#############################################################################  
###README###  
#############################################################################  
  
#Author information#  
# Elliot Johnson-Hall  
# Orcid: 0009-0003-5105-034X  
# elliot@elliotjh.com  
  
#############################################################################  
###SETUP###  
#############################################################################  
  
#Install and load required libraries  
if (!require(pacman))  
 install.packages("pacman")

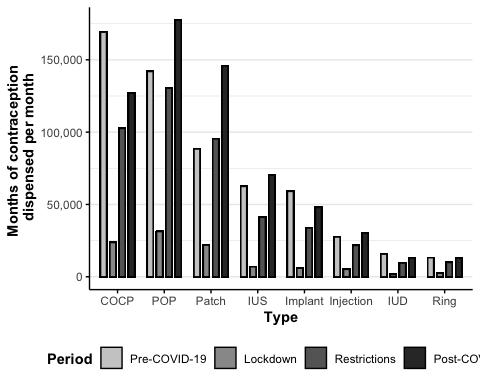
## Loading required package: pacman

pacman::p\_load(  
 "jsonlite",  
 "tidyverse",  
 "glue",  
 "ckanr",  
 "ggplot2",  
 "here",  
 "lubridate",  
 "patchwork",  
 "gt",  
 "vroom",  
 "ckanr"  
)  
  
#ggplot2 Theme  
theme\_base <- function() {  
 theme\_minimal() +  
 theme(  
 panel.grid.major.x = element\_blank(),  
 panel.grid.minor.x = element\_blank(),  
 legend.position = "bottom",  
 plot.tag = element\_text(face = "bold"),  
 axis.title = element\_text(face = "bold"),  
 legend.title = element\_text(face = "bold"),  
 plot.title = element\_text(  
 face = "bold",  
 hjust = 0.5,  
 size = 16  
 ),  
 axis.line = element\_line(  
 colour = "black",  
 linewidth = 0.5,  
 linetype = "solid",  
 lineend = "butt"  
 ),  
 axis.ticks = element\_line(colour = "black"),  
 axis.ticks.length = unit(1, "mm")  
 )  
}  
#############################################################################  
###LOAD DATASET###  
#############################################################################  
  
#Load Data.RDS  
if (file.exists(here("Data", "Data.RDS"))) {  
 CompleteDataset <- read\_rds(here("Data", "Data.RDS"))  
} else {  
 if (exists("CompleteDataset") == FALSE) {  
 CompleteDataset <- NULL  
 } else {  
 CompleteDataset <- CompleteDataset  
 }  
}  
  
###TO ACCESS NHS SCOTLAND OPEN DATA API UNCOMMENT THE FOLLOWING LINES###  
###THIS WILL TAKE A LONG TIME TO LOAD!###  
  
#Connect to NHS Scotland Open Data API and import data on contraception prescriptions dispensed  
# BaseURL <- "https://www.opendata.nhs.scot/"  
#  
# ResourceIDs <-  
# package\_show(id = "84393984-14e9-4b0d-a797-b288db64d088",  
# url = BaseURL,  
# as = "table")$resources %>% select(id)  
#  
# NumberOfRecordsFound <- nrow(ResourceIDs)  
# NumberOfRecordsRetrieved <-  
# length(unique(CompleteDataset$RecordNumber))  
#  
# if (NumberOfRecordsFound > NumberOfRecordsRetrieved) {  
# ChoiceToUpdate <-  
# menu(  
# title = glue(  
# "{NumberOfRecordsFound - NumberOfRecordsRetrieved} new records found. Would you like to update the data?"  
# ),  
# choices = c("Yes", "No")  
# )  
#  
# if (ChoiceToUpdate == 1L) {  
# message("Updating data. This may take some time.")  
# for (Record in 1:NumberOfRecordsFound) {  
# message(glue("Retrieving record {Record} of {NumberOfRecordsFound}."))  
# ResourceID <- ResourceIDs[Record,]  
# SQL <-  
# glue(  
# 'SELECT \* from "{ResourceID}" WHERE ',  
# '"BNFItemCode"',  
# " LIKE '07030%' OR ",  
# '"BNFItemCode"',  
# " LIKE '21040%'"  
# )  
# TempResult <- ds\_search\_sql(SQL, url = BaseURL, as = "json")  
# DataList <- fromJSON(TempResult)  
# assign(glue("DF{Record}"), DataList$result$records)  
# assign(glue("DF{Record}"),  
# get(glue("DF{Record}")) %>% mutate(RecordNumber = glue("{Record}")))  
# message(glue(  
# "Success! {round(  
# (Record / NumberOfRecordsFound \* 100), 2)  
# }% complete."  
# ))  
# }  
# DataFrameList <- mget(ls(pattern = "DF"))  
# if (is.null(CompleteDataset) == FALSE) {  
# DFTemp <- bind\_rows(DataFrameList)  
# CompleteDataset <- left\_join(CompleteDataset, DFTemp)  
# remove(list = ls(pattern = "DF"))  
# } else {  
# CompleteDataset <- bind\_rows(DataFrameList)  
# remove(list = ls(pattern = "DF"))  
# }  
# }  
# else {  
# message("Exited successfully.")  
# }  
# }  
#  
#Write Dataset TO DISK Function  
# WriteOut <- function(Dataframe) {  
# if (dir.exists(here("Data")) == TRUE) {  
# unlink(here("Data"), recursive = TRUE)  
# dir.create(here("Data"))  
# write\_rds(CompleteDataset, file = here("Data", "Data.RDS"))  
# } else {  
# dir.create(here("Data"))  
# write\_rds(CompleteDataset, file = here("Data", "Data.RDS"))  
# }  
# }  
#  
# #Write out Data.RDS  
# CompleteDataset <- CompleteDataset %>%  
# select(c(NumberOfPaidItems:PaidDateMonth, RecordNumber))  
# WriteOut(CompleteDataset)  
  
#############################################################################  
###TABLE 1 - BNF ITEM CODE EXAMPLES###  
#############################################################################  
  
#Table 1 - BNF Item Codes  
Table1Data <-  
 tibble(  
 "Truncated BNF Item Code" = c(  
 "0703021\*",  
 "0703010\*",  
 "0703022M\*",  
 "0703022N\*",  
 "0703023\*",  
 "21040\*",  
 "0703022P\*",  
 "0703050\*",  
 "0703010E0BG\*",  
 "0703011\*"  
 ),  
 "Category" = c(  
 "POP",  
 "COCP",  
 "Injection",  
 "Injection",  
 "IUS",  
 "IUD",  
 "Implant",  
 "EC",  
 "Patch",  
 "Ring"  
 ),  
 "Example BNF Item Description" = c(  
 "Desogestrel Tablet 75mcg",  
 "Rigevidon Tablet",  
 "Depo-Provera Injection 150mg/ml 1ml Pre-filled Syringes",  
 "Noristerat Injection 200mg/ml 1ml Ampoules",  
 "Mirena Intra-uterine System",  
 "T-Safe 380A QL Intra-uterine Contraceptive Device",  
 "Nexplanon Implant 68mg",  
 "Upostelle Tablet 1500mcg",  
 "Evra Transdermal Patch",  
 "NuvaRing 0.12mg/0.015mg per day Vaginal Delivery System"  
 )  
 )  
Table1 <- gt(Table1Data) %>%  
 tab\_header(  
 title = md(  
 "\*\*Table 1\*\* Truncated BNF item codes used during data extraction and example medicines in these categories."  
 )  
 )  
Table1

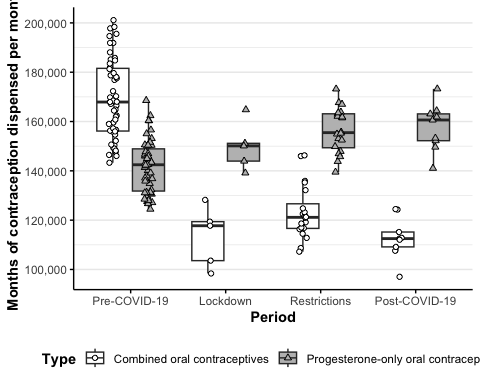
Table 1: **Table 1** Truncated BNF item codes used during data extraction and example medicines in these categories.

| Truncated BNF Item Code | Category | Example BNF Item Description |
| --- | --- | --- |
| 0703021\* | POP | Desogestrel Tablet 75mcg |
| 0703010\* | COCP | Rigevidon Tablet |
| 0703022M\* | Injection | Depo-Provera Injection 150mg/ml 1ml Pre-filled Syringes |
| 0703022N\* | Injection | Noristerat Injection 200mg/ml 1ml Ampoules |
| 0703023\* | IUS | Mirena Intra-uterine System |
| 21040\* | IUD | T-Safe 380A QL Intra-uterine Contraceptive Device |
| 0703022P\* | Implant | Nexplanon Implant 68mg |
| 0703050\* | EC | Upostelle Tablet 1500mcg |
| 0703010E0BG\* | Patch | Evra Transdermal Patch |
| 0703011\* | Ring | NuvaRing 0.12mg/0.015mg per day Vaginal Delivery System |

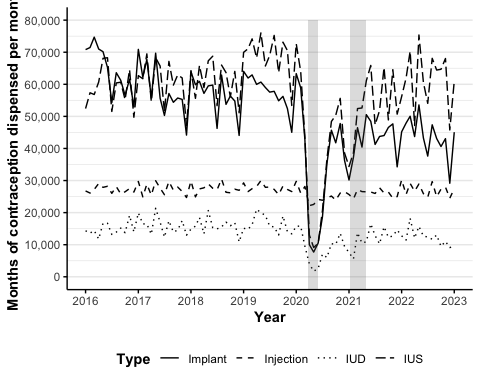
#############################################################################  
###DATA WRANGLING###  
#############################################################################  
# Periods  
# Months between dates  
# Jan 2016 to Apr 2020  
Period1 <- 51  
# Apr 2020 to Apr 2022  
Period2 <- 24  
# May 2022 to Jan 2023  
Period3 <- 8  
TotalMonths <- sum(Period1 + Period2 + Period3)  
  
#Wrangle CompleteDataset  
CompleteDataset <- CompleteDataset %>%  
 filter(between(ym(PaidDateMonth), ym("201601"), ym("202301"))) %>%  
 mutate(  
 PaidDateMonth = ym(PaidDateMonth),  
 PaidQuantity = as.integer(PaidQuantity),  
 # 1 = Pre-COVID-19; 2 = During COVID-19; 3 = Post-COVID-19  
 Period = as.factor(  
 case\_when(  
 PaidDateMonth < dmy("01/04/2020") ~ "1",  
 between(PaidDateMonth, dmy("01/04/2020"), dmy("01/04/2022")) ~ "2",  
 PaidDateMonth > dmy("01/04/2022") ~ "3"  
 )  
 ),  
 TotalMonths = case\_when(Period == 1 ~ 31119114,  
 Period == 2 ~ 12500003,  
 Period == 3 ~ 5520534),  
 Months = case\_when(Period == 1 ~ Period1,  
 Period == 2 ~ Period2,  
 Period == 3 ~ Period3),  
 Type = as.factor(  
 case\_when(  
 str\_detect(BNFItemCode, "0703021") ~ "POP",  
 str\_detect(BNFItemCode, "0703010") &  
 str\_detect(BNFItemDescription, "TAB") ~ "COCP",  
 str\_detect(BNFItemCode, "0703022M|0703022N") ~ "Injection",  
 str\_detect(BNFItemCode, "0703023") ~ "IUS",  
 str\_detect(BNFItemCode, "21040") ~ "IUD",  
 str\_detect(BNFItemCode, "0703022P") ~ "Implant",  
 str\_detect(BNFItemCode, "0703050") ~ "EC",  
 str\_detect(BNFItemCode, "0703010E0BG") ~ "Patch",  
 str\_detect(BNFItemCode, "0703011") ~ "Ring",  
 str\_detect(BNFItemCode, "0703030") ~ "Jelly"  
 )  
 ),  
 ProtectionLength = case\_when(  
 str\_detect(BNFItemCode, "0703021") ~ 1,  
 str\_detect(BNFItemCode, "0703010") &  
 str\_detect(BNFItemDescription, "TAB") ~ 1,  
 str\_detect(BNFItemCode, "0703022M|0703022N") ~ 3,  
 str\_detect(BNFItemCode, "0703023") ~ 60,  
 str\_detect(BNFItemCode, "21040") ~ 60,  
 str\_detect(BNFItemCode, "0703022P") ~ 36,  
 str\_detect(BNFItemCode, "0703010E0BG") ~ 21,  
 str\_detect(BNFItemCode, "0703011") ~ 21  
 ),  
 PackSize = case\_when(  
 str\_detect(BNFItemCode, "0703021") ~ 21,  
 str\_detect(BNFItemCode, "0703010") &  
 str\_detect(BNFItemDescription, "TAB") ~ 21,  
 str\_detect(BNFItemCode, "0703022M|0703022N") ~ 1,  
 str\_detect(BNFItemCode, "0703023") ~ 1,  
 str\_detect(BNFItemCode, "21040") ~ 1,  
 str\_detect(BNFItemCode, "0703022P") ~ 1,  
 str\_detect(BNFItemCode, "0703010E0BG") ~ 3,  
 str\_detect(BNFItemCode, "0703011") ~ 1  
 ),  
 MonthsContraception = (PaidQuantity / PackSize) \* ProtectionLength,  
 Group = case\_when(  
 str\_detect(Type, "POP|COCP") ~ "Oral",  
 str\_detect(Type, "IUS|IUD|Injection|Implant") ~ "LARC",  
 str\_detect(Type, "EC") ~ "EC",  
 str\_detect(Type, "POP|COCP|IUS|IUD|Injection|Implant|EC", negate = T) ~ "Other"  
 ),  
 ECType = case\_when(  
 str\_detect(BNFItemCode, "0703050B") ~ "ULI",  
 str\_detect(BNFItemCode, "0703050A") ~ "LEV"  
 ),  
 Lockdown = as.factor(  
 case\_when(  
 between(PaidDateMonth, dmy("24/03/2020"), dmy("29/05/2020")) ~ "Y",  
 between(PaidDateMonth, dmy("05/01/2021"), dmy("26/04/2021")) ~ "Y",  
 .default = "N"  
 )  
 ),  
 Period2 = factor(  
 case\_when(  
 PaidDateMonth < dmy("01/04/2020") & Lockdown == "N" ~ "1",  
 between(PaidDateMonth, dmy("01/04/2020"), dmy("01/04/2022")) &  
 Lockdown == "N" ~ "2",  
 between(PaidDateMonth, dmy("01/04/2020"), dmy("01/04/2022")) &  
 Lockdown == "Y" ~ "4",  
 PaidDateMonth > dmy("01/04/2022") & Lockdown == "N" ~ "3"  
 )  
 )  
 )  
CompleteDataset$MonthsContraception <-  
 CompleteDataset$MonthsContraception %>% replace\_na(0)  
CompleteDataset <- CompleteDataset %>%  
 filter(!Type == "Jelly")  
CompleteDataset$Period2 <-  
 fct\_relevel(CompleteDataset$Period2, "1", "4", "2", "3")  
  
#############################################################################  
###FIGURES###  
#############################################################################  
  
#Figure1  
Figure1 <- CompleteDataset %>%  
 filter(!Type == "EC") %>%  
 group\_by(Period2, Type) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 ggplot(aes(  
 x = fct\_reorder(Type,-Sum),  
 y = Sum,  
 fill = Period2  
 )) +  
 geom\_bar(  
 width = 0.5,  
 position = position\_dodge(width = 0.8),  
 stat = "identity",  
 colour = "black"  
 ) +  
 theme\_base() +  
 scale\_fill\_manual(  
 values = c("grey80", "grey60", "grey40", "grey20"),  
 labels = c("Pre-COVID-19", "Lockdown", "Restrictions", "Post-COVID-19"),  
 name = "Period"  
 ) +  
 labs(x = "Type", y = "Months of contraception \n dispensed per month") +  
 scale\_y\_continuous(labels = scales::comma)  
Figure1



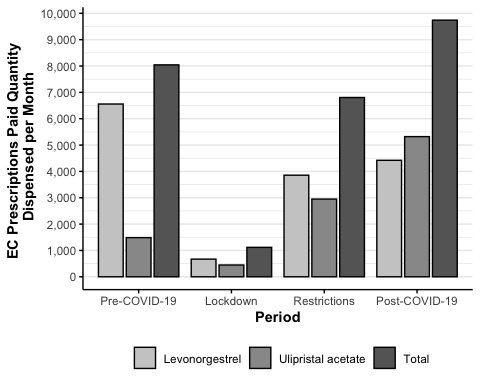
#Figure2  
PLOT\_COCPvsPOP <- CompleteDataset %>%  
 filter(Type %in% c("COCP", "POP")) %>%  
 group\_by(PaidDateMonth, Type, Period2) %>%  
 reframe(Sum = sum(MonthsContraception)) %>%  
 ggplot(aes(y = Sum, x = Period2, fill = Type)) +  
 geom\_boxplot(outlier.shape = NA) +  
 geom\_point(  
 position = position\_jitterdodge(jitter.height = 0, jitter.width = 0.5),  
 aes(group = Type, shape = Type),  
 colour = "black"  
 ) +  
 theme\_base() +  
 labs(y = "Months of contraception dispensed per month",  
 x = "Period") +  
 scale\_y\_continuous(labels = scales::comma,  
 breaks = seq(from = 100000, to = 200000, by = 20000)) +  
 scale\_shape\_manual(  
 values = c(21, 24),  
 name = "Type",  
 labels = c(  
 "Combined oral contraceptives",  
 "Progesterone-only oral contraceptives"  
 )  
 ) +  
 scale\_x\_discrete(labels = c("Pre-COVID-19", "Lockdown", "Restrictions", "Post-COVID-19")) +  
 scale\_fill\_manual(  
 name = "Type",  
 values = c("white", "grey"),  
 labels = c(  
 "Combined oral contraceptives",  
 "Progesterone-only oral contraceptives"  
 )  
 )  
Figure2 <- PLOT\_COCPvsPOP  
Figure2



#Figure3  
LARC\_LinePlot <- CompleteDataset %>%  
 filter(!Type %in% c("COCP", "POP", "Patch", "Ring", "Jelly", "EC")) %>%  
 select(c(PaidDateMonth, Period, Type, MonthsContraception)) %>%  
 group\_by(PaidDateMonth, Type) %>%  
 reframe(Sum = sum(MonthsContraception)) %>%  
 ggplot(aes(y = Sum, x = PaidDateMonth, linetype = Type)) +  
 geom\_line() +  
 theme\_base() +  
 scale\_y\_continuous(labels = scales::comma,  
 breaks = seq(from = 0, to = 800000, by = 10000)) +  
 labs(y = "Months of contraception dispensed per month", x = "Year") +  
 scale\_linetype\_manual(values = c(1, 2, 3, 5)) +  
 scale\_x\_date(date\_breaks = "1 year", date\_labels = "%Y") +  
 theme(panel.grid.minor.x = element\_blank()) +  
 coord\_cartesian(xlim = c(dmy("01/01/2016"), dmy("01/01/2023")),  
 ylim = c(0, 80000),  
 clip = 'off')  
  
LARC\_LinePlot <- LARC\_LinePlot + annotate(  
 "rect",  
 xmin = c(dmy("24/03/2020"), dmy("05/01/2021")),  
 xmax = c(dmy("29/05/2020"), dmy("26/04/2021")),  
 ymin = c(-4000,-4000),  
 ymax = c(80000, 80000),  
 alpha = .2  
)  
Figure3 <- LARC\_LinePlot  
Figure3



#Figure4  
Figure4 <- CompleteDataset %>%  
 filter(Type == "EC") %>%  
 group\_by(Period2, ECType) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique() %>%  
 group\_by(Period2) %>%  
 pivot\_wider(names\_from = ECType, values\_from = Sum) %>%  
 mutate(TOT = sum(LEV + ULI)) %>%  
 pivot\_longer(cols = LEV:TOT,  
 names\_to = "Type",  
 values\_to = "Sum") %>%  
 mutate(Type = factor(Type, levels = c("LEV", "ULI", "TOT"))) %>%  
 ggplot(aes(  
 x = Period2,  
 y = Sum,  
 fill = Type,  
 group = Type  
 )) +  
 geom\_col(  
 position = position\_dodge(width = 0.9),  
 width = 0.8,  
 colour = "black"  
 ) +  
 labs(x = "Period", y = "EC Prescriptions Paid Quantity \n Dispensed per Month", fill = "") +  
 theme\_base() +  
 scale\_fill\_manual(  
 values = c("grey80", "grey60", "grey40"),  
 labels = c("Levonorgestrel", "Ulipristal acetate", "Total")  
 ) +  
 scale\_y\_continuous(labels = scales::comma,  
 breaks = seq(0, 10000, by = 1000)) +  
 scale\_x\_discrete(labels = c("Pre-COVID-19", "Lockdown", "Restrictions", "Post-COVID-19"))  
  
Figure4



#############################################################################  
###SUPPLEMENTARY INFORMATION###  
#############################################################################  
  
CompareTimesTypes <- function(type) {  
 Lockdown <- CompleteDataset %>%  
 filter(Lockdown == "Y",  
 Type == glue("{type}")) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PostLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "3",  
 Type == glue("{type}")) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PreLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "1",  
 Type == glue("{type}")) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PeriLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "2",  
 Type == glue("{type}")) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
  
 return(unlist(  
 c(PreLockdown,  
 Lockdown #/ PreLockdown  
 ,  
 PeriLockdown #/ PreLockdown  
 ,  
 PostLockdown #/ PreLockdown  
 )  
 ))  
}  
  
  
CompareTimesECTypes <- function(ECtype) {  
 Lockdown <- CompleteDataset %>%  
 filter(Lockdown == "Y",  
 ECType == glue("{ECtype}")) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique()  
 PostLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "3",  
 ECType == glue("{ECtype}")) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique()  
 PreLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "1",  
 ECType == glue("{ECtype}")) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique()  
 PeriLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "2",  
 ECType == glue("{ECtype}")) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique()  
  
 return(unlist(  
 c(PreLockdown,  
 Lockdown #/ PreLockdown  
 ,  
 PeriLockdown #/ PreLockdown  
 ,  
 PostLockdown #/ PreLockdown  
 )  
 ))  
}  
  
  
  
CompareTimesEC <- function(group) {  
 Lockdown <- CompleteDataset %>%  
 filter(Lockdown == "Y",  
 Group == glue("{group}")) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique()  
 PostLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "3",  
 Group == glue("{group}")) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique()  
 PreLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "1",  
 Group == glue("{group}")) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique()  
 PeriLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "2",  
 Group == glue("{group}")) %>%  
 reframe(Sum = sum(PaidQuantity) / Months) %>%  
 unique()  
  
 return(unlist(  
 c(  
 PreLockdown,  
 Lockdown #/ PreLockdown  
 ,  
 PeriLockdown #/ PreLockdown  
 ,  
 PostLockdown #/ PreLockdown  
 )  
 ))  
}  
  
  
#Table3 Data  
  
CompareAll <- function(...) {  
 Lockdown <- CompleteDataset %>%  
 filter(Lockdown == "Y") %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PostLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "3") %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PreLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "1") %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PeriLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "2") %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
  
 return(unlist(  
 c(  
 PreLockdown,  
 Lockdown #/ PreLockdown  
 ,  
 PeriLockdown #/ PreLockdown  
 ,  
 PostLockdown #/ PreLockdown  
 )  
 ))  
}  
  
CompareTimes <- function(group) {  
 Lockdown <- CompleteDataset %>%  
 filter(Lockdown == "Y",  
 Group == glue("{group}")) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PostLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "3",  
 Group == glue("{group}")) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PreLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "1",  
 Group == glue("{group}")) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
 PeriLockdown <- CompleteDataset %>%  
 filter(Lockdown == "N",  
 Period == "2",  
 Group == glue("{group}")) %>%  
 reframe(Sum = sum(MonthsContraception) / Months) %>%  
 unique()  
  
 return(unlist(  
 c(  
 PreLockdown,  
 Lockdown #/ PreLockdown  
 ,  
 PeriLockdown #/ PreLockdown  
 ,  
 PostLockdown #/ PreLockdown  
 )  
 ))  
}  
  
  
#Table3  
Table3 <-  
 tibble(  
 "Timeframes" = c("Pre-Lockdown", "Lockdown", "Restrictions", "Post-Lockdown"),  
 "ALL CONTRACEPTION" = CompareAll(),  
 "All LARC" = CompareTimes("LARC"),  
 "IUS" = CompareTimesTypes("IUS"),  
 "IUD" = CompareTimesTypes("IUD"),  
 "Implant" = CompareTimesTypes("Implant"),  
 "Injection" = CompareTimesTypes("Injection"),  
 "All Oral" = CompareTimes("Oral"),  
 "COCP" = CompareTimesTypes("COCP"),  
 "POP" = CompareTimesTypes("POP"),  
 "All Other" = CompareTimes("Other"),  
 "Patch" = CompareTimesTypes("Patch"),  
 "Ring" = CompareTimesTypes("Ring"),  
 "All EC" = CompareTimesEC("EC"),  
 "Uli" = CompareTimesECTypes("ULI"),  
 "Levo" = CompareTimesECTypes("LEV")  
 ) %>%  
 gt() %>%  
 #fmt\_percent(decimals = 2) %>%  
 fmt\_number(decimals = 0, use\_seps = T) %>%   
 tab\_spanner(label = "LARCs", columns = c("All LARC":"Injection")) %>%  
 tab\_spanner(label = "Oral", columns = c("All Oral":"POP")) %>%  
 tab\_spanner(label = "Other", columns = c("All Other":"Ring")) %>%  
 tab\_spanner(label = "EC", columns = c("All EC":"Levo")) %>%  
 tab\_style(locations = cells\_body(columns = c("All LARC", "All Other", "All Oral", "All EC")),  
 style = list(cell\_text(weight = "bold"), cell\_fill(color = "grey80")))  
Table3

|  |  | LARCs | | | | | Oral | | | Other | | | EC | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Timeframes | ALL CONTRACEPTION | All LARC | IUS | IUD | Implant | Injection | All Oral | COCP | POP | All Other | Patch | Ring | All EC | Uli | Levo |
| Pre-Lockdown | 576,986 | **164,543** | 62,432 | 15,641 | 59,004 | 27,466 | **310,952** | 169,096 | 141,855 | **101,491** | 88,340 | 13,151 | **8,043** | 1,486 | 6,557 |
| Lockdown | 98,642 | **19,424** | 6,912 | 1,482 | 5,913 | 5,116 | **54,852** | 23,637 | 31,215 | **24,366** | 21,865 | 2,502 | **1,118** | 447 | 670 |
| Restrictions | 444,055 | **105,964** | 41,102 | 9,455 | 33,730 | 21,676 | **232,923** | 102,547 | 130,376 | **105,168** | 95,311 | 9,857 | **6,804** | 2,949 | 3,855 |
| Post-Lockdown | 624,474 | **161,381** | 70,260 | 13,080 | 47,853 | 30,188 | **304,309** | 126,891 | 177,418 | **158,784** | 145,840 | 12,944 | **9,741** | 5,320 | 4,421 |

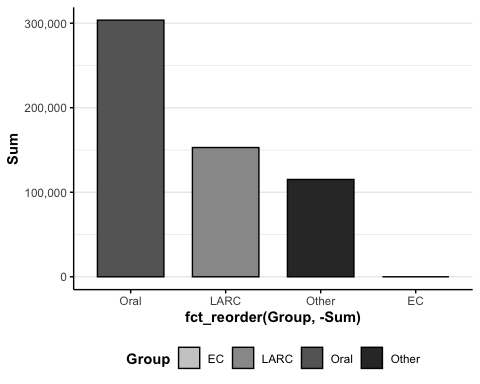
## Oral contraception  
  
CompleteDataset %>%  
 select(c("Group", "MonthsContraception")) %>%  
 group\_by(Group) %>%  
 filter(Group == "Oral") %>%  
 summarise(mean = sum(MonthsContraception) / TotalMonths)

## # A tibble: 1 × 2  
## Group mean  
## <chr> <dbl>  
## 1 Oral 303610.

CompleteDataset %>%  
 select(c("Group", "PaidQuantity")) %>%  
 group\_by(Group) %>%  
 filter(Group == "Oral") %>%  
 summarise(mean = sum(PaidQuantity) / TotalMonths)

## # A tibble: 1 × 2  
## Group mean  
## <chr> <dbl>  
## 1 Oral 6375804.

#Months contraception dispensed by Type of contraception  
CompleteDataset %>%  
 select(c("Group", "MonthsContraception")) %>%  
 group\_by(Group) %>%  
 reframe(Sum = sum(MonthsContraception) / TotalMonths) %>%  
 ggplot(aes(  
 x = fct\_reorder(Group,-Sum),  
 y = Sum,  
 fill = Group  
 )) +  
 geom\_bar(  
 position = position\_dodge(width = 0.9),  
 stat = "identity",  
 width = 0.7,  
 colour = "black"  
 ) +  
 theme\_base() +  
 scale\_fill\_manual(values = c("grey80", "grey60", "grey40", "grey20")) +  
 scale\_y\_continuous(labels = scales::comma)



#Mean contraception dispensed per period  
CompleteDataset %>%  
 filter(Type %in% c("COCP", "POP")) %>%  
 select(c(PaidDateMonth, Period, Type, MonthsContraception)) %>%  
 group\_by(PaidDateMonth, Type) %>%  
 reframe(Sum = sum(MonthsContraception)) %>%  
 mutate(Period = as.factor(  
 case\_when(  
 PaidDateMonth < dmy("01/04/2020") ~ "1",  
 between(PaidDateMonth, dmy("01/04/2020"), dmy("01/04/2022")) ~ "2",  
 PaidDateMonth > dmy("01/04/2022") ~ "3"  
 )  
 )) %>%  
 group\_by(Period, Type) %>%  
 summarise(mean(Sum))

## `summarise()` has grouped output by 'Period'. You can override using the  
## `.groups` argument.

## # A tibble: 6 × 3  
## # Groups: Period [3]  
## Period Type `mean(Sum)`  
## <fct> <fct> <dbl>  
## 1 1 COCP 169096.  
## 2 1 POP 141855.  
## 3 2 COCP 121137.  
## 4 2 POP 155127.  
## 5 3 COCP 112792.  
## 6 3 POP 157705.

CompareType <- "Implant"  
  
During <- CompleteDataset %>%  
 filter(Type == CompareType, Lockdown == "Y") %>%  
 summarise(mean = mean(MonthsContraception))  
  
Pre <- CompleteDataset %>%  
 filter(Type == CompareType, Lockdown == "N", Period == "1") %>%  
 summarise(mean = mean(MonthsContraception))  
  
During / Pre

## mean  
## 1 0.8758107

BNFDesc <- "DEPO"  
  
During <- CompleteDataset %>%  
 filter(str\_detect(BNFItemDescription, BNFDesc), Lockdown == "Y") %>%  
 summarise(mean = mean(MonthsContraception))  
  
Pre <- CompleteDataset %>%  
 filter(str\_detect(BNFItemDescription, BNFDesc),  
 Lockdown == "N",  
 Period == "1") %>%  
 summarise(mean = mean(MonthsContraception))  
  
During / Pre

## mean  
## 1 0.7431125

#############################################################################  
###ENDS###  
#############################################################################