Data cleaning can be a challenge, so I hope this helps the process for someone out there. This is a tiny, but valuable function for those who deal with data collected from non-ideal forms. As *nearly* always, this depends on the tidyverse library. You may want to rename the function from fml, but it does best describe dealing with mangled data.

This function retuns the first, middle, and last names for a given name or list of names. Missing data is represented as NA.

**Usage on Existing Dataframe**

Setting up a dataframe with manged names and missing first, middle, and last names.

df <- data.frame(names = c("John Jacbon Jingle",

"Heimer Schmitt",

"Cher",

"John Jacbon Jingle Heimer Schmitt",

"Mr. Anderson",

"Sir Patrick Stewart",

"Sammy Davis Jr.")) %>%

add\_column(First = NA) %>%

add\_column(Middle = NA) %>%

add\_column(Last = NA)

| **Row** | **names** | **First** | **Middle** | **Last** |
| --- | --- | --- | --- | --- |
| 1 | John Jacob Jingle | NA | NA | NA |
| 2 | Heimer Schmitt | NA | NA | NA |
| 3 | Cher | NA | NA | NA |
| 4 | John Jacob Jingle Heimer Schmitt | NA | NA | NA |
| 5 | Mr. Anderson | NA | NA | NA |
| 6 | Sir Patrick Stewart | NA | NA | NA |
| 7 | Sammy Davis Jr. | NA | NA | NA |

Replacing the first, middle, and last name values…

df[,c("First","Middle","Last")] <- df$names %>% fml

| **Row** | **names** | **First** | **Middle** | **Last** |
| --- | --- | --- | --- | --- |
| 1 | John Jacbon Jingle | John | Jacbon | Jingle |
| 2 | Heimer Schmitt | Heimer | NA | Schmitt |
| 3 | Cher | Cher | NA | NA |
| 4 | John Jacbon Jingle Heimer Schmitt | John | Jacbon-Jingle-Heimer | Schmitt |
| 5 | Mr. Anderson | NA | NA | Anderson |
| 6 | Sir Patrick Stewart | Patrick | NA | Stewart |
| 7 | Sammy Davis Jr. | Sammy | NA | Davis |

Values Changed

* In roe 1 All names were found
* In row 2 the middle name was skipped
* In row 3 only a first name was found
* In row 4 the middle names were collapsed
* In row 5 only a last name was found
* In row 6 the title Sir was omitted
* In row 7 the title Jr. was omitted

Using with a single name.

fml("Matt Sandy")

|  | **V1** | **V2** | **V3** |
| --- | --- | --- | --- |
| Matt Sandy | Matt | NA | Sandy |

**The Function**

fml <- function(mangled\_names) {

titles <- c("MASTER", "MR", "MISS", "MRS", "MS",

"MX", "JR", "SR", "M", "SIR", "GENTLEMAN",

"SIRE", "MISTRESS", "MADAM", "DAME", "LORD",

"LADY", "ESQ", "EXCELLENCY","EXCELLENCE",

"HER", "HIS", "HONOUR", "THE",

"HONOURABLE", "HONORABLE", "HON", "JUDGE")

mangled\_names %>% sapply(function(name) {

split <- str\_split(name, " ") %>% unlist

original\_length <- length(split)

split <- split[which(!split %>%

toupper %>%

str\_replace\_all('[^A-Z]','')

%in% titles)]

case\_when(

(length(split) < original\_length) &

(length(split) == 1) ~ c(NA,

NA,

split[1]),

length(split) == 1 ~ c(split[1],NA,NA),

length(split) == 2 ~ c(split[1],NA,

split[2]),

length(split) == 3 ~ c(split[1],

split[2],

split[3]),

length(split) > 3 ~ c(split[1],

paste(split[2:(length(split)-1)],

collapse = "-"),

split[length(split)])

)

}) %>% t %>% return

}

**Improvements**

I recommend improving upon this if you want to integrate this function (or attributes of this function) into your workflow. Naming the output or using lists so you can just get partial returns fml("John Smith")$Last could come in handy.

Additional cases could also be created, such as when names are entered Last, First M..