

Calculadora en R

Calculadora codigo + op

+ **Suma** ejemplo:

```
print(10 + 10)
```

```
## 20
```

- **Resta** Ejemplo:

```
print(10-10)
```

```
## 0
```

** **Multiplicación**** Ejemplo:

```
print(5*5)
```

```
## 25
```

/ **División** Ejemplo:

```
print(10/5)
```

```
## 2.0
```

^ **Potencia** Ejemplo:

```
print(40^2)
```

```
## 42
```

%% **Cociente entero en R pero si es python es //** Ejemplo:

```
print(10//5)
```

```
## 2
```

%% **Resto de la división entera en R pero en python es %** Ejemplo:

```
print(10%5)
```

```
## 0
```

sqrt(x) Raiz cuadrada Ejemplo:

```
sqrt(25)
```

```
## [1] 5
```

exp(x) Exponente de “e” o “Número Euler” que tiene un valor de 2.718282 Ejemplo:

```
exp(1)
```

```
## [1] 2.718282
```

log logartimo y se lee, logaritmo en base x a y Ejemplo:

```
log(8, 2)
```

```
## [1] 3
```

factorial(x) Se define como el número factorial de un número entero positivo(son 0, 1, 2,3,4,5,6,7...) n como $n! = n * (n-1)!$

```
factorial(5)
```

```
## [1] 120
```

choose(n,m) coeficiente binomial se define (n/m) el numero de los subconjuntos de k elemntos que tiene un conjunto de n elemntos ejemplo:

```
choose(5, 3)
```

```
## [1] 10
```

apartir de ahora R solo lo lee en **Radianes** para pasarlo de grados a radianes toca aplicar la regla de 3

180 grados es = a pi radián

por ejemplo: **COS(60PI/180) SIN(60PI/180) asin(0.8660254) * 180/pi**

```
sin(pi/2)
```

```
## [1] 1
```

sin(x) el seno que es Opuesto sobre Hipotenusa

```
sin(2)
```

```
## [1] 0.9092974
```

cos(x) coseno que es adyacente sobre Hipotenusa

```
cos(5)
```

```
## [1] 0.2836622
```

tan(x) tangente que es opuesto sobre adyacente

```
tan(5)
```

```
## [1] -3.380515
```

Luego estan Arcoseno, arcocos y arcotan es la inversa del Seno, Coseno y Tangente

Algoritmo de euclides son segmentos AB y CD se dice que son *conmensurables* si existe un tercer segmento u tal que AB y CD pueden ambos medirse de forma exacta utilizando el segmento u como unidad.

print(x,n) Muestra las ncifras significativa del número x

```
print(2,3)
```

```
## [1] 2
```

round(x,n) redondea a n cifras aparte redondea a la cifra par por ejemplo si tengo 1.35 redondea a 4 y si es 1.25 redondea a 2

```
round(sqrt(2),3)
```

```
## [1] 1.414
```

floor(x) parte entera por defecto de x

```
floor(sqrt(2))
```

```
## [1] 1
```

ceiling(x) parte entera por exceso de x

```
ceiling(sqrt(2))
```

```
## [1] 2
```

trunc(x) parte entera de x, eliminando la parte decimal es como el floor en negativo

```
trunc(sqrt(2))
```

```
## [1] 1
```

$a+bi$ número complejo o número imaginarios

`complex(real=...,imaginary=...)` eso si lo quiero hacer de forma binómica

`complex(modulus=...,argument=...)` eso si lo hago de forma pola?????

`sqrt(as.complex(-x))` una raíz para un número complejo

`Re(x)` Obtiene la parte real de un número complejo

`Im(x)` Obtiene la parte imaginaria de un número complejo

`Mod(x)` obtiene el modulo es el número real positivo que mide su tamaño y generaliza el valor absoluto de un número real.

`Arg(x)` Obtiene el argumento de un número complejo

`Conj(x)` obtiene el conjugado