

cMatrixf

Constructors

Constructor	Description
cMatrixf()	Creates a matrix with default values.
cMatrixf(cVector4f, cVector4f, cVector4f, cVector4f)	Creates a matrix using the given vectors as column data.
cMatrixf(float, float, float, float, float, float, float, float, float, float, float, float, float, float, float, float)	Creates a matrix using the given values as cell data.

Fields

cMatrixf has no public fields.

Functions

Return Type	Function Name	Parameters	Description
float	GetElement	uint64, uint64, const	
cVector3f	GetRight	const	
void	SetRight	const cVector3f &in avVec	
cVector3f	GetUp	const	
void	SetUp	const cVector3f &in avVec	
cVector3f	GetForward	const	
void	SetForward	const cVector3f &in avVec	
cVector3f	GetTranslation	const	
void	SetTranslation	const cVector3f &in avTrans	
void	SetRotation	float afXX, float afXY, float afXZ, float afYX, float afYY, float afYZ, float afZX, float afZY, float afZZ	
void	SetRotation	const cMatrixf &in a_mtxRot	
cMatrixf	GetRotation	const	
cMatrixf	GetTranspose	const	

Constructors

Constructor	Description
cMatrixf()	Creates a matrix with default values.
cMatrixf(cVector4f, cVector4f, cVector4f, cVector4f)	Creates a matrix using the given vectors as column data.
cMatrixf(float, float, float, float, float, float, float, float, float, float, float, float, float, float)	Creates a matrix using the given values as cell data.

Fields

cMatrixf has no public fields.

Functions

Return Type	Function Name	Parameters	Description
float	GetElement	uint64, uint64, const	
cVector3f	GetRight	const	
void	SetRight	const cVector3f &in avVec	
cVector3f	GetUp	const	
void	SetUp	const cVector3f &in avVec	
cVector3f	GetForward	const	
void	SetForward	const cVector3f &in avVec	
cVector3f	GetTranslation	const	
void	SetTranslation	const cVector3f &in avTrans	
void	SetRotation	float afXX, float afXY, float afXZ, float afYX, float afYY, float afYZ, float afZX, float afZY, float afZZ	
void	SetRotation	const cMatrixf &in a_mtxRot	
cMatrixf	GetRotation	const	
cMatrixf	GetTranspose	const	

Remarks

To retrieve a value from a matrix, use the GetElement function above. The parameters for the GetElement function use the format GetElement(columnIndex, rowIndex).

```
cMatrixf m1(cVector4f(, 1, 2, 3),
            cVector4f(4, 5, 6, 7),
            cVector4f(8, 9, 10, 11),
            cVector4f(12, 13, 14, 15));
```

```
float f = m1.GetElement(1, 2);  
  
// value of f: 9
```

To do matrix computations, use the [cMath_MatrixXXX](#) family of functions.

```
cMatrixf m1(cVector4f(1, 1, 1, 1),  
            cVector4f(2, 2, 2, 2),  
            cVector4f(3, 3, 3, 3),  
            cVector4f(4, 4, 4, 4));  
  
cMatrixf m2(cVector4f(5, 5, 5, 5),  
            cVector4f(6, 6, 6, 6),  
            cVector4f(7, 7, 7, 7),  
            cVector4f(8, 8, 8, 8));  
  
cMatrixf m3 = cMath_MatrixMul(m1, m2);  
  
// value of m3: { 26, 26, 26, 26,  
//                52, 52, 52, 52,  
//                78, 78, 78, 78,  
//                104, 104, 104, 104 }
```

From:

<https://wiki.frictionalgames.com/> - **Frictional Game Wiki**

Permanent link:

<https://wiki.frictionalgames.com/hpl3/community/scripting/classes/cmatrixf>

Last update: **2017/10/23 00:50**

