

# 로컬 캐시

## 개요

다운로드 받은 파일을 클라이언트 장비에 임시로 캐싱

## 기능

- 동일 URL에 다운로드 요청 시, 한 번만 다운로드 하도록 처리
- bytes[] 및 Texture2D 로드 지원
- 캐시파일 관리 및 무결성 검사
- 메모리 관리 (C# ArrayPool 사용)

## API

네임스페이스 : LocalCache

### Static API

클래스명 : Cache

API	용도
UniTask<byte[]> <b>LoadBytesAsync</b> (string fileName, string url, LoadCacheCallbacks loadCacheCb, DownloadCallbacks downloadCb, DownloadRequestCallbacks downloadReqCb)	비동기 다운로드 (byte[])
UniTask <b>LoadTexture2DAsync</b> (string fileName, string url)	비동기 다운로드 (Texture2D)
void <b>PurgeCache</b> (string fileName)	캐시 삭제
void <b>PurgeCache</b> (eCacheType type)	eCacheType 내 캐시 모두 삭제
void <b>DeleteAllCache</b> ()	모든 캐시 삭제
void <b>PurgeTemp</b> (string fileName)	임시 파일 삭제
void <b>PurgeTemp</b> (eCacheType type)	eCacheType 내 임시 파일 모두 삭제
void <b>DeleteAllTemp</b> ()	모든 임시파일 삭제
void <b>SetCacheType</b> (eCacheType type)	캐시 타입 선택
void <b>PrintInfo</b> ()	캐시 정보 출력

eCacheType

이름	종류
DEFAULT	기본 캐시 타입

Request API

클래스명 : Cache.Request

API	용도
Request <b>NewRequest</b> (string fileName, string url)	새로운 다운로드 요청을 위한 Request 생성 (static)
Request <b>WithLoadCacheCallbacks</b> (LoadCacheCallbacks cb)	캐시 로드 콜백 등록
Request <b>WithDownloadCallbacks</b> (DownloadCallbacks cb)	다운로드 콜백 등록
Request <b>WithDownloadRequestCallbacks</b> (DownloadRequestCallbacks cb)	다운로드 요청 콜백 등록
UniTask<byte[]> <b>LoadBytesAsync</b> ()	비동기 다운로드 (byte[])
UniTask <b>LoadTexture2DAsync</b> ()	비동기 다운로드 (Texture2D)

Callbacks

클래스명 : Callbacks

Delegates

이름	파라미터 설명
void <b>OnProgress</b> (int currentRead, long totalRead, long totalSize)	currentRead : 현재 읽은 데이터 (bytes) totalRead : 총 읽은 데이터 (bytes) totalSize : 총 파일 크기 (bytes)
void <b>OnHttpStatus</b> (HttpStatusCode statusCode)	statusCode : HttpResponseMessage 상태 코드
void <b>OnDownloadComplete</b> (string hash, long totalSize)	hash : 다운로드 받은 파일의 해시값 (0이면 해시가 유효하지 않음) totalSize : 총 파일 크기 (bytes)
void <b>OnLoadComplete</b> (bool success, long totalSize, Memory bytes)	success : 성공/실패 totalSize : 총 파일 크기 (bytes) bytes : 데이터 (bytes)
void <b>OnDownloadHandler</b> (DownloadHandler handler)	handler : 다운로드 제어, 상태 확인 등을 위한 클래스 (LocalCache.DownloadHandler)

이름	파라미터 설명
void <b>OnDownloadRequest</b> (DownloadRequest request)	request : 요청 큐로 전달된 다운로드 요청 상태에 대한 모니터링 클래스 (LocalCache.DownloadRequest)

### Callbacks

이름	설명
OnProgress	로컬 캐시 파일 로드 진행 상황
OnLoadComplete	로컬 캐시 파일 로드 완료
OnProgress	다운로드 진행 상황
OnHttpStatus(OnFailed)	다운로드 실패
OnDownloadComplete	다운로드 성공
OnDownloadHandler	다운로드 제어
OnDownloadRequest	다운로드 요청

### 사용 예시

```

// 비동기 bytes 로드
var bytes = await Cache.LoadBytesAsync(_fileName, _url);
if (bytes != null)
{
    var tex = new Texture2D(1, 1);
    tex.LoadImage(bytes);
}

// 비동기 Texture2D 로드
var tex = await Cache.LoadTexture2DAsync(_fileName, _url); // tex : Texture2D
_image.texture = tex;

// Request 로드 Texture2D
var loadCacheCallbacks = new Callbacks.LoadCacheCallbacks
{
    OnProgress = OnCacheLoadProgress,
    OnLoadComplete = OnCacheLoadComplete,
};

var downloadCallbacks = new Callbacks.DownloadCallbacks
{
    OnProgress = OnDownloadProgress,
    OnFailed = OnFailed,
    OnDownloadComplete = OnDownloadComplete,
    OnDownloadHandler = OnDownloadHandler
};

```

```
var downloadRequestCallbacks = new Callbacks.DownloadRequestCallbacks
{
    OnDownloadRequest = OnDownloadRequest,
};

var tex= await Cache.Request.NewRequest(_fileName, _url)
    .WithLoadCacheCallbacks(loadCacheCallbacks)
    .WithDownloadCallbacks(downloadCallbacks)
    .WithDownloadRequestCallbacks(downloadRequestCallbacks)
    .LoadTexture2DAsync();
_image.texture = tex;
```

## 텍스처 캐시 연동

텍스처 캐시 기능의 일부로 로컬캐시를 사용

