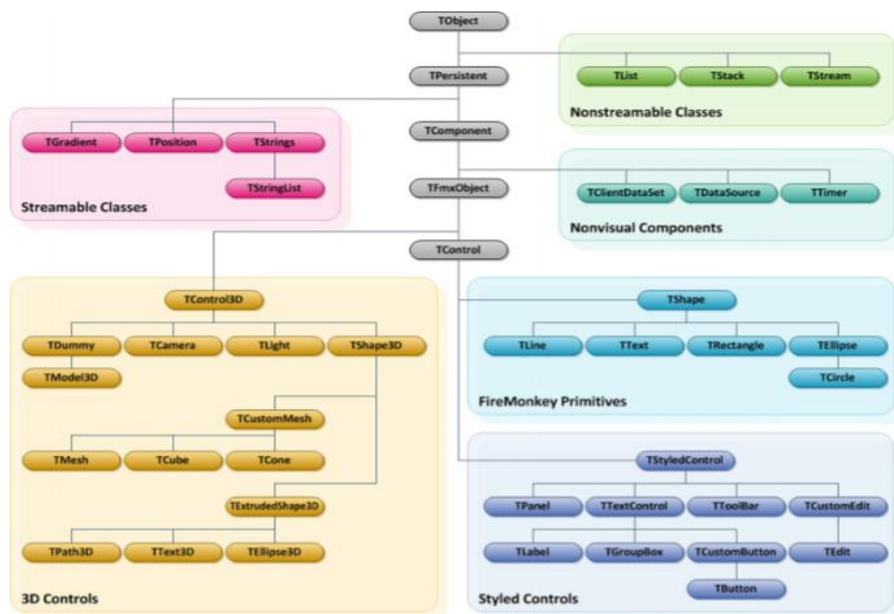


화면에 3D 적용하기



파이어몽키 3D 폼은 대화형 사용자 인터페이스에 대한 조명, 텍스처, 애니메이션 등을 사용하여 완전하게 GPU 구동 3D 표면을 제공합니다. 파이어몽키 3D를 사용하려면 컴퓨터가 그래픽 처리 장치(GPU)를 가지고 있어야 합니다. 대부분의 컴퓨터는 3D 지원 그래픽 칩이 마련되어 있습니다.

3D Forms



■ TForm3D

3d 오브젝트나 컨트롤을 포함 할 수 있는 화면

■ TViewPort3D

자체는 3d 컴포넌트이나 그 안에 2D에 올려서 사용할 수 있다.

파이어몽키 3D 좌표 시스템

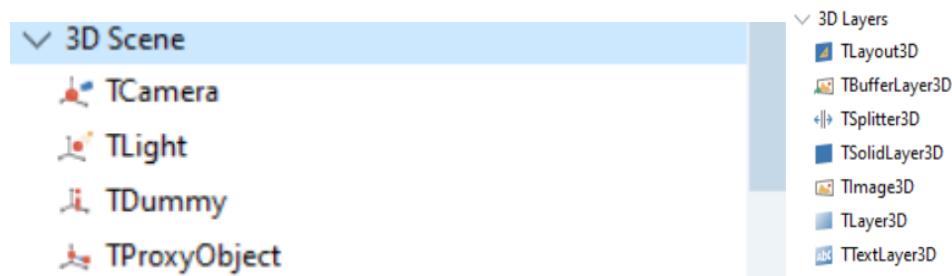
3D 오브젝트나 컨트롤의 Position 속성은 오브젝트의 위치에 대한 X, Y 및 Z 값을 포함합니다. 3D

오브젝트에서, 위치는 3D 공간에 대해 상대적입니다. 3D 컨트롤에서, 위치는 부모의 위치를 기준으로 합니다. X, Y 및 Z의 값은 단일 부동 소수점 타입입니다.

- 양수 X – 3D 오브젝트를 부모의 중앙에서 오른쪽으로
- 양수 Y – 3D 오브젝트를 부모의 중앙에서 아래로
- 양수 Z – 3D 오브젝트는 사용자 (화면에)에서 먼 거리에 있습니다

3D 오브젝트들

파이어몽키는 3D 오브젝트 몇 가지 종류를 제공합니다:



카메라

3D 공간의 모든 뷰는 카메라에 의해 제어됩니다. 카메라의 위치와 방향(3D 회전)은 보는 것을 결정합니다. 폼 디자이너에서 사용되는 디자인 카메라가 항상 있으며, 런타임에서 디폴트입니다. 디자인 카메라는 직접 음수-Z축 위에(음수-Y 방향), X-Z 평면에 수직으로, 약간 아래쪽 각진, 위치 0,0,0은 보기의 중심에 있습니다. 다른 카메라를 사용하려면 TViewport3D 또는 TForm3D의 UsingDesignCamera 속성을 False로 설정하고 Camera 속성에 TCamera를 할당합니다. 씬은 여러 카메라를 가질 수 있습니다. Camera 속성에 다른 하나를 할당 한 후에 수동으로 다시 Repaint를 호출해야 합니다. 볼 수 있는 볼륨은 카메라 앞에서 가까운 평면으로 절두체(frustum)입니다. 카메라와 그 앞 평면 사이에 어떤 것도 보기에서 잘립니다. 시야의 각도는 수직으로 고정되어 있습니다: 뷰포트 / 폼을 더 높게 하는 것은 모든 것을 더 크게 합니다 반면 더 넓게 하면 측면이 더 넓게 보여집니다. 위치와 크기의 단위는 시야의 스케일을 기준으로 합니다. 화면 투영(프로젝션) 카메라에 의해 결정되는 오브젝트의 관점뿐만 아니라, 오브젝트는 3D 게임의 고정 상태 표시기처럼 카메라 포인트에 상관없이 보여지도록 설정할 수 있습니다. 이것은 Projection 속성을 기본 pjCamera에서 pjScreen으로 변경하여 수행됩니다: λ Position.Z 에서 제로값: o XY 0,0는 2D와같이 왼쪽 상단 모서리에 있습니다. (3D 위치

TLight

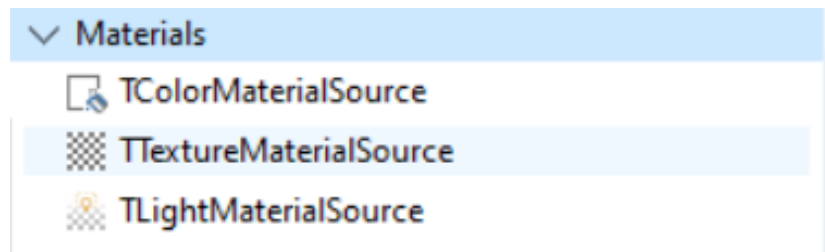
일반적으로 3D 개체는 빛이 추가되지 않은 한 특징이 없는 검은 색 덩어리입니다. 하나 이상의 TLight 오브젝트는 3D 오브젝트의 LightType 속성에 따라 3D 공간에서 빛을 정의합니다:

- ItDirectional - 방향 표시등이 주어진 각도에서 일정합니다. 태양의 빛은 일반적인 아날로그입니다: 지구의 지역화된 곳으로부터 멀리 떨어진 - 마일이 아닌 야드로 - 모든 아이템들이 동시에 비춰집니다. 방향 표시등의 위치는 효과와 관련이 없습니다. (폼 디자이너에서 클릭하려고 하면 그 위치는 연관이 있습니다) 중요한 것은, RotationAngle에 의해 정의된 빛 포인트의 방향과 부모의 RotationAngle입니다.
- ItPoint - 포인트 등은 노출 된 전구 (줄기가 없는)같은 것입니다. 그것은 모든 방향으로 방출하고 거리가 떨어져 있습니다. RotationAngle는 아무런 영향을 미치지 않습니다. 중요한 것은 부모의 위치 및 회전에 영향을 받는 위치입니다
- ItSpot -스팟 라이트는 위치와 회전 모두에 따라 달라집니다, 그리고 거리와 떨어집니다. 3D 공간에서 평면 2D 개체를 필요하지 않으며, 빛의 영향을 받지 않습니다. 그들은 추가된 3D 관점과 같이 평소와 같이 나타납니다.

물질(Material)

3D 개체의 표면은 그 물질에 의해 정의됩니다. 원시 3D 개체는 하나의 Material 속성을 가지고 있습니다.

- ColorMaterialSource
- TextureMaterialSource
- LightMaterialSource

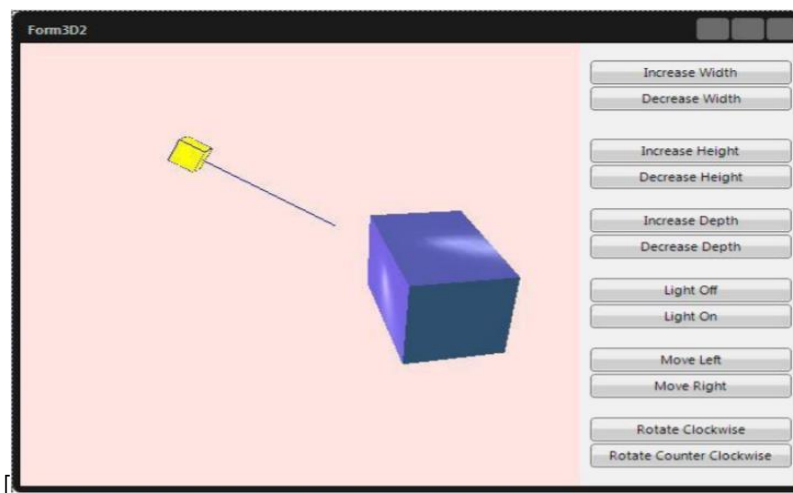


2D 와 3D 혼합

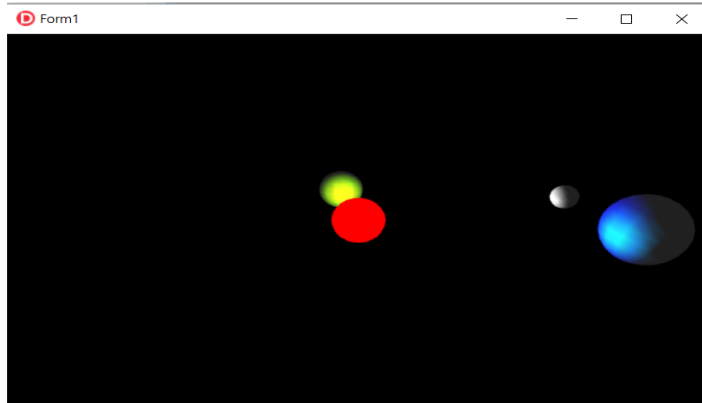
3D 오브젝트는 3D 컨테이너에 배치합니다. 2D 컨테이너(TForm과 같은)에 직접 배치 하면 3D 개체는(TCube와 같은) 렌더링 되지 않습니다. 반대로, 3D 컨테이너에(TForm3D과 같은) 직접 배치한 다면 2D 오브젝트는(TButton과 같은) 렌더링 되지 않습니다. 두 클래스는 2D 및 3D 콘텐츠의 혼합 중첩을 허용 합니다:

- TLayer3D는 2D 콘텐츠가 포함 된 3D 개체입니다. 3D 공간에 존재 하는 유리의 직사각형 시트 같습니다.
- TViewport3D는 3D 콘텐츠가 포함 된 2D 개체입니다. TForm3D 마찬가지로 3D 공간에 "창"입니다. 이러한 컨테이너들을 중첩할 수 있습니다. 예를 들어, 아래와 같은 개체 계층 구조를 가질 수 있습니다: 또한, 몇 클래스는 2D 콘텐츠를 직접 3D로 호스팅 합니다:

- > TForm3D
 - > TCube
 - > TLayer3D
 - > TButton
 - > TViewport3D
 - > TCube



[샘플 따라하기] 3D 태양계 만들기



1. 새로운 모바일 프로젝트를 생성하되 3D 어플리케이션으로 시작한다.
2. 그리고 화면을 가로로 고정하고 폼의 컬러를 검정으로 변경한다.
3. 폼위에 TGrid3d 컴포넌트를 놓고 다음과 같이 속성을 변경한다.

컴포넌트	속성	값
TGrid3d	height	15
	Width	20
	RotaionAngle.x	90

4. TMaterialSource 컴포넌트를 3개 내려 놓고 각각의 diffuse 속성을 Blue, White, SpringGreen으로 설정한다.
5. TSphere를 배치하고 다음과 속성을 변경한다.

컴포넌트	속성	값
TSphere	Depth	2
	Height	2
	Width	2
	Name	SunSphere

6. TLight를 Sphere 하위에 배치하고 LightType을 Point로 설정한다.
7. 이제 TDummy를 사용해 보자. TDummy를 배치하고 Name 속성을 'EDummy'로 변경한다. 이 곳에 지구와 달을 배치하도록 한다.
8. TCylinder 컴포넌트를 EDummy의 차일드 오브젝트로 배치하고 다음과 같이 속성을 설정한다.

컴포넌트	속성	값
TCylinder	Depth	3
	Height	3
	Width	3
	MaterialSource	LightMaterialSource1
	Name	EarthSphere
	RotationAngle,Z	23.4 (지구의 자전축 기울기)
	Position,X	10

9. TFloatAnimation을 EDummy에 배치하고 속성을 설정한다.

컴포넌트	속성	값
TFloatAnimation	Duration	12 (공전주기 12개월)
	Enabled	True
	Loop	True
	PropertyName	RotationAngle,Y
	StopValue	360

10. TDummy를 하나 더 배치하되 EarthSphere의 자식 오브젝트로 배치하고 이름은 'Mdummy'로 설정한다. 달은 지구에 속해 있으므로 이곳에 달 오브젝트를 배치한다.

11. TCylinder 컴포넌트를 Mdummy의 차일드 오브젝트로 배치하고 다음과 같이 속성을 설정한다.

컴포넌트	속성	값
TCylinder	Depth	1
	Height	1
	Width	1
	MaterialSource	LightMaterialSource2
	Name	MoonSphere
	Position,X	-3

12. TFloatAnimation을 MDummy에 배치하고 속성을 설정한다.

컴포넌트	속성	값
TFloatAnimation	Duration	1 (공전주기 1개월)
	Enabled	True
	Loop	True
	PropertyName	RotationAngle,Y
	StopValue	360

13. 태양계를 공전하는 지구와 달이 잘 표현되고 있는지 확인해 보겠습니다.

14. VDummy는 금성을 표현한 것으로 지구와 같은 방식으로 하나 더 추가해보겠다. 금성의 공전 주기는 7.5개월이므로 FloatAnimation의 Duration 값을 7.5로 설정한다.