|  |
| --- |
| 07 3D 게임 프로그래밍을 위한 기초 수학 - 7. 삼각함수(원, 라디안, 파이)  3D 게임 프로그래밍을 위한 기초 수학에 대한 연재 강좌를 시작합니다.  저도 수학에 대해서 잘 모르지만 공부하면서 알게된 지식을 공유하기 위한 목적으로 올립니다.  벡터와 삼각함수 부분을 연재해 나갈 계획이며,  수학에 기초가 약한 분들을 대상으로 쉽게 쓰려고 합니다.  따라서 이미 베테랑이신 분들은 다 아는 내용일지도 모릅니다.  혹시 내용중 잘못된 부분에 대한 지적이나 의견을 주시면 너무 감사하겠습니다.  적극적인 피드백 부탁드립니다.^^  지난 강좌  [3D 게임 프로그래밍을 위한 기초 수학 - 1. 시작](http://lab.gamecodi.com/board/zboard.php?id=GAMECODILAB_Lecture_series&no=121&z=)  [3D 게임 프로그래밍을 위한 기초 수학 - 2. 벡터, 벡터의 덧셈](http://lab.gamecodi.com/board/zboard.php?id=GAMECODILAB_Lecture_series&no=122&z=)  [3D 게임 프로그래밍을 위한 기초 수학 - 3. 벡터의 뺄셈, 벡터와 스칼라의 곱셈](http://lab.gamecodi.com/board/zboard.php?id=GAMECODILAB_Lecture_series&no=123&z=)  [3D 게임 프로그래밍을 위한 기초 수학 - 4. 벡터의 내적](http://lab.gamecodi.com/board/zboard.php?id=GAMECODILAB_Lecture_series&no=125&z=)  [3D 게임 프로그래밍을 위한 기초 수학 - 5. 벡터의 외적](http://lab.gamecodi.com/board/zboard.php?id=GAMECODILAB_Lecture_series&no=126&z=0)  [3D 게임 프로그래밍을 위한 기초 수학 - 6. 삼각함수(sin, cos, tan)](http://lab.gamecodi.com/board/zboard.php?id=GAMECODILAB_Lecture_series&no=127&z=1) |

**원과 삼각함수**

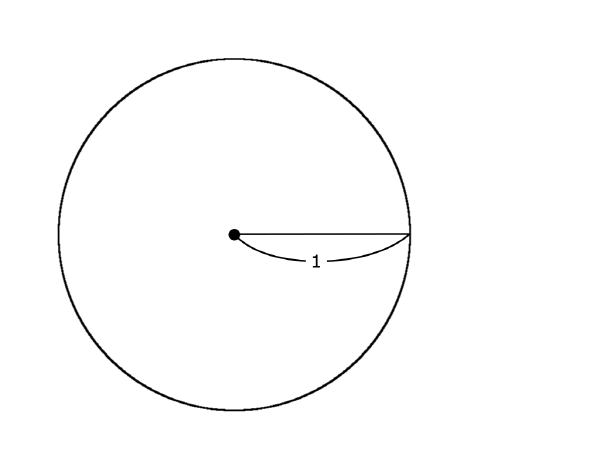
이제 삼각함수가 어떤 것인지 감이 잡히시나요?

이번에 배울 내용은 삼각함수와 원과의 관계에 대한 것입니다.

삼각함수란 이름만 들어보면 원과는 전혀 관계가 없을 것처럼 보이죠?

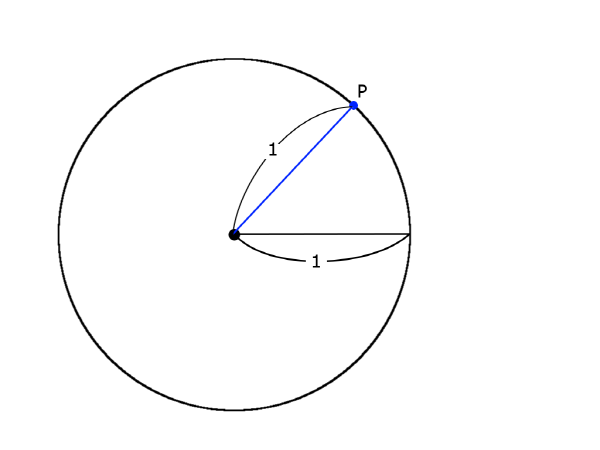
여기서는 원 위의 한 점을 삼각함수를 통해 나타내는 과정을 간단히 설명하도록 하겠습니다.

(삼각함수의 단위원에 대한 정확한 정의는 다른 수학 전문 서적을 참고해 주시기 바랍니다)



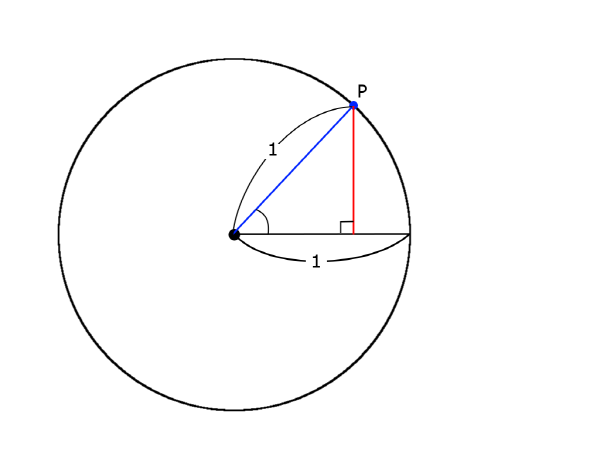
위 그림은 반지름이 1인 원입니다.

1m든 1cm든 단위는 붙이기 나름이니 무엇을 사용하던 상관 없습니다.



원을 따라서 아무 곳이나 한 점을 찍어 그곳을 점 P라고 한다면

중심에서 P까지 이어지는 선의 길이도 반지름과 같습니다.

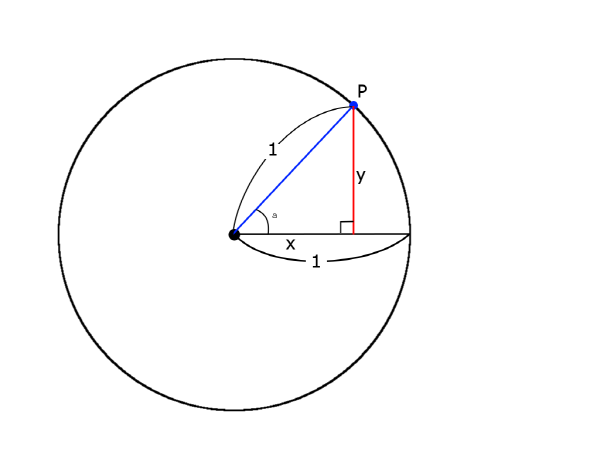


이제 점 P에서 수직으로 선을 내려 바닥과 직각이 되도록 만들면

빗변의 길이가 1인 직각 삼각형 한 개가 생깁니다.

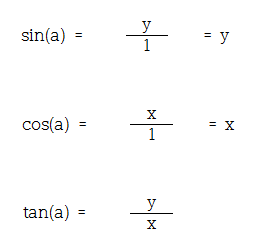
직각 삼각형에 대해서는 삼각함수를 적용할 수 있다고 말씀드렸죠?

원 안에 생긴 직각 삼각형에 대해서 각각 사인, 코사인, 탄젠트를 적용해 봅시다.



직각 삼각형의 밑변을 x, 높이를 y, 각도를 a라고 정하도록 하겠습니다.

먼저 사인부터 적용시켜 보죠.



각각 위와 같이 나타낼 수 있습니다.

따라서 점 P의 좌표 (x,y)는 x = cos(a), y = sin(a) 라고 말할 수 있죠.

더 정확하게 말한다면 아래와 같은 공식이 나옵니다.

x = cos(a) \* r

y = sin(a) \* r

여기서 r은 반지름을 뜻하는데 위 그림에서는 반지름이 1이기 때문에 생략했던 것입니다.

만약 반지름이 1이 아니라면 반지름을 곱해줘야 원하는 위치가 나오겠죠.

이 과정으로 알 수 있는 사실은 삼각함수를 통해서 원의 둘레 위의 한 점의 위치를 나타낼 수 있다는 것입니다.

이것을 응용하면 원 모양으로 움직이는 캐릭터를 구현할 수 있겠죠.

**파이(∏)**

3.141592...

만약 자동차 바퀴의 둘레를 재려고 한다면 어떤 방법을 써야 할까요?

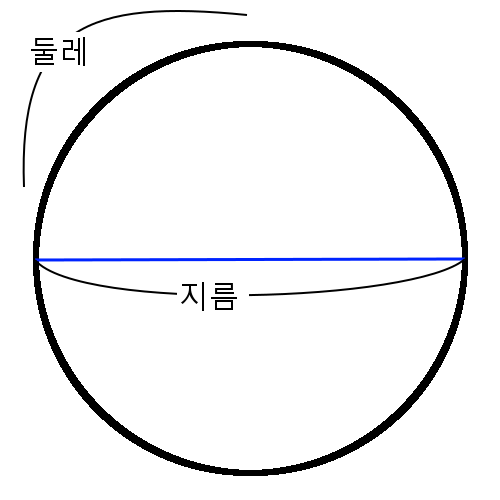
바퀴 둘레에 페인트칠을 해서 한 바퀴 굴린 다음 페인트가 찍힌 길이를 재면 되겠죠.

또는 줄자를 이용하여 바퀴 둘레를 감싸서 잴 수도 있을 겁니다.

또 다른 방법으로는 바퀴 둘레를 잇는 많은 직선들을 만들어 그 직선의 길이를 합친 값을 이용할 수도 있을 것 같습니다.

이 방법들은 원의 둘레를 재기 위해 아주 아주 옛날 사람들이 사용했던 방법들입니다.

이런 방법들로 원의 둘레를 구하다가 원의 지름과 둘레 사이에는 어떠한 일정한 비율이 있다는 것을 발견하게 되었죠.

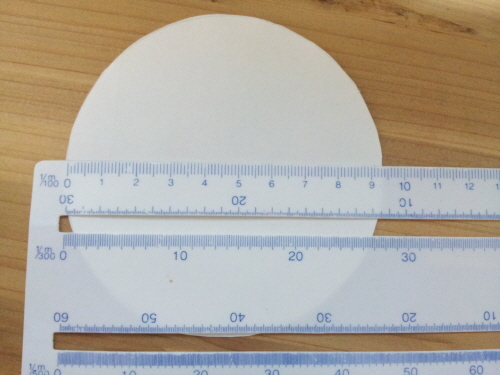


   원의 지름과 둘레에는 일정한 비율이 있지 않을까?

그렇다면 원의 지름과 둘레를 실제로 재 보도록 하겠습니다.



박스 테이프의 둘레 = 29.5cm



박스 테이프의 지름 = 9.4cm

주위에서 흔히 볼 수 있는 박스 테이프를 사용했습니다.

줄자를 돌려 재본 결과 약 29.5cm가 나왔습니다.

지름은 좀 더 정확하게 재기 위하여 테이프의 모양대로 종이에 똑같이 그린 뒤 잘라서

반을 접어 측정하였습니다.

약 9.4cm가 나오는군요.

이제 두 값의 비율을 계산해 보도록 하죠.

비율하면 분수가 딱 떠오르시죠?

클릭시 이미지 새창.

테이프의 둘레는 지름에 비해 약 3.13829정도 크다는 값이 나왔네요.

학창시절 많이 들어본 3.14라는 값과 똑같진 않지만 상당히 비슷한 수치입니다.

(손으로 측정한 것이기 때문에 약간의 오차는 어쩔 수 없군요)

이 비율을 좀 어려운 말로 **원주율** 이라고 부르며 **파이(∏)**라고 표시합니다.

이 원주율은 원의 크기에 상관없이 항상 일정합니다(원은 모두 닮은 도형이므로).

원주율(圓周率)은 원의 지름에 대한 둘레의 비율을 나타내는 수학 상수이다. 수학과 물리학의 여러 분야에 두루 쓰인다. 그리스 문자 π로 표기하고, 파이(π)라고 읽는다.

(출처 : [https://ko.wikipedia.org/wiki/원주율](https://ko.wikipedia.org/wiki/%EC%9B%90%EC%A3%BC%EC%9C%A8))

**라디안**

프로그래밍 언어에서 삼각함수를 사용할 때에는 도 보다는 라디안 이라는 값을 사용합니다.

그렇다면 우리에게 익숙한 각도를 라디안으로 변환하려면 어떻게 해야 하는지 알아야 겠죠.

단순히 변환만을 위한다면 아래처럼 특정한 값을 곱해주기만 하면 됩니다.

도를 라디안으로    90° x 0.01743 =  1.5696 rad(라디안 이라고 읽습니다)

라디안을 도로     2 rad x 57.2957 =  114.59°

하지만 적어도 라디안을 사용한다면 그게 무엇인지는 알아야 하지 않을까요?

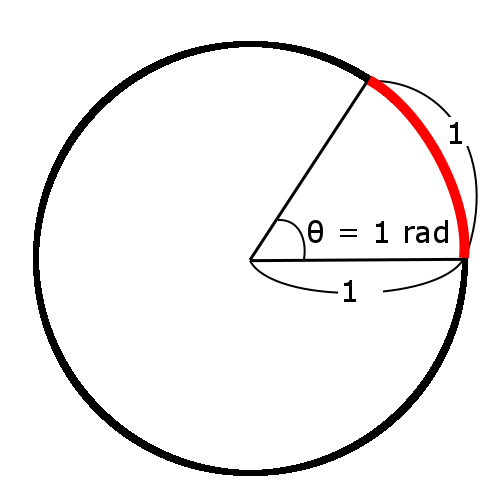
아래는 국어사전에서 검색한 라디안의 뜻입니다.

<수학> 원둘레 위에서 반지름의 길이와 같은 길이를 갖는 호에 대응하는 중심각의 크기. 1라디안은 약 57도 17분 44.8초이다. 기호는 rad. [비슷한 말] 호도(弧度).

(출처 : <http://krdic.naver.com/detail.nhn?docid=11447300>)

역시 한번 듣고 이해할 수 있는 말이 아니죠.

한방에 이해할 수 있도록 그림으로 배워봅시다.

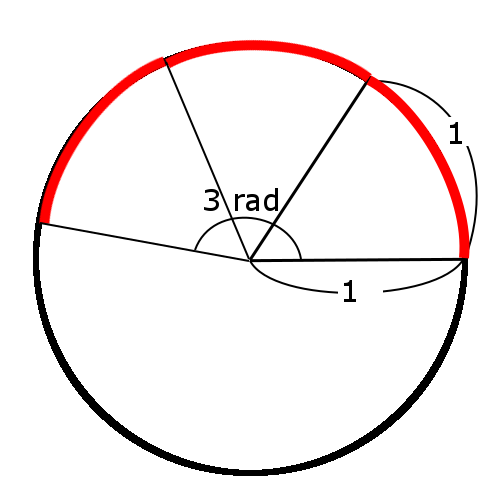


라디안은 바로 이런 모습입니다.

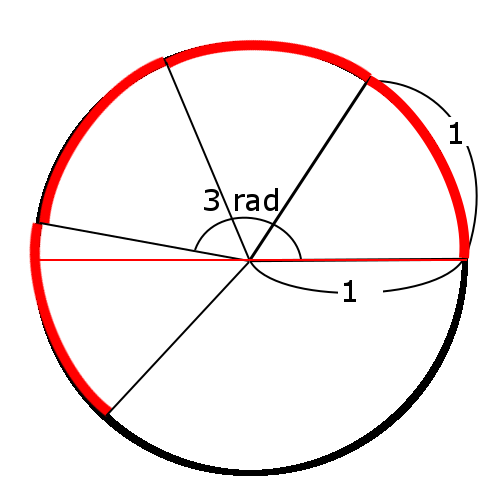
반지름이 1일 때 원을 따라 반지름의 길이 만큼 쭉 올라가서 생긴 각도의 크기를 1 라디안 이라고 부릅니다. 피자의 한 조각처럼 생겼죠.

각도로 치면 대략 57도쯤 됩니다.

그렇다면 2라디안, 3라디안은 얼마 만큼인지 예상할 수 있겠죠?



어려울 것 없이 1라디안이 세 개 모이면 3라디안이 됩니다.



마찬가지로 3라디안에서 1라디안을 더하면 4라디안이 되겠죠.

이제 180도 부분에 빨간색 선을 그어보겠습니다.

위 그림에서 보면 180도 부분은 3라디안과 4라디안 사이의 어딘가가 됩니다.

그 어딘가는 바로 **3.14라디안** 흔히 **파이(∏)**라고 불리우는 값입니다.

이전 파트에서 파이는 원주율(지름과 둘레의 비율)이라고 말씀드렸었죠.

이 말을 다시 하면 **지름이 1일 때** **원의 둘레가 3.14 또는 파이(∏)**라고 할 수 있습니다.

지름이 1이라는 것은 반지름이 0.5라는 뜻이므로 **반지름이 0.5일 때 원의 둘레가 파이(∏)**다 라고 할 수도 있겠네요.

그렇다면 반지름이 1이라면 원의 둘레는 2 x ∏가 됩니다. 줄여서 2∏라고도 씁니다.

(반지름이 0.5에서 1로 두 배가 되었으니 그 둘레도 두 배가 되니까 2를 곱해줘야 겠죠?)

둘레가 2∏라면 그의 반, 즉 180도 만큼의 길이는 ∏가 됩니다.

180도를 라디안으로 표시하면 ∏ 라디안 또는 그냥 ∏ 인 셈이죠.

(라디안은 붙이기도 하고 안 붙이기도 합니다)

정리해보면,

∏(파이)는 지름이 1일 때 원의 둘레를 의미하는 문자이다.

그렇다면 반지름이 1일 때 ∏는 원의 둘레의 절반이다(180도 만큼).

결국 원의 둘레는 반지름이 두 개이므로 2 x ∏ x r (r은 반지름을 뜻한다) 즉 2∏r 이 된다.

원의 둘레를 구하는 공식 2∏r에는 이런 뜻이 있는 것입니다.

또한 ∏는 그냥 ∏ 또는 ∏ 라디안 이라고 표기합니다.

∏값이 3.14이니 3.14 또는 3.14 라디안 이라고 해도 되죠.

뒤에 라디안 이라는 글자는 붙여도 되고 안 붙여도 됩니다.

안 붙어있으면 라디안이라고 이해하면 되고, 각도(°)가 붙어 있으면 도 라고 이해하면 됩니다.

이렇게 각도를 라디안으로 표시하는 방법을 호도법 이라고 부릅니다.

이름에서도 알 수 있듯이 호도법은 호의 길이로 각도를 표시하는 체계인 것입니다.

자, 이제 도를 라디안으로 변환하고 라디안을 도로 변환할 때 사용했던 그 특정한 숫자의 의미를 이해할 수 있는 순간이 왔습니다.

방금 180도가 ∏라디안 이라고 했으니 1도는 몇 라디안일까요?

풀이) 180° => ∏ 라디안 이므로,

1° => ∏ / 180

1° => 3.141592 / 180 = 0.01743

반대로 ∏가 180도 라고 했으니 1라디안은 몇 도 일까요?

풀이) ∏ => 180° 이므로,

3.14 라디안 => 180° 와 같으니,

1 라디안 => 180 / 3.141592 = 57.2957°

∏는 소수점 이하가 끝나지 않고 계속 이어지는 수입니다. 따라서 보통 3.14라는 근사값을 사용합니다.

이번 강좌를 끝으로 3D 게임 프로그래밍을 위한 기초 수학 강좌를 마칩니다.

강좌 대부분이 이론 위주로 되어 있어 게임 프로그래밍을 하는데 당장 써먹을만한 내용은 별로 없었을 겁니다.

하지만 셰이더 프로그래밍 처럼 수학적인 이해가 필요할 때 다시한번 들춰보시면 유용하리라 생각됩니다.

다음번에 또 기회가 되면 실제 코드속에서 응용하는 부분에 대한 내용을 공유해 보도록 하겠습니다.

부족한 글 읽어주셔서 감사합니다.