

## Ejercicio dplyr

Con el dataset diamonds:

```
library(dplyr) # primero cargamos paquete dplyr
```

- Filtrar los diamantes con corte “Ideal”.

```
diamonds_ideal <- filter(diamonds, cut == "Ideal")
```

	carat	cut	color	clarity	depth	table	price	x	y	z
1	0.23	Ideal	E	SI2	61.5	55.0	326	3.95	3.98	2.43
2	0.23	Ideal	J	VS1	62.8	56.0	340	3.93	3.90	2.46
3	0.31	Ideal	J	SI2	62.2	54.0	344	4.35	4.37	2.71
4	0.30	Ideal	I	SI2	62.0	54.0	348	4.31	4.34	2.68
5	0.33	Ideal	I	SI2	61.8	55.0	403	4.49	4.51	2.78
6	0.33	Ideal	I	SI2	61.2	56.0	403	4.49	4.50	2.75
7	0.33	Ideal	J	SI1	61.1	56.0	403	4.49	4.55	2.76
8	0.23	Ideal	G	VS1	61.9	54.0	404	3.93	3.95	2.44
9	0.32	Ideal	I	SI1	60.9	55.0	404	4.45	4.48	2.72
10	0.30	Ideal	I	SI2	61.0	59.0	405	4.30	4.33	2.63
11	0.35	Ideal	I	VS1	60.9	57.0	552	4.54	4.59	2.78
12	0.30	Ideal	D	SI1	62.5	57.0	552	4.29	4.32	2.69
13	0.30	Ideal	D	SI1	62.1	56.0	552	4.30	4.33	2.68
14	0.28	Ideal	G	VVS2	61.4	56.0	553	4.19	4.22	2.58
15	0.32	Ideal	I	VVS1	62.0	55.3	553	4.39	4.42	2.73
16	0.26	Ideal	E	VVS2	62.9	58.0	554	4.02	4.06	2.54

- Seleccionar las columnas carat, cut, color, price y clarity.

```
seleccion <- select(diamonds_ideal,carat, cut, color, price,clarity)
```

	carat	cut	color	price	clarity
1	0.23	Ideal	E	326	SI2
2	0.23	Ideal	J	340	VS1
3	0.31	Ideal	J	344	SI2
4	0.30	Ideal	I	348	SI2
5	0.33	Ideal	I	403	SI2
6	0.33	Ideal	I	403	SI2
7	0.33	Ideal	J	403	SI1
8	0.23	Ideal	G	404	VS1
9	0.32	Ideal	I	404	SI1

- **Crear una nueva columna precio/quilate.**

```
NuevaColumna <- mutate(seleccion, precio_quilate =price/carat)
```

	carat	cut	color	price	clarity	precio_quilate
1	0.23	Ideal	E	326	SI2	1417.391
2	0.23	Ideal	J	340	VS1	1478.261
3	0.31	Ideal	J	344	SI2	1109.677
4	0.30	Ideal	I	348	SI2	1160.000
5	0.33	Ideal	I	403	SI2	1221.212
6	0.33	Ideal	I	403	SI2	1221.212
7	0.33	Ideal	J	403	SI1	1221.212
8	0.23	Ideal	G	404	VS1	1756.522
9	0.32	Ideal	I	404	SI1	1262.500
10	0.30	Ideal	I	405	SI2	1350.000
11	0.35	Ideal	I	552	VS1	1577.143
12	0.30	Ideal	D	552	SI1	1840.000
13	0.30	Ideal	D	552	SI1	1840.000

- **Agrupar los diamantes por color.**

```
by_color <- group_by(diamonds_ideal, color)
```

# Una vez agrupado, podemos ver porejemplo características medias por color

```
mean_by_color <- summarise (by_color, mean(carat),mean(depth),mean(price))
```

	color	mean(carat)	mean(depth)	mean(price)
1	D	0.5657657	61.67837	2629.095
2	E	0.5784012	61.68668	2597.550
3	F	0.6558285	61.67556	3374.939
4	G	0.7007146	61.70016	3720.706
5	H	0.7995249	61.73319	3889.335
6	I	0.9130291	61.79360	4451.970
7	J	1.0635937	61.82199	4918.186

- **Calcular la media del precio/quilate para cada uno de los grupos anteriores.**

```
by_color <- group_by(NuevaColumna, color)
pr_med_quil <- summarise(by_color, mean(precio_quilate))
```

	color	mean(precio_quilate)
1	D	3806.532
2	E	3683.172
3	F	4097.519
4	G	4163.999
5	H	3846.066
6	I	3808.071
7	J	3733.766

- **Ordenar por precio/quilate de forma descendente.**

```
# primero cambio nombre de la columna a PrecioMedioQuilate para poder usar arrange
colnames(pr_med_quil)[2]<- "PrecioMedioQuilate"
```

```
# Ordenamos por PrecioMedioQuilate de forma descendente
ordenado <- arrange(pr_med_quil, desc(PrecioMedioQuilate))
```

	color	PrecioMedioQuilate
1	G	4163.999
2	F	4097.519
3	H	3846.066
4	I	3808.071
5	D	3806.532
6	J	3733.766
7	E	3683.172