc(string categoryID, string description)
```

카테고리 설명을 변경합니다.

리포지토리 타입이 NONE인 경우에는 REPOSITORY_NONE_BACKUP=true인 경우에만 동작합니다.

Parameters:

categoryID - 카테고리 ID(Full Path로 설정)
description - 카테고리 설명

ModifyItemName

```
public string ModifyItemName(string itemID, string newItemName, string comment)
```

아이템 이름을 변경하고 변경된 아이템 ID를 반환합니다.

Parameters:

itemID - 이름을 변경할 아이템 ID

newItemName - 변경할 아이템 이름

comment - 주석문

ModifyItemDescription

```
public bool ModifyItemDescription(string itemID, string description)
```

지정한 아이템 ID에 해당하는 아이템의 설명을 변경하고 변경 성공 여부를 반환합니다.

Parameters:

itemID - 설명을 변경할 아이템의 ID

description - 아이템 설명 내용

ModifyItemDatetime

```
public bool ModifyItemDatetime(string itemID, long datetime)
```

아이템 생성 시간을 변경합니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

itemID - 시간을 변경할 아이템의 ID

datetime - 변경할 시간. 0 또는 음수로 설정할 경우 현재 시간으로 적용됨

TransferCategory

```
public bool TransferCategory(string categoryID, string targetCategoryID)
```

지정한 카테고리를 다른 카테고리로 이동하고 카테고리 이동 성공 여부를 가져옵니다.

Parameters:

categoryID - 카테고리를 이동할 카테고리 ID

targetCategoryID - 이동할 카테고리 ID

TransferItems

```
public bool TransferItems(string[] itemIDs, string targetCategoryID)
```

지정한 아이템의 카테고리를 이동하고 카테고리 이동 성공 여부를 가져옵니다.

Parameters:

itemIDs - 카테고리를 이동할 아이템 ID

targetCategoryID - 이동할 카테고리 ID

EncryptFile

```
public OZAttributeList EncryptFile(string[] itemIDs)
```

OZR 파일을 암호화하여 저장하고, 암호화를 실패한 파일 리스트를 리턴합니다.

이미 암호화된 아이템은 다시 암호화하지 않습니다.

Parameters:

itemIDs - 암호화할 아이템 ID(Full Path로 설정)

EncryptFile

```
public OZAttributeList EncryptFile(string categoryID, bool isRecursive)
```

OZR 파일을 암호화하여 저장하고, 암호화를 실패한 파일 리스트를 리턴합니다.

이미 암호화된 아이템은 다시 암호화하지 않습니다.

Parameters:

categoryID - 암호화할 아이템을 가진 카테고리 ID(Full Path로 설정)

isRecursive - 하위 카테고리 포함 여부

DecryptFile

```
public OZAttributeList DecryptFile(string[] itemIDs)
```

OZR 파일을 복호화하여 저장하고, 복호화를 실패한 파일 리스트를 리턴합니다.

이미 복호화된 아이템은 다시 복호화하지 않습니다.

Parameters:

itemIDs - 복호화할 아이템 ID(Full Path로 설정)

DecryptFile

```
public OZAttributeList DecryptFile(string categoryID, bool isRecursive)
```

OZR 파일을 복호화하여 저장하고, 복호화를 실패한 파일 리스트를 리턴합니다.

이미 복호화된 아이템은 다시 복호화하지 않습니다.

Parameters:

categoryID - 복호화할 아이템을 가진 카테고리 ID(Full Path로 설정)

isRecursive - 하위 카테고리 포함 여부

SetUserDefinedResourceSync

```
public int SetUserDefinedResourceSync(string alias, string[] itemIDs, string comment, string[] errorMsg)
```

보고서를 사용자가 정의한 리소스로 동기화합니다.

Parameters:

alias - ozudr.properties에서 설정한 앨리어스 이름

itemIDs - 보고서 전체 경로

comment - 주석

errorMsg - 에러 메시지

Deleting categories and items

DeleteCategories

```
public bool[] DeleteCategories(string[] categoryIDs, bool[] toBeDestroyed, string comment)
```

지정한 카테고리 ID에 해당하는 카테고리를 삭제합니다.

Parameters:

categoryIDs - 삭제할 카테고리의 ID

toBeDestroyed - 카테고리를 영구 삭제할 지 여부

comment - 주석문

UnDeleteCategories

```
public bool[] UnDeleteCategories(string[] categoryIDs, string comment)
```

삭제된 카테고리를 복원하고 복원 성공 여부를 가져옵니다. 카테고리 삭제 시 "toBeDestroyed=false"로 설정하였을 경우 삭제된 카테고리의 복원이 가능합니다.

Parameters:

categoryIDs - 삭제된 카테고리의 ID

comment - 주석문

DeleteItems

```
public bool[] DeleteItems(string[] itemIDs, bool[] toBeDestroyed, string comment)
```

지정한 아이템을 리파지토리에서 삭제하고 아이템 삭제 성공 여부를 가져옵니다.

Parameters:

itemIDs - 삭제할 아이템 ID

toBeDestroyed - 아이템을 영구 삭제할 지 여부

comment - 주석문

UnDeleteItems

```
public bool[] UndeleteItems(string[] itemIDs, string comment)
```

삭제된 아이템을 복원하고 복원 성공 여부를 가져옵니다. 아이템 삭제 시 "toBeDestroyed=false"로 설정하였을 경우 삭제된 아이템의 복원이 가능합니다.

Parameters:

itemIDs - 복원할 아이템 ID

comment - 주석문

Getting categories and items

GetCategoryID

```
public string GetCategoryID(string itemID)
```

지정한 아이템이 존재하는 카테고리의 ID를 반환합니다.

Parameters:

itemID - 카테고리의 ID를 얻을 아이템 ID

GetCategoryInfo

```
public ICategoryInfo GetCategoryInfo(string categoryID)
```

지정한 카테고리의 정보를 가져옵니다.

Parameters:

categoryID - 정보를 가져올 카테고리 ID

GetCategoryInfos

```
public ICategoryInfo[] GetCategoryInfos(string categoryID)
```

지정한 카테고리에 속해 있는 모든 하위 카테고리 정보를 가져옵니다.

Parameters:

categoryID - 아이템의 개수를 얻을 카테고리의 ID

GetItem

```
public Stream GetItem(string itemID, int version, bool isCompressed)
```

지정한 아이템 ID에 해당하는 아이템 중 지정한 버전의 아이템을 가져옵니다.

Parameters:

itemID - 가져올 아이템의 ID

version - 가져올 아이템의 버전

isCompressed - 압축 여부

GetItems

```
public Stream[] GetItems(string[] itemIDs, bool[] isCompressed, bool[]  
isObjStreams)
```

지정한 아이템ID에 해당하는 아이템을 가져옵니다.

Parameters:

itemIDs - 가져올 아이템 ID

isCompressed - 압축 여부

isObjStreams - ODI 객체 스트림 필요 여부

GetItems

```
public Stream[] GetItems(string[] itemIDs, long[] modifiedTimes, bool[]  
isCompressed, bool[] isObjStreams)
```

지정한 아이템ID에 해당하는 아이템을 가져옵니다.

Parameters:

itemIDs - 가져올 아이템 ID

modifiedTimes - 압축 여부

isCompressed - 아이템을 최종 변경한 시간

isObjStreams - ODI 객체 스트림 필요 여부

GetItemInfo

```
public IItemInfo GetItemInfo(string itemID)
```

지정한 아이템 ID에 해당하는 아이템의 정보를 가져옵니다.

Parameters:

itemID - 정보를 가져올 아이템 ID

GetItemInfos

```
public IItemInfo[] GetItemInfos(string categoryID)
```

지정한 카테고리에 속해 있는 모든 아이템 정보를 가져옵니다.

Parameters:

categoryID - 아이템 정보를 얻을 카테고리 ID

GetItemInfoListInCategory

```
public IItemInfo[] GetItemInfoListInCategory(string categoryID, bool isRecursive)
```

아이템 정보를 가져옵니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

categoryID - 카테고리 ID(Full Path로 설정)

isRecursive - 하위 카테고리 포함 여부

SearchItem

```
public ISearchResult[] SearchItem(string categoryID, string checkOutUser, string startDate, string endDate, SearchOptions options, string fileFilter, string criteria)
```

지정한 카테고리의 아이템 중 지정한 옵션 및 조회 조건에 해당하는 아이템을 가져옵니다.

Parameters:

categoryID - 카테고리를 이동할 아이템 ID

checkOutUser - 체크아웃 계정

startDate - 시작일

endDate - 종료일

options - 조회 옵션

fileFilter - 파일확장자

criteria - 조회 조건

HasItem

```
public bool HasItem(string itemID)
```

지정한 아이템의 존재 여부를 확인합니다.

Parameters:

itemID - 아이템 ID

GetItemCount

```
public int GetItemCount(string categoryID)
```

지정한 카테고리에 속해 있는 모든 아이템의 개수를 가져옵니다.

Parameters:

categoryID - 아이템의 개수를 얻을 카테고리의 ID

Setting up category and item permissions

ModifyGroupAuthorityToCategory

```
public bool ModifyGroupAuthorityToCategory(string groupID, string categoryID, Authority permission)
```

지정한 카테고리 ID에 대한 지정한 그룹 ID의 권한 정보를 변경하고 변경 성공 여부를 반환합니다.

Parameters:

groupID - 그룹 ID

categoryID - 카테고리 ID

permission - 권한 정보

ModifyUserAuthorityToCategory

```
public bool ModifyUserAuthorityToCategory(string userID, string categoryID, Authority permission)
```

지정한 카테고리 ID에 대한 지정한 사용자 ID의 권한 정보를 변경하고 변경 성공 여부를 반환합니다.

Parameters:

userID - 사용자 ID

categoryID - 카테고리 ID

permission - 권한 정보

ModifyGroupAuthorityToItem

```
public bool ModifyGroupAuthorityToItem(string groupID, string itemID, Authority permission)
```

지정한 아이템에 대한 지정한 그룹 ID의 권한 정보를 변경하고 변경 성공 여부를 반환합니다.

Parameters:

groupID - 그룹 ID

itemID - 아이템 ID

permission - 권한 정보

ModifyUserAuthorityToItem

```
public bool ModifyUserAuthorityToItem(string userID, string itemID, Authority permission)
```

지정한 아이템에 대한 사용자의 권한 정보를 변경하고 변경 성공 여부를 가져옵니다.

Parameters:

userID - 사용자 ID

itemID - 아이템 ID

permission - 권한 정보

Getting category and item permissions

GetGroupAuthorityToCategory

```
public Authority GetGroupAuthorityToCategory(string groupID, string categoryID)
```

지정한 카테고리 ID에 대한 지정한 그룹 ID의 권한 정보를 가져옵니다.

Parameters:

groupID - 그룹 ID

categoryID - 카테고리 ID

GetGroupInfosOfCategory

```
public IGroupInfo[] GetGroupInfosOfCategory(string categoryID, Authority permission)
```

지정한 카테고리 ID에 대해 permission 이상의 권한이 부여된 모든 그룹 정보를 가져옵니다.

Parameters:

categoryID - 카테고리 ID

permission - 권한 정보

GetGroupAuthorityToItem

```
public Authority GetGroupAuthorityToItem(string groupID, string itemID)
```

지정한 아이템에 대한 지정한 그룹 ID의 권한 정보를 가져옵니다.

Parameters:

groupID - 그룹 ID

itemID - 아이템 ID

GetGroupInfosOfItem

```
public IGroupInfo[] GetGroupInfosOfItem(string itemID, Authority permission)
```

지정한 아이템 ID에 대해 permission 이상의 권한이 부여된 모든 그룹 정보를 가져옵니다.

Parameters:

itemID - 아이템 ID

permission - 권한 정보

GetUserAuthorityToCategory

```
public Authority GetUserAuthorityToCategory(string userID, string categoryID)
```

지정한 카테고리에 대한 지정한 사용자 ID의 권한 정보를 가져옵니다.

Parameters:

userID - 사용자 ID

categoryID - 카테고리 ID

GetUserInfosOfCategory

```
public IUserInfo[] GetUserInfosOfCategory(string categoryID, Authority permission)
```

지정한 카테고리 ID에 대해 permission 이상의 권한이 부여된 모든 사용자 정보를 가져옵니다.

Parameters:

categoryID - 카테고리 ID

permission - 권한 정보

GetUserAuthorityToItem

```
public Authority GetUserAuthorityToItem(string userID, string itemID)
```

지정한 아이템에 대한 지정한 사용자 ID의 권한 정보를 가져옵니다.

Parameters:

userID - 사용자 ID

itemID - 아이템 ID

GetUserInfosOfItem

```
public IUserInfo[] GetUserInfosOfItem(string itemID, Authority permission)
```

지정한 아이템 ID에 대해 permission 이상의 권한이 부여된 모든 사용자 정보를 가져옵니다.

Parameters:

itemID - 아이템 ID

permission - 권한 정보

GetCategoryInfosOfGroup

```
public ICategoryInfo[] GetCategoryInfosOfGroup(string groupID, string categoryID,  
Authority permission)
```

지정한 카테고리의 하위 카테고리 중에 지정한 그룹 ID의 권한이 permission 이상인 모든 카테고리의 목록을 가져옵니다.

Parameters:

groupID - 그룹 ID

categoryID - 카테고리 ID

permission - 권한 정보

GetCategoryInfosOfUser

```
public ICategoryInfo[] GetCategoryInfosOfUser(string userID, string categoryID,  
Authority permission)
```

지정한 카테고리의 하위 카테고리 중에 지정한 사용자의 권한이 permission 이상인 모든 카테고리의 목록을 가져옵니다.

Parameters:

userID - 사용자 ID

categoryID - 카테고리 ID

permission - 권한 정보

GetItemInfosOfGroup

```
public IItemInfo[] GetItemInfosOfGroup(string groupID, string categoryID, Authority permission)
```

지정한 카테고리의 아이템 중에 지정한 그룹 ID의 권한이 permission 이상인 모든 아이템 목록을 가져옵니다.

Parameters:

groupID - 그룹 ID

categoryID - 카테고리 ID

permission - 권한 정보

GetItemInfosOfUser

```
public IItemInfo[] GetItemInfosOfUser(string userID, string categoryID, Authority permission)
```

지정한 카테고리의 아이템 중에 지정한 사용자의 권한이 permission 이상인 모든 아이템 목록을 가져옵니다.

Parameters:

userID - 사용자 ID

categoryID - 카테고리 ID

permission - 권한 정보

Using item version control

CheckOut

```
public Stream[] CheckOut(string[] itemIDs, string[] checkOutFolders, long[] localFileTimes, bool[] isCompressed, string[] checkOutCmts)
```

지정한 아이템을 체크 아웃할 폴더에 체크 아웃합니다.

Parameters:

itemIDs - 체크 아웃할 아이템의 ID

checkOutFolders - 체크 아웃할 폴더 이름

localFileTimes - 체크 아웃할 아이템의 로컬 파일 시간

isCompressed - 압축 여부

checkOutCmts - 주석

CheckIn

```
public bool[] CheckIn(string[] itemIDs, bool[] isCompressed, Stream[] items,  
string comment, bool[] keepCheckOut)
```

시간을 지정하여 아이템을 체크인합니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

itemIDs - 체크인할 아이템의 ID

isCompressed - 압축 여부

items - 아이템의 입력 스트림

comment - 주석문

keepCheckOut - 체크 아웃 상태를 유지할지 여부

CheckIn

```
public bool[] CheckIn(string[] itemIDs, bool[] isCompressed, Stream[] items,  
string comment, bool[] keepCheckOut, long[] datetimes)
```

시간을 지정하여 아이템을 체크인합니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

itemIDs - 체크인할 아이템의 ID

isCompressed - 압축 여부

items - 아이템의 입력 스트림

comment - 주석문

datetimes - 설정할 시간. 0 또는 음수로 설정할 경우 현재 시간으로 적용됨

UndoCheckOut

```
public Stream[] UndoCheckOut(string[] itemIDs, bool[] replaceItems, bool[] isCompressed)
```

지정한 아이템을 체크 아웃 취소합니다.

Parameters:

itemIDs - 체크 아웃 취소할 아이템의 ID

replaceItems - 로컬 작업 폴더의 아이템을 가져올지 여부

isCompressed - 압축 여부

Upload

```
public bool Upload(string categoryID, Stream in)
```

압축된 아이템을 지정한 카테고리에 압축 해제한 후 압축 해제 성공 여부를 반환합니다.

Parameters:

categoryID - 카테고리 ID

in - 압축한 스트림

IsCheckoutUser

```
public bool[] isCheckoutUser(string[] ItemIDs) throws OZCPEException
```

현재 사용자가 지정한 아이템을 체크 아웃했는지 여부를 확인합니다.

Parameters:

itemIDs - 아이템 ID

GetHistoryInfos

```
public IHistoryInfo[] GetHistoryInfos(string itemID)
```

지정한 아이템의 히스토리 정보를 가져옵니다.

Parameters:

itemID - 히스토리 정보를 가져올 아이템의 ID

GetHistoryItemByDatetime

```
public Stream GetHistoryItemByDatetime(string itemID, long datetime, bool isCompressed, bool isObjStream)
```

지정한 시간 기준으로 최근 아이템을 가져옵니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

itemID - 가져올 아이템 ID

datetime - 가져올 시간

isCompressed - 압축 여부

isObjStream - 가져올 아이템이 ODI인 경우 Object Stream으로 가져올지 여부

GetHistoryItemListByDatetime

```
public IHistoryInfo[] GetHistoryItemListByDatetime(string itemID)
```

지정한 아이템의 히스토리 정보를 날짜순으로 가져옵니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

itemID - 히스토리 정보를 가져올 아이템의 ID

GetDeletedItemInfos

```
public IItemInfo[] GetDeletedItemInfos(string categoryID)
```

지정한 카테고리의 삭제된 아이템 정보를 가져옵니다.

"toBeDestroyed=false"로 설정하여 삭제한 아이템의 정보만 가져옵니다.

Parameters:

categoryID - 삭제된 아이템 정보를 가져올 카테고리 ID

GetDeletedItemHistoryInfos

```
public IHistoryInfo[] GetDeletedItemHistoryInfos(string itemID)
```

해당 삭제된 아이템의 정보를 가져옵니다.

"toBeDestroyed=false"로 설정하여 삭제한 아이템의 정보만 가져옵니다.

Parameters:

itemID - 정보를 가져올 아이템의 ID

ModifyHistoryItemComment

```
public bool ModifyHistoryItemComment(string itemID, int itemVersion, string  
newComment) throws OZCPEException
```

아이템 히스토리 중 특정 버전의 주석을 수정합니다.

Parameters:

itemID - 아이템 ID(Full Path로 설정)

itemVersion - 주석을 수정할 버전

newComment - 수정할 주석

ModifyHistoryItemDatetime

```
public bool ModifyHistoryItemDatetime(string itemID, int itemVersion, long datetime)
```

지정한 버전의 아이템 시간을 변경합니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

itemID - 시간을 변경할 아이템의 ID

itemVersion - 시간을 변경할 버전

datetime - 변경할 시간. 0 또는 음수로 설정할 경우 현재 시간으로 적용됨

RemoveHistory

```
public bool RemoveHistory(string itemID, int version)
```

지정한 아이템에 대해 특정 버전의 히스토리를 삭제합니다.

Parameters:

itemID - 히스토리를 삭제할 아이템의 ID

version - 히스토리를 삭제할 버전

RollBack

```
public bool RollBack(string itemID, int version, string comment)
```

지정한 아이템을 지정한 버전으로 복원시킵니다.

Parameters:

itemID - 복원시킬 아이템의 ID

version - 복원시킬 버전

comment - 주석문

Distribute items

Distribute

```
public Stream Distribute(string categoryID, bool isRecursive)
```

지정한 카테고리의 아이템을 배포 파일 형식으로 저장합니다.

Parameters:

categoryID - 카테고리 ID

isRecursive - 하위 카테고리 포함 여부

DistributeRepository

```
public Stream DistributeRepository(string[] itemIDs)
```

아이템을 배포 파일 형식으로 저장합니다.

Parameters:

itemIDs - 아이템 ID(Full Path로 설정)

DistributeRepositoryByDatetime

```
public Stream DistributeRepositoryByDatetime(string[] itemIDs)
```

지정한 아이템마다 가장 마지막에 체크인된 아이템을 가져와서 배포 파일 형식으로 저장합니다.

리파지토리 타입이 NONE인 경우에는 지원하지 않습니다.

Parameters:

itemIDs - 아이템 ID(Full Path로 설정)

Getting error

GetLastErrorCode

```
public OZErrorCode GetLastErrorCode(int index)
```

마지막에 호출한 메소드의 에러 코드를 가져옵니다.

Parameters:

index - 메소드 호출 시 요청 인덱스

GetLastErrorMessage

```
public OZErrorCode GetLastErrorMessage(int index)
```

마지막에 호출한 메소드의 에러 메시지를 가져옵니다.

Parameters:

RepositoryException

OZRepositoryException 기능과 관련된 속성을 제공하는 클래스입니다.

Constructor Detail

OZRepositoryException

```
public OZRepositoryException(string _msg)
```

OZRepositoryException 관련 기능을 수행합니다.

Parameters:

_msg - 에러 메시지

OZRepositoryException

```
public OZRepositoryException(int code, string _msg)
```

OZRepositoryException 관련 기능을 수행합니다.

Parameters:

code - 에러 코드

_msg - 에러 메시지

Property Detail

ErrorCode

```
public OZErrorCode ErrorCode
```

에러 코드

Message

```
public string Message
```

에러 메시지

Interface

OZ Repository API에서 제공하는 인터페이스는 아래와 같습니다.

- [ICategoryInfo](#)
- [IGroupInfo](#)
- [IHistoryInfo](#)
- [IItemInfo](#)
- [ILoginInfo](#)
- [IRepository](#)
- [IRepositoryCategory](#)
- [IRepositoryExternalLogin](#)
- [IRepositoryGroup](#)
- [IRepositoryGroupAuthorityToCategory](#)
- [IRepositoryGroupAuthorityToItem](#)
- [IRepositoryHistory](#)
- [IRepositoryItem](#)
- [IRepositoryMultiLoginUser](#)
- [IRepositoryUser](#)
- [IRepositoryUserAuthorityToCategory](#)
- [IRepositoryUserAuthorityToItem](#)
- [IUserInfo](#)

ICategoryInfo

카테고리 정보를 제공하는 인터페이스입니다.

Property Detail

ID

`string ID`

현재 카테고리의 ID를 가져옵니다.

Name

string Name

현재 카테고리의 이름을 가져옵니다.

ParentID

string ParentID

현재 카테고리의 상위 카테고리 ID를 가져옵니다.

ParentName

string ParentName

현재 카테고리의 상위 카테고리 이름을 가져옵니다.

Comment

string Comment

현재 카테고리의 주석문을 가져옵니다.

Description

string Description

현재 카테고리의 설명 내용을 가져옵니다.

UpdateTime

long UpdateTime

현재 카테고리를 갱신한 시간 정보를 가져옵니다.

CreatedUserID

string CreatedUserID

현재 카테고리를 생성한 사용자 ID를 가져옵니다.

CreatedUserName

string CreatedUserName

현재 카테고리를 생성한 사용자 이름을 가져옵니다.

Permission

Authority Permission

현재 카테고리의 권한 정보를 가져옵니다.

IGroupInfo

그룹 정보를 제공하는 인터페이스입니다.

Property Detail

Administrators

IDictionary Administrators

해당 그룹의 관리자 정보를 가져옵니다.

관리자 정보는 "key=사용자ID,value=사용자이름"으로 구성된 HashMap으로 반환됩니다.

ID

string ID

해당 그룹의 그룹 ID를 가져옵니다.

Name

string Name

해당 그룹의 그룹 이름을 가져옵니다.

ParentID

string ParentID

해당 그룹의 상위 그룹 ID를 가져옵니다.

ParentName

string ParentName

해당 그룹의 상위 그룹 이름을 가져옵니다.

Description

string Description

해당 그룹의 설명 내용을 가져옵니다.

UpdateTime

long UpdateTime

해당 그룹을 갱신한 시간 정보를 가져옵니다.

Permission

Authority Permission

해당 그룹의 권한 정보를 가져옵니다.

IHistoryInfo

히스토리 정보를 제공하는 인터페이스입니다.

Property Detail

ItemID

string ItemID

아이템 ID를 가져옵니다.

ItemName

string ItemName

아이템 이름을 가져옵니다.

Version

int Version

아이템의 버전 개수를 가져옵니다.

CheckInTime

long CheckInTime

아이템의 버전 별 체크인 시간을 가져옵니다.

CheckInUser

string CheckInUser

아이템의 버전 별 체크인 사용자 ID를 가져옵니다.

CheckInComment

string CheckInComment

아이템의 버전 별 체크인 주석문을 가져옵니다.

ItemInfo

아이템 정보를 제공하는 인터페이스입니다.

Property Detail

IsCheckedOut

`bool IsCheckedOut`

해당 아이템의 체크 아웃 여부를 가져옵니다

IsDeletable

`bool IsDeletable`

해당 아이템의 삭제 여부를 가져옵니다.

"toBeDestroyed =false"로 설정하여 삭제한 아이템의 정보만 가져옵니다.

ID

`string ID`

해당 아이템의 아이템 ID를 가져옵니다.

Name

`string Name`

해당 아이템의 아이템 이름을 가져옵니다.

CategoryID

`string CategoryID`

해당 아이템이 속한 카테고리의 카테고리 ID를 가져옵니다.

CategoryName

`string CategoryName`

해당 아이템이 속한 카테고리의 카테고리 이름을 가져옵니다.

UpdateTime

long UpdateTime

해당 아이템을 갱신한 시간 정보를 가져옵니다.

Description

string Description

해당 아이템의 설명 내용을 가져옵니다.

Comment

string Comment

해당 아이템의 주석문을 가져옵니다.

CreatedUserID

string CreatedUserID

해당 아이템을 생성한 사용자 ID를 가져옵니다.

CreatedUserName

string CreatedUserName

해당 아이템을 생성한 사용자 이름을 가져옵니다.

CheckOutUser

string CheckOutUser

해당 아이템을 체크 아웃한 사용자 이름을 가져옵니다.

CheckOutLocalPath

string CheckOutLocalPath

체크 아웃한 아이템의 폴더 이름을 가져옵니다.

Permission

Authority Permission

해당 아이템의 권한 정보를 가져옵니다.

ILoginInfo

로그인 정보를 제공하는 인터페이스입니다.

Property Detail

ClientType

ClientType ClientType

클라이언트에서 호출한 클라이언트의 종류를 가져옵니다.

MACAddress

string MACAddress

클라이언트로부터 MAC Address 정보를 가져옵니다.

IPAddress

string IPAddress

클라이언트로부터 IP Address 정보를 가져옵니다.

HDDSerialNo

string HDDSerialNO

클라이언트로부터 HDD Serial 번호를 가져옵니다.

ProtocolVersion

int ProtocolVersion

클라이언트로부터 프로토콜 버전을 가져옵니다.

UserID

string UserID

클라이언트가 로그인한 사용자 ID를 가져옵니다.

UserName

string UserName

클라이언트가 로그인한 사용자 이름을 가져옵니다.

SessionID

string SessionID

클라이언트가 로그인 시 생성된 세션 ID를 가져옵니다.

ClientIP

`string ClientIP`

클라이언트가 요청한 클라이언트 IP를 가져옵니다.

IRepository

리포지토리를 구성하는 객체를 가져오거나 특정 기능을 실행하는 인터페이스입니다.

Property Detail

Property

`oz.util.IStringDictionary Property`

리포지토리 설정을 변경합니다.

Version

`int Version`

리포지토리 버전을 정의한 후 정의한 버전을 가져옵니다.

Category

`IRepositoryCategory Category`

카테고리 인터페이스를 구현한 경우 구현된 객체를 가져옵니다.

Item

`IRepositoryItem Item`

아이템 인터페이스를 구현한 경우 해당 구현된 객체를 가져옵니다.

ItemHistory

`IRepositoryHistory ItemHistory`

아이템 히스토리 인터페이스를 구현한 경우 해당 구현된 객체를 가져옵니다.

Group

`IRepositoryGroup Group`

그룹 인터페이스를 구현한 경우 해당 구현된 객체를 가져옵니다.

User

`IRepositoryUser User`

유저 인터페이스를 구현한 경우 해당 구현된 객체를 가져옵니다.

MultiLoginUser

`IRepositoryMultiLoginUser MultiLoginUser`

멀티 로그인 유저 인터페이스를 구현한 경우 해당 구현된 객체를 가져옵니다.

GroupAuthorityToCategory

`IRepositoryGroupAuthorityToCategory GroupAuthorityToCategory`

카테고리에 대한 그룹의 권한 인터페이스를 구현한 경우 해당 구현된 클래스를 가져옵니다.

GruopAuthorityToItem

`IRepositoryGroupAuthorityToItem GruopAuthorityToItem`

아이템에 대한 그룹의 권한 인터페이스를 구현한 경우 해당 구현된 객체를 가져옵니다.

UserAuthorityToCtegory

`IRepositoryUserAuthorityToCategory UserAuthorityToCtegory`

카테고리에 대한 사용자의 권한 인터페이스를 구현한 경우 해당 구현된 클래스를 가져옵니다.

UserAuthorityToItem

`IRepositoryUserAuthorityToItem UserAuthorityToItem`

아이템에 대한 사용자의 권한 인터페이스를 구현한 경우 해당 구현된 클래스를 가져옵니다.

Method Detail

Open

`void Open()`

서버 구동 시 리파지토리를 엽니다.

Close

`void Close()`

서버 종료 시 리파지토리를 종료합니다.

Login

```
object Login(string username, string password, ILoginInfo loginInfo, HttpContext ctx)
```

클라이언트를 호출할 때 해당 메소드를 호출하여 로그인한 세션 정보를 가져옵니다.

Parameters:

UserName - 접속한 사용자 이름

password - 접속한 사용자의 패스워드

loginInfo - 로그인 정보

ctx - HTTP 접속 시 현재 컨텍스트. 윈도우즈 서비스 형태의 서버인 경우 null 값을 설정합니다.

Logout

```
bool Logout(object session, string userName, ILoginInfo loginInfo, HttpContext ctx)
```

지정한 사용자 ID에 해당하는 사용자를 로그아웃 즉, 접속 해제시키며 로그아웃 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

userName - 사용자 이름

loginInfo - 로그인 정보

ctx - HTTP 접속 시 현재 컨텍스트. 윈도우즈 서비스 형태의 서버인 경우 null 값을 설정합니다.

UpdateSessionState

```
void UpdateSessionState(object session, SessionState state)
```

클라이언트를 호출할 때 해당 메소드를 호출하여 로그인한 세션 정보를 가져옵니다.

Parameters:

session - 세션 ID

state - 서버 상태(None, BareServer, FromServer)

IRepositoryCategory

카테고리 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

CategoryAccess AccessType

카테고리 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- CreateCategories = 0x00000002
- ModifyCategoryName = 0x00000004
- DeleteCategories = 0x00000008
- UndeleteCategories = 0x00000010
- HasItem = 0x00000020
- GetItemCount = 0x00000040
- GetItemInfos = 0x00000080
- GetCategoryID = 0x00000100
- GetCategoryInfo = 0x00000200
- GetDeletedItemInfos = 0x00000400
- SearchItem = 0x00000800,
- GetCategoryInfos = 0x00001000
- TransferItems = 0x00002000
- TransferCategory = 0x00004000

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

CreateItems

```
string[] CreateItems(object session, string[] itemNames, string[] descriptions,  
string[] categoryIDs, Stream[] items, string comment, OZErrorCode[] errCodes,  
string[] errMsgs)
```

새로운 아이템을 생성하고 생성된 아이템의 ID를 반환합니다.

Parameters:

session - 세션 ID

itemNames - 새로 생성할 아이템의 ID

descriptions - 새로 생성할 아이템의 설명 내용

categoryIDs - 새로 생성할 아이템이 카테고리 ID

items - 새로 생성할 아이템의 입력 스트림

comment - 주석문

errCodes - 에러 코드

errMsgs - 에러 메시지

CreateCategories

```
string[] CreateCategories(object session, string[] categoryNames, string[]  
parentCategoryIDs, string comment, OZErrorCode[] errCodes, string[] errMsgs)
```

카테고리를 새로 생성하고 생성된 카테고리의 ID를 반환합니다.

Parameters:

session - 세션 ID

categoryNames - 생성할 카테고리의 이름

parentCategoryIDs - 생성할 카테고리의 상위 카테고리 ID

comment - 주석문

errCodes - 에러 코드

errMsgs - 에러 메시지

ModifyCategoryName

```
string ModifyCategoryName(object session, string categoryID, string  
newCategoryName, string comment)
```

지정한 카테고리 ID에 해당하는 카테고리의 이름을 변경합니다.

Parameters:

session - 세션 ID

categoryID - 이름을 변경할 카테고리의 ID

newCategoryName - 변경할 카테고리 이름

comment - 주석문

DeleteCategories

```
bool[] DeleteCategories(object session, string[] categoryIDs, bool[]  
toBeDestroyed, string comment, OZErrorCode[] errCodes, string[] errMsgs)
```

지정한 카테고리 ID에 해당하는 카테고리를 삭제합니다.

Parameters:

session - 세션 ID

categoryIDs - 삭제할 카테고리의 ID

toBeDestroyed - 카테고리를 영구 삭제할 지 여부

comment - 주석문

errCodes - 에러 코드

errMsgs - 에러 메시지

UndeleteCategories

```
bool[] UndeleteCategories(object session, string[] categoryIDs, string comment,  
OZErrorCode[] errCodes, string[] errMsgs)
```

삭제된 카테고리를 복원하고 복원 성공 여부를 가져옵니다.

서버 버전이 2007년 09월 01일 이전 버전일 경우 해당 메소드가 지원되지 않습니다.

카테고리 삭제 시 "isDestroys=false"로 설정하였을 경우 삭제된 카테고리의 복원이 가능합니다.

Parameters:

session - 세션 ID

CategoryIDs - 복원할 카테고리의 ID

comment - 주석문

errorCode - 에러 코드

errorMsg - 에러 메시지

GetItemCount

```
int GetItemCount(object session, string categoryID)
```

지정한 카테고리에 속해 있는 모든 아이템의 개수를 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 아이템의 개수를 얻을 카테고리의 ID

GetItemInfos

IItemInfo[] GetItemInfos(object session, string categoryID)

지정한 카테고리에 속해 있는 모든 아이템 정보를 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 아이템의 정보를 얻을 카테고리의 ID

GetCategoryInfos

ICategoryInfo[] GetCategoryInfos(object session, string categoryID)

지정한 카테고리의 정보를 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 정보를 가져올 카테고리 ID

GetCategoryID

string GetCategoryID(object session, string itemID)

지정한 아이템이 존재하는 카테고리의 ID를 반환합니다.

Parameters:

session - 세션 ID

itemID - 카테고리의 ID를 얻을 아이템 ID

GetCategoryInfo

ICategoryInfo GetCategoryInfo(object session, string categoryID)

지정한 카테고리의 정보를 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 정보를 가져올 카테고리 ID

GetDeletedItemInfos

IItemInfo[] GetDeletedItemInfos(object session, string categoryID)

지정한 카테고리의 삭제된 아이템 정보를 가져옵니다.

"toBeDestroyed=false"로 설정하여 삭제한 아이템의 정보만 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 삭제된 아이템 정보를 가져올 카테고리 ID

TransferItems

```
bool TransferItems(object session, string[] itemIDs, string targetCategoryID)
```

지정한 아이템의 카테고리를 이동하고 카테고리 이동 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

itemIDs - 카테고리를 이동할 아이템 ID

targetCategoryID - 이동할 카테고리 ID

TransferCategory

```
bool TransferCategory(object session, string categoryID, string targetCategoryID)
```

지정한 카테고리를 다른 카테고리로 이동하고 카테고리 이동 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 카테고리를 이동할 카테고리 ID

targetCategoryID - 이동할 카테고리 ID

IRepositoryExternalLogin

외부 API와 연계하여 로그인하는 기능을 설정하는 인터페이스입니다.

Method Detail

CreateUser

```
string CreateUser(oz.framework.log.OZLog log, string id, string pwd)
```

외부 API와 연계하여 로그인할 사용자 ID를 생성합니다.

Parameters:

log - 서버 로그

id - 생성할 사용자 ID

pwd - 생성할 사용자 비밀번호

Login

```
string Login(oz.framework.log.OZLog log, string id, string pwd)
```

외부 API와 연계된 로그인 ID로 로그인하고 에러 발생 시 에러 메시지를 리턴합니다.

Parameters:

log - 서버 로그

id - 로그인할 사용자 ID

pwd - 로그인할 사용자 비밀번호

IRepositoryGroup

그룹 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

GroupAccess AccessType

그룹 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- CreateGroup = 0x00000001
- ModifyGroupName = 0x00000002
- ModifyGroupDescription = 0x00000004
- DeleteGroup = 0x00000008
- CreateUser = 0x00000010
- TransferUser = 0x00000020
- GetUserInfos = 0x00000040

- GetGroupInfo = 0x00000080
- GetSubGroupInfos = 0x00000100
- GetParentGroupInfo = 0x00000200
- GetGroupID = 0x00000400
- AddGroupAdministrator = 0x00000800
- RemoveGroupAdministrator = 0x00001000
- IsGroupAdministrator = 0x00002000
- GetAdministrators = 0x00004000
- TransferGroup = 0x00008000

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

CreateGroup

string CreateGroup(object session, string groupName, string parentGroupID, string description)

새로운 그룹을 생성하고 생성된 그룹의 ID를 반환합니다.

Parameters:

session - 세션 ID

groupName - 새로 생성할 그룹의 이름

parentGroupID - 새로 생성할 그룹의 상위 그룹 ID

description - 그룹에 대한 설명 내용

ModifyGroupName

string ModifyGroupName(object session, string groupID, string groupName)

지정한 그룹 ID에 해당하는 그룹의 이름을 변경하고 변경된 그룹 ID를 반환합니다.

Parameters:

session - 세션 ID

groupID - 그룹의 이름을 변경할 그룹 ID

groupName - 변경할 그룹의 이름

ModifyGroupDescription

`bool ModifyGroupDescription(object session, string groupID, string description)`

지정한 그룹 ID에 해당하는 그룹의 설명을 변경하고 변경 성공 여부를 반환합니다.

Parameters:

session - 세션 ID

groupID - 그룹의 설명을 변경할 그룹 ID

description - 변경할 그룹의 설명

DeleteGroup

`bool DeleteGroup(object session, string groupID)`

지정한 그룹 ID에 해당하는 그룹을 삭제하고 삭제 성공 여부를 반환합니다.

Parameters:

session - 세션 ID

groupID - 삭제할 그룹 ID

CreateUser

`string CreateUser(object session, string userName, string password, string groupID, string description)`

지정한 그룹에 새로운 사용자 ID를 생성하고 생성된 사용자 ID를 반환합니다.

Parameters:

session - 세션 ID

userName - 사용자 이름

password - 패스워드

groupID - 그룹ID

description - 사용자에게 대한 설명 내용

GetUserInfos

`IUserInfo[] GetUserInfos(object session, string groupID)`

지정한 그룹 ID에 등록되어 있는 모든 사용자의 정보를 가져옵니다.

Parameters:

session - 세션 ID

groupID - 사용자의 정보를 가져올 그룹 ID

GetGroupInfo

IGroupInfo GetGroupInfo(object session, string groupID)

지정한 그룹 ID에 해당하는 그룹의 정보를 가져옵니다.

Parameters:

session - 세션 ID

groupID - 그룹 ID

GetSubGroupInfos

IGroupInfo[] GetSubGroupInfos(object session, string groupID)

지정한 그룹의 하위 그룹 정보를 가져옵니다.

Parameters:

session - 세션 ID

groupID - 하위 그룹의 정보를 가져올 그룹 ID

GetParentGroupInfo

IGroupInfo GetParentGroupInfo(object session, string groupID)

지정한 그룹의 상위 그룹의 정보를 가져옵니다.

Parameters:

session - 세션 ID

groupID - 상위 그룹의 정보를 가져올 그룹 ID

GetGroupID

string GetGroupID(object session, string userID)

지정한 사용자가 속해 있는 그룹의 정보를 가져옵니다.

Parameters:

session - 세션 ID

userID - 그룹 정보를 가져올 사용자 ID

AddGroupAdministrator

bool AddGroupAdministrator(object session, string userID, string groupID)

지정한 그룹의 그룹 관리자를 추가하고 추가 성공 여부를 반환합니다.

Parameters:

session - 세션 ID

userID - 그룹 관리자로 추가할 사용자 ID

groupID - 그룹 관리자를 추가할 그룹 ID

RemoveGroupAdministrator

bool RemoveGroupAdministrator(object session, string userID, string groupID)

지정한 그룹의 그룹 관리자를 해제하고 해제 성공 여부를 반환합니다.

Parameters:

session - 세션 ID

userID - 그룹 관리자 권한을 해제할 사용자 ID

groupID - 그룹 관리자를 해제할 그룹 ID

IsGroupAdministrator

bool IsGroupAdministrator(object session, string userID, string groupID)

지정한 사용자 ID에 해당하는 사용자가 해당 그룹의 관리자인지 여부를 확인합니다.

Parameters:

session - 세션 ID

userID - 관리자인지 체크할 사용자 ID

groupID - 그룹 ID

GetAdministrators

IUserInfo[] GetAdministrators(object session, string groupID)

지정한 그룹의 그룹 관리자 정보를 가져옵니다.

Parameters:

session - 세션 ID

groupID - 그룹 관리자 정보를 가져올 그룹 ID

TransferUser

bool TransferUser(object session, string userID, string newGroupID)

지정한 사용자의 그룹을 이동하고 그룹 이동 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

userID - 그룹을 이동할 사용자 ID

newGroupID - 이동할 그룹 ID

TransferGroup

`bool TransferGroup(object session, string groupID, string targetGroupID)`

지정한 그룹을 다른 그룹으로 이동하고 그룹 이동 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

userID - 그룹을 이동할 그룹 ID

targetGroupID - 이동할 그룹 ID

IRepositoryGroupAuthorityToCategory

카테고리에 대한 그룹 권한 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

`AuthorityAccess AccessType`

카테고리에 대한 그룹 권한 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- ModifyGroupAuthToCategory = 0x00000001
- GetGroupAuthToCategory = 0x00000002
- GetGroupInfosOfCategory = 0x00000004
- GetItemInfosInCategoryOfGroup = 0x00000008
- GetCategoryInfosOfGroup = 0x00010000

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

Modify

`bool Modify(object session, string categoryID, string groupID, Authority permission)`

지정한 카테고리나 그룹의 권한 정보를 변경하고 변경 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 권한 정보를 변경할 카테고리 ID

groupID - 권한 정보를 변경할 그룹 ID

permission - 권한 정보. 1 : VIEW. 2 : READ. 4 : WRITE. 권한 정보는 OR 조건으로 처리해야 합니다.

GetAuthority

`Authority GetAuthority(object session, string groupID, string categoryID)`

지정한 카테고리에 대한 지정한 그룹의 그룹 권한을 가져옵니다.

Parameters:

session - 세션 ID

groupID - 그룹 ID

categoryID - 권한 정보를 가져올 카테고리 ID

GetGroupInfos

`IGroupInfo[] GetGroupInfos(object session, string categoryID, Authority permission)`

지정한 카테고리 ID에 대해 permission 이상의 권한을 가지고 있는 모든 사용자 정보를 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 카테고리 ID

permission - 권한 정보

GetItemInfos

```
IItemInfo[] GetItemInfos(object session, string groupID, string categoryID,  
Authority permission)
```

지정한 카테고리의 아이템 중에 지정한 그룹의 권한이 permission 이상인 모든 아이템의 목록을 가져옵니다.

Parameters:

session - 세션 ID

groupID - 그룹 ID

categoryID - 카테고리 ID

permission - 권한 정보

GetCategoryInfos

```
ICategoryInfo[] GetCategoryInfos(object session, string groupID, string  
categoryID, Authority permission)
```

지정한 카테고리의 하위 카테고리 중에 지정한 그룹의 권한이 permission 이상인 모든 카테고리의 목록을 가져옵니다.

Parameters:

session - 세션 ID

groupID - 그룹 ID

categoryID - 카테고리 ID

permission - 권한 정보

IRepositoryGroupAuthorityToItem

아이템에 대한 그룹 권한 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

AuthorityAccess AccessType

아이템에 대한 그룹 권한 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- ModifyGroupAuthToItem = 0x00000010
- GetGroupAuthToItem = 0x00000020
- GetGroupInfosOfItem = 0x00000040
- GetItemInfosOfGroup = 0x00000080

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

Modify

bool Modify(object session, string groupID, string itemID, Authority permission)

지정한 아이템 ID에 대한 지정한 그룹 ID의 권한 정보를 변경하고 변경 성공 여부를 반환합니다.

Parameters:

session - 세션 ID

groupID - 그룹 ID

itemID - 아이템 ID

permission - 권한 정보

GetAuthority

Authority GetAuthority(object session, string groupID, string itemID)

지정한 아이템 ID에 대한 지정한 그룹 ID의 권한 정보를 가져옵니다.

Parameters:

session - 세션 ID

groupID - 그룹 ID

itemID - 아이템 ID

GetGroupInfos

IGroupInfo[] GetGroupInfos(object session, string itemID, Authority permission)

지정한 아이템 ID에 대해 permission 이상의 권한이 부여된 모든 그룹 정보를 가져옵니다.

Parameters:

session - 세션 ID

itemID - 아이템 ID

permission - 권한 정보

GetItemInfos

IItemInfo[] GetItemInfos(object session, string groupID, Authority permission)

지정한 그룹 ID에 해당하는 그룹에게 permission 이상의 권한이 부여된 모든 아이템의 정보를 가져옵니다.

Parameters:

session - 세션 ID

groupID - 그룹 ID

permission - 권한 정보

IRepositoryHistory

아이템 히스토리에 대한 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

HistoryAccess AccessType

아이템 히스토리 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- RollBack = 0x00000001
- GetItem = 0x00000002
- GetHistoryInfos = 0x00000004
- GetDeletedItemHistoryInfos = 0x00000008
- RemoveHistory = 0x00000010

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

RollBack

`bool RollBack(object session, string itemID, int version, string comment)`

지정한 아이템을 지정한 버전으로 복원시킵니다.

Parameters:

session - 세션 ID

itemID - 복원시킬 아이템의 ID

version - 복원시킬 버전

comment - 주석문

GetItem

`Stream GetItem(object session, string itemID, int version)`

지정한 아이템 ID에 해당하는 아이템 중 지정한 버전의 아이템을 가져옵니다.

Parameters:

session - 세션 ID

itemID - 가져올 아이템의 ID

version - 가져올 아이템의 버전

GetHistoryInfos

`IHistoryInfo[] GetHistoryInfos(object session, string itemID)`

지정한 아이템의 히스토리 정보를 가져옵니다.

Parameters:

session - 세션 ID

itemID - 히스토리 정보를 가져올 아이템의 ID

GetDeleteHistoryItemInfo

IHistoryInfo[] GetDeletedItemHistoryInfos(object session, string itemID)

삭제된 히스토리 리스트를 가져옵니다.

"toBeDestroyed =false"로 설정하여 삭제한 아이템의 정보만 가져옵니다.

Parameters:

session - 세션 ID

itemIDs - 히스토리 정보를 가져올 아이템의 ID

RemoveHistory

bool RemoveHistory(object session, string itemID, int version)

지정한 아이템에 대해 특정 버전의 히스토리를 삭제합니다.

Parameters:

session - 세션 ID

itemID - 히스토리를 삭제할 아이템의 ID

version - 히스토리를 삭제할 버전

IRepositoryItem

아이템에 대한 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

CategoryAccess AccessType

아이템 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- CreateItems = 0x00000001
- ModifyItemName = 0x00000002
- DeleteItems = 0x00000004
- UndeleteItems = 0x00000008
- ModifyItemDescription = 0x00000010

- GetItemInfo = 0x00000040
- HasItem = 0x00000020
- GetItemsUnconditionally = 0x00000080,
- GetItems = 0x00000100
- CheckOut = 0x00000200,
- CheckIn = 0x00000400
- UndoCheckOut = 0x00000800
- IsCheckOutUser = 0x00001000

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

CreateItems

string[] CreateItems(object session, string[] itemNames, string[] descriptions, Stream[] items, string comment, OZErrorCode[] errCodes, string[] errMsgs)

새로운 아이템을 생성하고 생성된 아이템의 ID를 반환합니다.

Parameters:

session - 세션 ID

itemNames - 새로 생성할 아이템의 ID

descriptions - 새로 생성할 아이템의 설명 내용

items - 새로 생성할 아이템의 입력 스트림

comment - 주석문

errCodes - 에러 코드

errMsgs - 에러 메시지

ModifyItemName

string ModifyItemName(object session, string itemID, string newItemName, string comment)

지정한 아이템 ID에 해당하는 아이템 이름을 변경하고 변경된 아이템 ID를 반환합니다.

Parameters:

session - 세션 ID

itemID - 이름을 변경할 아이템 ID

newItemName - 변경할 아이템 이름

comment - 주석문

DeleteItems

```
bool[] DeleteItems(object session, string[] itemIDs, bool[] toBeDestroyed, string comment, OZErrorCode[] errCodes, string[] errMsgs)
```

지정한 아이템을 리파지토리에서 삭제하고 아이템 삭제 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

itemIDs - 삭제할 아이템의 ID

toBeDestroyed - 아이템을 영구 삭제할 지 여부

comment - 주석문

errCodes - 에러 코드

errMsgs - 에러 메시지

UnDeleteItems

```
bool[] UndeleteItems(object session, string[] itemIDs, string comment, OZErrorCode[] errCodes, string[] errMsgs)
```

삭제된 아이템을 복원하고 복원 성공 여부를 가져옵니다.

아이템 삭제 시 "toBeDestroyed =false"로 설정하였을 경우 삭제된 아이템의 복원이 가능합니다.

Parameters:

session - 세션 ID

itemIDs - 복원할 아이템 ID

comment - 주석문

errCodes - 에러 코드

errMsgs - 에러 메시지

ModifyItemDescription

```
bool ModifyItemDescription(object session, string itemID, string description)
```

지정한 아이템 ID에 해당하는 아이템 설명을 변경하고 변경 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

itemID - 변경할 아이템 ID

description - 변경할 내용

GetItemInfo

```
IItemInfo GetItemInfo(object session, string itemID)
```

지정한 아이템 ID에 해당하는 아이템 정보를 가져옵니다.

Parameters:

session - 세션 ID

itemID - 아이템 정보를 가져올 아이템 ID

HasItem

```
bool HasItem(object session, string itemID)
```

지정한 아이템의 존재 여부를 가져옵니다.

Parameters:

session - 세션 ID

itemID - 아이템 ID

GetItems

```
Stream[] GetItems(object session, string[] itemIDs, OZErrorCode[] errCodes,  
string[] errMsgs)
```

지정한 아이템을 가져온 후 아이템 스트림을 가져옵니다.

Parameters:

session - 세션 ID

itemIDs - 가져올 아이템 ID

errCodes - 에러 코드

errMsgs - 에러 메시지

GetItems

```
Stream[] GetItems(object session, string[] itemIDs, long[] modifiedTimes,  
ItemOptions[] options, OZErrorCode[] errCodes, string[] errMsgs)
```

지정한 아이템을 가져온 후 아이템 스트림을 가져옵니다.

Parameters:

session - 세션 ID

itemIDs - 가져올 아이템 ID

modifiedTimes - 클라이언트 시간

options - 에러 메시지 처리에 대한 옵션

errCodes - 에러 코드

errMsgs - 에러 메시지

Checkout

```
Stream[] Checkout(object session, string[] itemIDs, string[] localCheckoutFolders,  
long[] localFileTimes, OZErrorCode[] errCodes, string[] errMsgs)
```

지정한 아이템을 지정한 사용자 ID로 체크 아웃할 폴더에 체크 아웃을 하고 로컬 파일의 시간이 아이템 시간 보다 작을 경우 아이템 스트림을 가져옵니다.

Parameters:

session - 세션 ID

itemIDs - 체크 아웃할 아이템의 ID

localCheckoutFolders - 체크 아웃할 로컬 폴더 이름

localFileTimes - 체크 아웃할 아이템의 로컬 파일 시간

errCodes - 에러 코드

errMsgs - 에러 메시지

CheckIn

```
bool[] CheckIn(object session, string[] itemIDs, Stream[] items, string comment,  
bool[] keepCheckOut, OZErrorCode[] errCodes, string[] errMsgs)
```

지정한 아이템을 지정한 사용자 ID로 체크인하고 체크인 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

itemIDs - 체크인할 아이템 ID

items - 체크인할 아이템 스트림

comment - 주석문

keepCheckOut - 체크 아웃 상태를 유지할지 여부

errCodes - 에러 코드

errMsgs - 에러 메시지

UndoCheckOut

```
Stream[] UndoCheckOut(object session, string[] itemIDs, bool[] replace,  
OZErrorCode[] errCodes, string[] errMsgs)
```

지정한 아이템을 체크 아웃 취소합니다.

Parameters:

session - 세션 ID

itemIDs - 체크 아웃 취소할 아이템의 ID

replaces - 로컬 파일 변경 여부

errCodes - 에러 코드

errMsgs - 에러 메시지

IsCheckOutUser

```
bool[] IsCheckOutUser(object session, string[] itemIDs)
```

지정한 사용자가 지정한 아이템을 체크 아웃했는지 여부를 확인합니다.

Parameters:

session - 세션 ID

itemIDs - 아이템 ID

IRepositoryMultiLoginUser

멀티 로그인에 대한 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

MultiLoginAccess AccessType

멀티 로그인 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- int ACCESS_MULTILoginUSER_NOT = 0x00000000
- int ACCESS_DISABLE_USER_LOGIN = 0x00000001
- int ACCESS_ENABLE_USER_LOGIN = 0x00000002

- `int ACCESS_IS_LOGIN_ENABLED = 0x00000004`

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

DisableLogin

`void DisableLogin(object session, string userID)`

지정한 세션 ID의 사용자를 로그인하지 못하도록 설정합니다.

Parameters:

session - 세션 ID

userID - 로그인하지 못하도록 할 사용자 ID

EnableUserLogin

`void EnableLogin(object session, string userID)`

지정한 세션 ID의 사용자를 로그인 가능하도록 설정합니다.

Parameters:

session - 세션 ID

userID - 로그인 가능하도록 할 사용자 ID

IsLoginEnabled

`bool IsLoginEnabled(object session, string userID)`

지정한 세션 ID의 사용자가 로그인이 가능한지 여부를 가져옵니다.

Parameters:

session - 세션 ID

userID - 로그인 가능 여부를 체크할 사용자 ID

IRepositoryUser

사용자에 대한 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

UserAccess AccessType

사용자 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- CreateUser = 0x00000001
- ModifyName = 0x00000002
- ModifyPassword = 0x00000004
- ModifyDescription = 0x00000008
- DeleteUser = 0x00000010
- GetUserInfo = 0x00000020
- CheckPassword = 0x00000040
- GetUserInfos = 0x00000080
- IsAdministrator = 0x00000100

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

CreateUser

string CreateUser(object session, string userName, string password, string description)

새로운 사용자를 생성하고 생성된 사용자의 ID를 반환합니다.

Parameters:

session - 세션 ID

userName - 사용자 이름

password - 패스워드

description - 사용자에게 대한 설명 내용

ModifyName

string ModifyName(object session, string userID, string newUserName)

지정한 사용자 ID에 해당하는 사용자의 이름을 변경하고 사용자 이름이 변경된 사용자 ID를 반환합니다.

Parameters:

session - 세션 ID

userID - 사용자 ID

newUserName - 변경할 사용자 이름

ModifyPassword

bool ModifyPassword(object session, string userID, string oldPassword, string newPassword)

지정한 사용자의 패스워드를 변경하고 변경 성공 여부를 반환합니다.

Parameters:

session - 세션 ID

userID - 패스워드를 변경할 사용자 ID

oldPassword - 변경 전 패스워드

newPassword - 변경할 패스워드

ModifyUserDescription

bool ModifyDescription(object session, string userID, string description)

지정한 사용자 ID에 해당하는 사용자의 사용자 설명을 변경하고 변경 성공 여부를 반환합니다.

Parameters:

session - 세션 ID

userID - 사용자 설명 내용을 변경할 사용자 ID

description - 변경할 사용자 설명

DeleteUser

bool DeleteUser(object session, string userID)

지정한 사용자 ID에 해당하는 사용자의 모든 정보를 삭제하고 삭제 성공 여부를 반환합니다.

Parameters:

session - 세션 ID

userID - 삭제할 사용자의 ID

GetUserInfo

IUserInfo GetUserInfo(object session, string userID)

지정한 사용자 ID에 해당하는 사용자의 사용자 정보를 가져옵니다.

Parameters:

session - 세션 ID

userID - 사용자 정보를 가져올 사용자 ID

CheckPassword

bool CheckPassword(object session, string userID, string password)

사용자의 비밀번호가 맞는지 여부를 체크합니다.

Parameters:

session - 세션 ID

userID - 비밀번호를 체크할 사용자 ID

password - 체크할 비밀번호

GetUserInfos

IUserInfo[] GetUserInfos(object session)

모든 사용자의 사용자 정보를 가져옵니다.

Parameters:

session - 세션 ID

IsAdministrator

bool IsAdministrator(object session, string userID)

지정한 사용자 ID에 해당하는 사용자가 리파지토리의 관리자인지 여부를 확인합니다.

Parameters:

session - 세션 ID

userID - 관리자인지 체크할 사용자 ID

AddAdmin

bool AddAdmin(object sID, string uID, string client_ip)

관리자 권한을 추가하고 추가 성공 여부를 반환합니다.

Parameters:

sID - 로그인한 사용자 ID

uID - 관리자 권한을 추가할 사용자 ID

client_ip - 클라이언트 IP

RemoveAdmin

```
bool RemoveAdmin(object sID, string uID, string client_ip)
```

관리자 권한을 삭제하고 삭제 성공 여부를 반환합니다.

Parameters:

sID - 로그인한 사용자 ID

uID - 관리자 권한을 삭제할 사용자 ID

client_ip - 클라이언트 IP

ModifyAllowip

```
bool ModifyAllowip(object sID, String uID, String allowip, string client_ip)
```

사용자의 허용 IP를 변경하고 변경 성공 여부를 반환합니다.

Parameters:

sID - 로그인한 사용자 ID

uID - 허용 IP를 변경할 사용자 ID

allowip - 허용 IP

client_ip - 클라이언트 IP

IsExternalLogin

```
bool IsExternalLogin
```

외부 API와 연계한 로그인 기능의 사용 여부를 가져옵니다.

IRepositoryUserAuthorityToCategory

카테고리에 대한 사용자 권한 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

AuthorityAccess AccessType

카테고리에 대한 사용자 권한 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- ModifyUserAuthToCategory = 0x00000100
- GetUserAuthToCategory = 0x00000200,
- GetUserInfosOfCategory = 0x00000400
- GetItemInfosInCategoryOfUser = 0x00000800
- GetCategoryInfosOfUser = 0x00020000

Repository

IRepositroy Repository

OZRepositroy 인터페이스를 구현한 클래스를 가져옵니다.

Method Detail

Modify

bool Modify(object session, string userID, string categoryID, Authority permission)

지정한 카테고리 ID에 대한 지정한 사용자 ID의 권한 정보를 변경하고 변경 성공 여부를 반환합니다.

Parameters:

session - 세션 ID
userID - 사용자 ID
categoryID - 카테고리 ID
permission - 권한 정보

GetAuthority

Authority GetAuthority(object session, string userID, string categoryID)

지정한 카테고리 ID에 대한 지정한 사용자 ID의 권한 정보를 가져옵니다.

Parameters:

session - 세션 ID

userID - 사용자 ID

categoryID - 카테고리 ID

GetUserInfos

`IUserInfo[] GetUserInfos(object session, string categoryID, Authority permission)`

지정한 카테고리 ID에 대해 permission 이상의 권한이 부여된 모든 사용자 정보를 가져옵니다.

Parameters:

session - 세션 ID

categoryID - 카테고리 ID

permission - 권한 정보

GetItemInfos

`IItemInfo[] GetItemInfos(object session, string userID, string categoryID, Authority permission)`

지정한 카테고리의 아이템 중에 지정한 사용자의 권한이 permission 이상인 모든 아이템의 목록을 가져옵니다.

Parameters:

session - 세션 ID

userID - 사용자 ID

categoryID - 카테고리 ID

permission - 권한 정보

GetCategoryInfos

`ICategoryInfo[] GetCategoryInfos(object session, string userID, string categoryID, Authority permission)`

지정한 카테고리의 하위 카테고리 중에 지정한 사용자의 권한이 permission 이상인 모든 카테고리의 목록을 가져옵니다.

Parameters:

session - 세션 ID

userID - 사용자 ID

categoryID - 카테고리 ID

permission - 권한 정보

IRepositoryUserAuthorityToItem

아이템에 대한 사용자 권한 정보를 가져오거나 설정하는 인터페이스입니다.

Property Detail

AccessType

AuthorityAccess AccessType

아이템에 대한 사용자 권한 인터페이스에 접근 가능한 메소드를 아래와 같은 숫자 값으로 가져옵니다.

- None = 0x00000000
- All = 0x7FFFFFFF
- ModifyUserAuthToItem = 0x00001000
- GetUserAuthToItem = 0x00002000
- GetUserInfosOfItem = 0x00004000
- GetItemInfosOfUser = 0x00008000

Repository

IRepository Repository

IRepository 인터페이스를 구현한 객체를 가져옵니다.

Method Detail

Modify

bool Modify(object session, string userID, string itemID, Authority permission)

지정한 아이템에 대한 사용자의 권한 정보를 변경하고 변경 성공 여부를 가져옵니다.

Parameters:

session - 세션 ID

userID - 사용자 ID

itemID - 아이템 ID

permission - 권한 정보

GetAuthority

Authority GetAuthority(object session, string userID, string itemID)

지정한 아이템에 대한 사용자의 권한 정보를 가져옵니다.

Parameters:

session - 세션 ID

userID - 사용자 ID

itemID - 권한 정보

GetUserInfos

IUserInfo[] GetUserInfos(object session, string itemID, Authority permission)

지정한 아이템에 대한 사용자의 권한이 permission 이상인 모든 아이템 정보를 가져옵니다.

Parameters:

session - 세션 ID

itemID - 아이템 ID

permission - 권한 정보

GetItemInfos

IItemInfo[] GetItemInfos(object session, string userID, Authority permission)

지정한 사용자의 권한이 permission 이상인 모든 아이템 정보를 가져옵니다.

Parameters:

session - 세션 ID

userID - 아이템 ID

permission - 권한 정보

IUserInfo

사용자 정보를 제공하는 인터페이스입니다.

Property Detail

ID

string ID

해당 사용자의 사용자 ID를 가져옵니다.

Name

string Name

해당 사용자의 사용자 이름을 가져옵니다.

GroupID

string GroupID

해당 사용자가 속한 그룹의 그룹 ID를 가져옵니다.

GroupName

string GroupName

해당 사용자가 속한 그룹의 그룹 이름을 가져옵니다.

Description

string Description

해당 사용자의 설명 내용을 가져옵니다.

UpdateTime

long UpdateTime

해당 사용자를 갱신한 시간 정보를 가져옵니다.

IsLoggedIn

bool IsLoggedIn

해당 사용자의 로그인 여부를 가져옵니다.

SessionID

string SessionID

해당 사용자가 로그인 시 생성된 세션 ID를 가져옵니다.

IsLoginEnabled

bool IsLoginEnabled

해당 사용자의 로그인 가능 여부를 가져옵니다.

LoginIP

`string LoginIP`

해당 사용자가 로그인한 PC의 IP 정보를 가져옵니다.

LastLoginTime

`long LastLoginTime`

해당 사용자가 마지막으로 로그인한 시간 정보를 가져옵니다.

Permission

`Authority Permission`

해당 사용자의 권한 정보를 가져옵니다.

User Defined Log API

UDL(User Defined Log)은 오즈 서버를 통하여 데이터를 조회할 경우 데이터 베이스나 보고서 관련 정보를 임의의 파일이나 DB에 사용자가 정의한 타입의 형식으로 기록할 수 있도록 하기 위한 인터페이스입니다. User Defined Log API는 다양한 [Interface](#)를 제공합니다.

Interface

User Defined Log API에서 제공하는 인터페이스는 아래와 같습니다.

- [IClearQuestLogTarget](#)
- [IDMLogTarget](#)
- [IFileStoreLogTarget](#)
- [IHTTPStoreLogTarget](#)
- [IMonitorLogTarget](#)
- [IProcedureLogTarget](#)
- [IRDBLogTarget](#)
- [ISAPLogTarget](#)
- [ISOAPLogTarget](#)
- [IUDSLogTarget](#)
- [IUserDefinedLogger](#)
- [IUserDefinedLogTarget](#)
- [IXMLLogTarget](#)

IClearQuestLogTarget

OZ UDL에서 Clear Quest 스토어 정보를 사용하기 위한 인터페이스로, Clear Quest 쿼리문이 실행된 후에 로그가 기록됩니다.

Method Detail

ExecuteType

```
public string ExecuteType
```

클라이언트에서 요청한 Clear Quest 실행 타입(ExecuteSQL, ExecuteQuery, ExecuteDynamicQuery)을 가져옵니다.

QuerySubType

```
public string QuerySubType
```

클라이언트에서 요청한 QuerySub 타입(Query, Chart)을 가져옵니다.

Query

```
public string Query
```

클라이언트에서 요청한 쿼리문을 가져옵니다.

QueryExecuteTime

```
public long QueryExecuteTime
```

클라이언트에서 요청한 쿼리문이 실행되는 시간을 가져옵니다.(단위: Milliseconds)

QueryState

```
public string QueryState
```

클라이언트에서 요청한 쿼리문이 정상적으로 실행되었는지 여부(true, false:ErrMsg)를 가져옵니다.

IDMLogTarget

OZ UDL에서 데이터 모듈 관련 정보를 사용하기 위한 인터페이스입니다.

Method Detail

ODIName

```
public string ODIName()
```

쿼리문을 실행한 ODI 이름을 가져옵니다.

DataSetName

```
public string DataSetName()
```

쿼리문을 실행한 데이터 셋 이름을 가져옵니다.

DataStoreName

```
public string DataStoreName()
```

쿼리문을 실행한 데이터 스토어 이름을 가져옵니다.

Parameters

```
public System.Connection.IDictionary Parameters
```

오즈 사용자 지정 파라미터를 Dictionary 형태로 가져옵니다.

IFileStoreLogTarget

OZ UDL에서 파일 스토어 정보를 사용하기 위한 인터페이스로, 파일 스토어의 파일을 읽어온 뒤 로그가 기록됩니다.

Method Detail

FilePath

```
public string FilePath
```

클라이언트에서 요청한 파일 경로를 가져옵니다.

FileAccessTime

```
public long FileAccessTime
```

클라이언트에서 요청한 파일을 가져오는 시간을 가져옵니다.

FileAccessState

```
public string FileAccessState
```

클라이언트에서 요청한 파일이 정상적으로 실행되었는지 여부(true, false:ErrMsg)를 가져옵니다.

IHTTPStoreLogTarget

OZ UDL에서 HTTP 스토어 정보를 사용하기 위한 인터페이스로, HTTP 스토어의 파일을 읽어온 뒤 로그가 기록됩니다.

Method Detail

URL

```
public string URL
```

클라이언트에서 요청한 URL을 가져옵니다.

HTTPAccessTime

```
public long HTTPAccessTime
```

클라이언트에서 요청한 HTTP 경로의 파일을 가져오는 시간을 가져옵니다. (단위: Milliseconds)

HTTPAccessState

```
public string HTTPAccessState
```

클라이언트에서 요청한 HTTP 경로의 파일이 정상적으로 실행되었는지 여부(true, false:ErrMsg)를 가져옵니다.

IMonitorLogTarget

OZ UDL에서 모니터 로그 관련 정보를 사용하기 위한 인터페이스입니다.

Method Detail

ThreadID

```
public int ThreadID()
```

서비스 스레드 아이디를 가져옵니다.

Mark

```
public string Mark()
```

서비스 시작 및 종료 표시(start, end)를 가져옵니다.

ThreadName

```
public string ThreadName()
```

서비스 스레드 이름을 가져옵니다.

ServiceTime

```
public long ServiceTime()
```

서비스 시작 시간과 종료 시간을 가져옵니다. (단위: Milliseconds)

FreeMemory

```
public long FreeMemory()
```

사용 가능한 메모리 크기를 가져옵니다.(단위: byte)

TotalMemory

```
public long TotalMemory()
```

전체 JVM 메모리 크기를 가져옵니다.(단위: byte)

ServiceCode

```
public int ServiceCode()
```

서비스 구분 코드를 가져옵니다.

MARK가 "end"일 경우에만 값이 리턴되며 "start"일 경우 -1을 리턴합니다.

ServiceStatus

```
public int ServiceStatus()
```

서비스 상태 코드를 가져옵니다.

MARK가 "end"일 경우에만 값이 리턴되며 "start"일 경우 -1을 리턴합니다.

성공일 경우 "9001", 실패일 경우 "9002"를 리턴합니다.

ServiceParameter

```
public string ServiceParameter()
```

클라이언트에서 서비스 요청 시 전송되는 파라미터 값을 가져옵니다.

MARK가 "end"일 경우에만 값이 리턴되며 "start"일 경우 -1을 리턴합니다.

DBSessionID

```
public string DBSessionID()
```

DBMS 세션 ID를 가져옵니다.

MARK가 "end"일 경우에만 값이 리턴되며 "start"일 경우 -1을 리턴합니다.

ExecuteTime

```
public string ExecuteTime()
```

서비스 처리 시 소요된 시간을 가져옵니다.(단위: Milliseconds)

MARK가 "end"일 경우에만 값이 리턴되며 "start"일 경우 -1을 리턴합니다.

ErrorCode

```
public string ErrorCode()
```

에러 발생 시 에러 코드를 가져옵니다.

ErrorMsg

```
public string ErrorMsg()
```

에러 발생 시 에러 메시지를 가져옵니다.

ErrorStackTrace

```
public string ErrorStackTrace()
```

에러 발생 시 Stack Trace를 가져옵니다.

DBConns

```
public string DBConns()
```

Alias별 DB사용갯수를 가져옵니다.

IProcedureLogTarget

OZ UDL에서 프로시저 데이터 관련 정보를 사용하기 위한 인터페이스로, 프로시저가 실행된 후 로그가 기록됩니다.

Method Detail

Alias

```
public string Alias
```

실행한 ODI에서 사용한 커넥션 풀의 DB 앨리어스 명을 가져옵니다.

ProcedureName

```
public string ProcedureName
```

클라이언트에서 요청한 프로시저 이름을 가져옵니다.

ParameterList

```
public IList ParameterList
```

클라이언트에서 요청한 프로시저 파라미터를 IList 형태로 가져옵니다.

ProcedureExecuteTime

```
public long ProcedureExecuteTime
```

클라이언트에서 요청한 프로시저가 수행되는 시간을 가져옵니다.(단위: Milliseconds)

ProcedureState

public string ProcedureState

클라이언트에서 요청한 프로시저가 정상적으로 실행되었는지 여부(true, false:ErrMsg)를 가져옵니다.

IRDBLogTarget

OZ UDL에서 RDB 스토어 관련 정보를 사용하기 위한 인터페이스로, 쿼리문이 실행된 후 로그가 기록됩니다.

Method Detail

Alias

public string Alias

실행한 ODI에서 사용한 커넥션 풀의 DB 앨리어스 명을 가져옵니다.

데이터 스토어에서 앨리어스를 사용하지 않는 경우 커넥션 풀의 키 값을 가져옵니다.

Query

public string Query

클라이언트에서 요청한 쿼리문을 가져옵니다.

PreparedQueryValue

public IList PreparedQueryValue

컴파일된 질의문을 사용할 경우 사용된 값을 가져옵니다.

QueryState

public string QueryState

클라이언트에서 요청한 쿼리문이 정상적으로 실행되었는지 여부를 가져옵니다.

QueryExecuteTime

public long QueryExecuteTime

클라이언트에서 요청한 쿼리문이 수행되는 시간을 가져옵니다.(단위: Milliseconds)

IsPrepared

```
public bool IsPrepared
```

컴파일된 질의문의 사용 여부를 가져옵니다.

QueryType

```
public QueryType QueryType
```

클라이언트에서 요청하여 실행된 쿼리문의 타입(select, insert, update, delete)을 가져옵니다.

ISAPLogTarget

OZ UDL에서 SAP 스토어 정보를 사용하기 위한 인터페이스로, 함수가 실행된 후에 로그가 기록됩니다.

Method Detail

FunctionName

```
public string FunctionName
```

클라이언트에서 요청한 함수 이름을 가져옵니다.

FunctionType

```
public string FunctionType
```

클라이언트에서 요청한 함수 타입을 가져옵니다.

InputParameters

```
public IDictionary InputParameters
```

클라이언트에서 요청한 입력 파라미터를 IDictionary 형태로 가져옵니다.

ResultSetTypes

```
public string ResultSetTypes
```

클라이언트에서 요청한 ResultSet의 타입(Structure, Table, SimpleFields)을 가져옵니다.

FunctionExecuteTime

```
public long FunctionExecuteTime
```

클라이언트에서 요청한 함수가 실행되는 시간을 가져옵니다.(단위: Milliseconds)

FunctionState

```
public string FunctionState
```

클라이언트에서 요청한 함수가 정상적으로 실행되었는지 여부(true, false:ErrMsg)를 가져옵니다.

ISOAPLogTarget

OZ UDL에서 SOAP 스토어 정보를 사용하기 위한 인터페이스로, Service Response 후에 로그가 기록됩니다.

Method Detail

ServiceName

```
public string ServiceName
```

클라이언트에서 요청한 서비스 이름을 가져옵니다.

Port

```
public string Port
```

클라이언트에서 요청한 서비스의 포트를 가져옵니다.

Operation

```
public string Operation
```

클라이언트에서 요청한 서비스의 오퍼레이션을 가져옵니다.

EndPoint

```
public string EndPoint
```

클라이언트에서 요청한 서비스를 호출한 EndPoint를 가져옵니다.

RequestXML

```
public string RequestXML
```

클라이언트에서 요청한 XML 내용을 가져옵니다.

ServiceExecuteTime

```
public long ServiceExecuteTime
```

클라이언트에서 요청한 서비스가 실행되는 시간을 가져옵니다.(단위: Milliseconds)

ServiceState

`public string ServiceState`

클라이언트에서 요청한 SOAP의 서비스가 정상적으로 실행되었는지 여부(true, false:ErrMsg)를 가져옵니다.

IUDSLogTarget

OZ UDL에서 사용자 데이터 스토어 정보를 사용하기 위한 인터페이스로, 사용자 데이터 스토어 실행문이 수행된 후 로그가 기록됩니다.

Method Detail

ExecuteCommand

`public string ExecuteCommand`

클라이언트에서 요청한 실행문을 가져옵니다.

QueryType

`public QueryType QueryType`

클라이언트에서 요청하여 실행된 쿼리문의 타입(select, insert, update, delete)을 가져옵니다.

CommandExecuteTime

`public long CommandExecuteTime`

클라이언트에서 요청한 실행문이 수행되는 시간을 가져옵니다.(단위: Milliseconds)

CommandState

`public string CommandState`

클라이언트에서 요청한 실행문이 정상적으로 실행되었는지 여부(true, false:ErrMsg)를 가져옵니다.

IUserDefinedLogger

OZ UDL에서 사용자 정의 로그를 사용하기 위한 인터페이스입니다.

Method Detail

Init

```
public void Init(OZConnection con, OZLog log)
```

오즈 서버 구동 시 사용자 정의 로그를 남길 대상을 정의합니다.

Parameters:

con - DB에 사용자 정의 로그를 기록할 경우 사용할 DB 커넥션 정보 및 기록할 로그 형식

log - 파일에 사용자 정의 로그를 기록할 경우 사용할 파일 정보 및 기록할 로그 형식

Trace

```
public void Trace(IUserDefinedLogTarget target, OZConnection con, OZLog log)
```

로그에 기록할 타겟 내용을 정의합니다.

Parameters:

target - 사용자 정의 로그 타겟

con - DB에 사용자 정의 로그를 기록할 경우 사용할 DB 커넥션 정보 및 기록할 로그 형식

log - 파일에 사용자 정의 로그를 기록할 경우 사용할 파일 정보 및 기록할 로그 형식

DbAlias

```
public string DbAlias
```

DB에 사용자 정의 로그를 기록할 경우 오즈 서버 커넥션 풀의 앨리어스 명을 가져옵니다.

IUserDefinedLogTarget

OZ UDL에서 클라이언트 정보를 사용하기 위한 인터페이스입니다.

Method Detail

IPAddress

```
public string IPAddress()
```

클라이언트가 요청한 오즈 서버 IP 주소를 가져옵니다

HttpRequest

```
public HttpRequest HttpRequest()
```

클라이언트에서 요청한 HttpRequest를 가져옵니다.

HttpContext

```
public HttpContext HttpContext()
```

클라이언트에서 HttpContext를 가져옵니다.

UserID

```
public string UserID()
```

클라이언트에서 쿼리문을 실행한 오즈 서버 사용자 ID를 가져옵니다

IXMLLogTarget

OZ UDL에서 XML 스토어 정보를 사용하기 위한 인터페이스로, XML 파일을 읽어온 뒤 로그가 기록됩니다

.

Method Detail

URL

```
public string URL
```

OZ UDL에서 XML 스토어 정보를 사용하기 위한 인터페이스로 XML 파일을 읽어온 뒤 로그가 기록됩니다.

URLAccessTime

```
public long URLAccessTime
```

클라이언트에서 요청한 URL의 파일을 가져오는 시간을 가져옵니다.(단위: Milliseconds)

URLAccessState

```
public string URLAccessState
```

클라이언트에서 요청한 URL의 파일이 정상적으로 실행되었는지 여부(true, false:ErrMsg)를 가져옵니다.

User Defined Resource API

UDR(User Defined Resource)은 보고서의 리소스 필드를 생성할 때 리소스 필드 값을 리포트 디자이너에서 일일이 설정하지 않고 데이터베이스에 입력된 리소스 정보로 한꺼번에 설정할 수 있도록 하는 인터페이스입니다.

User Defined Resource API는 다양한 [Class](#)와 [Interface](#)를 제공합니다.

Class

User Defined Resource API에서 제공하는 클래스는 아래와 같습니다.

- [OZUDRInfo](#)
- [OZUDRResult](#)
- [Resource](#)

OZUDRInfo

OZ UDR 정보를 제공하는 클래스입니다.

Constructor Detail

OZUDRInfo

```
public OZUDRInfo(string category, string reportname, string[] language, Resource[] resource)
```

OZ UDR 정보를 가져옵니다.

Parameters:

category - 보고서 카테고리

reportname - 보고서 이름

language - 언어 리스트

resource - 리소스 정보 객체

Property Detail

Category

```
public string Category
```

카테고리 이름

Reportname

`public string Reportname`

리포트 이름

Language

`public string[] Language`

언어 리스트

Resource

`public Resource[] Resource`

리소스

OZUDRResult

OZ UDR 실행 결과를 제공하는 클래스입니다.

Constructor Detail

OZUDRResult

`public OZUDRResult(bool isUpdate, OZUDRInfo info)`

OZ UDR 실행 결과를 가져옵니다.

Parameters:

isUpdate - 업데이트 여부

info - 사용자가 정의한 리소스 정보를 다시 저장한 객체

Property Detail

Update

`public bool Update`

업데이트 여부

UDRInfo

`public OZUDRInfo UDRInfo`

사용자가 정의한 리소스 정보를 다시 저장한 객체

Resource

리소스 정보를 제공하는 클래스입니다.

Constructor Detail

Resource

```
public Resource(string name, string desc, string[] value)
```

리소스 정보를 가져옵니다.

Parameters:

name - 리소스 필드 이름

desc - 리소스 설명

value - 언어별 리소스 값. OZUDRInfo language index와 같은 index 값이어야 함

Property Detail

Name

```
public string Name
```

리소스 필드 이름

Desc

```
public string Desc
```

리소스 설명

Value

```
public string[] Value
```

언어별 리소스 값

DescHash

```
public Hashtable DescHash
```

리소스 설명을 파싱할 Hash에 key, value 형태로 가지고 있는 객체

Interface

User Defined Resource API에서 제공하는 인터페이스는 아래와 같습니다.

- [IOZUserDefinedResource](#)

IOZUserDefinedResource

리소스 동기화 기능을 제공하는 인터페이스입니다.

Method Detail

SyncResource

```
public OZUDRRResult SyncResource(OZConnection ozconn, OZLog cat, OZUDRInfo info)
```

리소스를 동기화합니다.

Parameters:

ozconn - DB의 Connection

cat - 서버 로그에 남을 카테고리

info - 보고서에 있는 리소스 정보 객체