iws.loader

# 특징

* 다양한 이미지 포멧 로딩 (Tiff, Jpeg, Jpeg2000, Png, JBig, BMP, gif).
* 이미지로딩 결과로 base64 스트링 제공. 또는 Canvas에서 직접 사용 가능한 픽셀 데이터 제공.

# 제공

dist/

|  |  |
| --- | --- |
| Iws.laoder.js | |
| Engine/ | IWSImageLib\_min.js |
|  |  |

# 설치

Include files:

|  |
| --- |
| <script src="./engine/IWSImageLib\_min.js"></script>  <script src="./iws.loader.js"></script> |

# 사용법

|  |
| --- |
| var loader = new iws.loader(arrayBuffer);  arrayBuffer는   * FileReader의 readAsArrayBuffer를 사용하여 얻은 바이너리 배열, 또는 * XMLHttpRequest 로 얻은 responseType 이 “arraybuffer”인 바이너리 배열 등이 올 수 있다.   Function addImgElement(src){  /\* src를 활용. \*/  }  for(var i=0; i<loader.getCount(); i++){  loader.getSrc(i+1, addImgElement);  }  loader.dispose();  또는  Function addCanvasElement(width, height, data){  /\* width, height, data 를 활용 \*/  }  for(var i=0; i<loader.getCount(); i++){  loader.getPage(i+1, addCanvasElement);  }  loader.dispose(); |

# Methods

* getCount()

: 페이지 수를 리턴 합니다. 멀티Tiff 파일의 경우는 페이지 수가 1개 이상일 수 있습니다.

* getSrc(nPage, callback)

: 원하는 페이지를 ‘img’ Element의 ‘src’에 넣을 수 있는 base64:png 형태의 데이터를 callback함수를 통하여 얻습니다.

* + nPage
    - Type : Number
    - 로딩할 페이지 번호 ( 1부터 시작)
  + callback
    - Type : Function
    - function(src)
      * src : Base64 스트링

{

/\*원하는 처리… \*/

}

* getPage(nPage, callback)

: 원하는 페이지를 ‘canvas’에 셋팅할 수 있는 데이터를 callback함수를 통하여 얻습니다.

* + nPage
    - Type : Number
    - 로딩할 페이지 번호 ( 1부터 시작)
  + callback
    - Type : Function
    - function( w, h, data )
      * w : 이미지 가로 사이즈
      * h : 이미지 세로 사이즈.
      * data : canvas.getContext('2d').putImageData의 파라미터 값으로 사용할 수 있는 데이터

{

/\*원하는 처리… \*/

}

* dispose()

: 이미지 로딩이 완료 되었을 경우 **필수로 호출**해야 함.

# License