Chaining Methods

You have the following functions defined below (read them *carefully!*):

```
fun is-fixed(r): r["fixed"]                 end
fun is-young(r): r["age"] < 4               end
fun nametag(r):  text(r["name"], 20, "red") end
```

The table `t` below represents four animals from the shelter:

name	sex	age	fixed	pounds
"Toggle"	"female"	3	true	48
"Fritz"	"male"	4	true	92
"Nori"	"female"	6	true	35.3
"Maple"	"female"	3	true	51.6

Match each Pyret expression (left) to the description of what it does (right).

`t.order-by("age", true)`	**1** (C)	**A**	Produces a table containing only Toggle and Maple
`t.filter(is-fixed)`	**2** (F)	**B**	Produces a table of only young, fixed animals
`t.build-column("sticker", nametag)`	**3** (D)	**C**	Produces a table, sorted youngest-to-oldest
`t.filter(is-young)`	**4** (A)	**D**	Produces a table with an extra column, named "sticker"
`t.filter(is-young)` `  .filter(is-fixed)`	**5** (B)	**E**	Produces a table containing Maple and Toggle, in that order
`t.filter(is-young)` `  .order-by("pounds", false)`	**6** (E)	**F**	Produces a table containing the same four animals
`t.build-column("label", nametag)` `  .order-by("age", true)`	**7** (H)	**G**	Won't run: will produce an error
`t.order-by("gendr", false)`	**8** (G)	**H**	Produces a table with an extra "label" column, sorted youngest-to-oldest