




ŻYWNOSĆ

AUTORZY:

MATYLDA ROGATTY

ALEKSANDRA ZIĘTEK

KLAUDIA TUŃSKA



Celem tego projektu była wizualizacja danych na temat produkcji, spożycia oraz importu i eksportu wybranych artykułów spożywczych w latach 2010-2019.

```
food = food[Y2012> 0 & !is.na(Y2012) & !is.na(Y2010) & Y2010> 0 & !is.na(Y2011)
           & Y2011> 0 & !is.na(Y2013) & Y2013> 0 & !is.na(Y2014) & Y2014> 0
           & !is.na(Y2015) & Y2015> 0 & !is.na(Y2016) & Y2016> 0
           & !is.na(Y2017) & Y2017> 0 & !is.na(Y2018) & Y2018> 0
           & !is.na(Y2019) & Y2019> 0]
```

```
food = food[, list(Area, Item, Element, `2010` = Y2010,
                  `2011` = Y2011, `2012` = Y2012, `2013` = Y2013,
                  `2014` = Y2014, `2015` = Y2015, `2016` = Y2016,
                  `2017` = Y2017, `2018` = Y2018, `2019` = Y2019)]
```

```
food = food[Element %in% c("Production", "Import Quantity", "Export Quantity",
                           "Food", "Tourist consumption")]
```

```
# skrócenie nazw krajów
```

```
# wybrane produkty
```

```
products = c("Cereals - Excluding Beer", "Sugar & Sweeteners", "Pulses",
             "Nuts and products", "Vegetables", "Fruits - Excluding Wine",
             "Wine", "Beer", 'Spices', "Bovine Meat",
             'Mutton & Goat Meat', "Poultry Meat", "Milk - Excluding Butter",
             'Eggs', "Fish, Seafood", "Pigmeat")
```

```
food = food[Item %in% products]
```


```
# ograniczenie krajów z małymi wartościami
food = food[Area %in% unique(food[["Area"]])[1:178]]

food[, s := `2010` + `2011` + `2012` + `2013` + `2014` + `2015` +
           `2016` + `2017` + `2018` + `2019`]
long = melt(food, id.vars = 'Area', measure.vars = 's')
small = unique(long[, su := sum(value),
                  by = c('Area')][, list(Area, su)][order(su)][, Area][1:60])
food = food[!(Area %in% small)]
food = food[!(Area %in% c('China, Hong Kong SAR', 'China, mainland',
                        'China, Taiwan Province of', "Côte d'Ivoire")))]
```



```
2
3   ### FUNKCJE
4   choose_year = function(data_table, year_str){
5     return (data_table[, colnames(data_table) %in%
6       c("Area", "Item", "Element", year_str),
7       with =FALSE])
8   }
9   imp_2018 = choose_year(import, "2018")
10
11   choose_product = function(dt, str_product){
12     return(dt[Item == str_product])
13   }
14   veg_exp = choose_product(export, "Vegetables")
15
16   choose_area = function(dt, str_area){
17     return(dt[Area == str_area])
18   }
19   china_exp = choose_area(export, "China")
20
```

```
00
01 ### WYKRESY
02 item_year_plot = function(DT, year, item){
03   dt = choose_product(choose_year(DT, year), item)
04   g = ggplot(dt, aes(x = reorder(Area, get(year)), y = get(year), fill = Area)) +
05     geom_bar(stat = 'identity') +
06     labs(title = paste(c('Amount of'), item, c('in countries in'), year),
07           y = "Amount (1000 tonnes)", x = "Country") +
08     theme_minimal() +
09     coord_flip() +
10     theme(legend.position = "None", plot.title = element_text(hjust = 0.5))
11   return(g)
12 }
13
```



Aplikacja została podzielona na moduły skupiające się na produkcji, konsumpcji, eksporcie i imporcie. W każdym z modułów umieściliśmy wykres i tabelkę obrazujące wybrane dane.

```
tabPanel("Data",  
  fluidRow(  
    column(width = 6,  
      selectInput(inputId = "region2",  
                   label = "Choose region",  
                   choices = c("-",country),  
                   selected = 1),  
  
      selectInput(inputId = "product2",  
                   label = "Choose product",  
                   choices = c("-",products),  
                   selected = 1),  
  
      submitButton("Update View")),  
    dataTableOutput("summary2")  
  ))),
```



```
5     tabsetPanel(
6         tabPanel("Production",
7             tabsetPanel(
8                 tabPanel("Product/Year Plot",
9
10                    sidebarPanel(
11                        selectInput(inputId = "productP1",
12                                label = "Choose product",
13                                choices = products,
14                                selected = 1),
15
16                        sliderInput(inputId = 'yearP1',
17                                label = 'Choose year',
18                                min = 2010,
19                                max = 2019,
20                                value = c(2010),
21                                sep = ''),
22
23                        submitButton("Update View")),
24
25                    mainPanel(
26                        plotOutput(outputId = "production_plot1",
27                                width = 1100, height = 900))),
28
29         tabPanel("Area/Year Plot"
```

```
2
3   # produkcja
4   output[["production_plot1"]] = renderPlot({
5     item_year_plot(production, toString(input$yearP1), input$productP1)
6   })
7
```