

# Засоби підготовки та аналізу даних

## Лабораторна робота 3

ФБ-24 Воровська Єва

Посилання на репозиторій (GitHub repository):

[https://github.com/yevavorov/ad\\_labs.git](https://github.com/yevavorov/ad_labs.git)

### Хід виконання роботи:

- Зареєструватись на сайті GitHub, створити репозиторій (саме створити, а не “форкнути” результат друга чи подруги), додати в репозиторій код та дані із лабораторної роботи №1, продемонструвати навички роботи з системою контролю версій git на роботі з проектом GitHub;
- Створити веб-додаток із використанням модуля Spyre, який дозволить: обрати часовий ряд VCI, TCI, VHI для набору даних із лабораторної роботи №1 (випадаючий список);
- Вибрати область, для якої буде виконуватись аналіз (випадаючий список);
- Зазначити інтервал тижнів, за які відбираються дані;
- Створити кілька вкладок для відображення таблиці із даними на графіку ходу індексів;
- Інші завдання на прохання викладача. Код розробленого додатку додати до створеного репозиторію.

## Код:

```
class LabaApp(server.App):
    title = "NOAA Data Vizualization"
    inputs = [{
        "type": "dropdown",
        "label": "VCI / TCI / CHI",
        "options": [{"label": "VCI", "value": "VCI"},
                    {"label": "TCI", "value": "TCI"},
                    {"label": "VHI", "value": "VHI"}],
        "key": "data_type",
        "action_id": "update_data"},
        {
        "type": "dropdown",
        "label": "Region",
        "options": [
            {"label": "Вінницька", "value": "1"},
            {"label": "Волинська", "value": "2"},
            {"label": "Дніпропетровська", "value": "3"},
            {"label": "Донецька", "value": "4"},
            {"label": "Житомирська", "value": "5"},
            {"label": "Закарпатська", "value": "6"},
            {"label": "Запорізька", "value": "7"},
            {"label": "Івано-Франківська", "value": "8"},
            {"label": "Київська", "value": "9"},
            {"label": "Кіровоградська", "value": "10"},
            {"label": "Луганська", "value": "11"},
            {"label": "Львівська", "value": "12"},
            {"label": "Миколаївська", "value": "13"},
            {"label": "Одеська", "value": "14"},
            {"label": "Полтавська", "value": "15"},
            {"label": "Рівенська", "value": "16"},
            {"label": "Сумська", "value": "17"},
            {"label": "Тернопільська", "value": "18"},
            {"label": "Харківська", "value": "19"},
            {"label": "Херсонська", "value": "20"},
            {"label": "Хмельницька", "value": "21"},
            {"label": "Черкаська", "value": "22"},
            {"label": "Чернівецька", "value": "23"},
            {"label": "Чернігівська", "value": "24"},
            {"label": "Крим", "value": "25"},
            {"label": "м. Київ", "value": "26"},
            {"label": "Севастополь", "value": "27"}],
        "key": "region",
        "action_id": "update_data"
    },
    {
```

```
"type": "text",
"label": "Week interval",
"key": "week_interval",
"value": "1-10",
"action_id": "update_data"
},
{
"type": "dropdown",
"label": "Year",
"options": [{"label": "1982", "value": "1982"},
{"label": "1983", "value": "1983"},
{"label": "1984", "value": "1984"},
{"label": "1985", "value": "1985"},
{"label": "1986", "value": "1986"},
{"label": "1987", "value": "1987"},
{"label": "1988", "value": "1988"},
{"label": "1989", "value": "1989"},
{"label": "1990", "value": "1990"},
{"label": "1991", "value": "1991"},
{"label": "1992", "value": "1992"},
{"label": "1993", "value": "1993"},
{"label": "1994", "value": "1994"},
{"label": "1995", "value": "1995"},
{"label": "1996", "value": "1996"},
{"label": "1997", "value": "1997"},
{"label": "1998", "value": "1998"},
{"label": "1999", "value": "1999"},
{"label": "2000", "value": "2000"},
{"label": "2001", "value": "2001"},
{"label": "2002", "value": "2002"},
{"label": "2003", "value": "2003"},
{"label": "2004", "value": "2004"},
{"label": "2005", "value": "2005"},
{"label": "2006", "value": "2006"},
{"label": "2007", "value": "2007"},
{"label": "2008", "value": "2008"},
{"label": "2009", "value": "2009"},
{"label": "2010", "value": "2010"},
{"label": "2011", "value": "2011"},
{"label": "2012", "value": "2012"},
{"label": "2013", "value": "2013"},
{"label": "2014", "value": "2014"},
{"label": "2015", "value": "2015"},
{"label": "2016", "value": "2016"},
{"label": "2017", "value": "2017"},
{"label": "2018", "value": "2018"},
{"label": "2019", "value": "2019"},
```

```

        {"label": "2020", "value": "2020"},
        {"label": "2021", "value": "2021"},
        {"label": "2022", "value": "2022"},
        {"label": "2023", "value": "2023"},
        {"label": "2024", "value": "2024"}],
    "key": "year",
    "action_id" : "update_data"
}]

controls = [{"type": "hidden", "id": "update_data"}]

tabs = ["Plot", "Table"]

outputs = [{
    "type": "plot",
    "id": "plot",
    "control_id": "update_data",
    "tab": "Plot"},
    {
    "type": "table",
    "id": "table_id",
    "control_id": "update_data",
    "tab": "Table",
    "on_page_load": True}]

def getData(self, params):
    region = params["region"]
    week_interval = params["week_interval"]
    year = params["year"]
    df = old_code_df()
    df = df[df["Area"] == int(region)]
    week_start, week_end = map(int, week_interval.split("-"))
    df = df[(df["Week"] >= week_start) & (df["Week"] <= week_end) &
(df["Year"] == int(year))]
    return df[["Year", "Week", "SMN", "SMT", "VCI", "TCI", "VHI"]]

def getAreaName(self, region):
    region_map = {
        "1": "Вінницька",
        "2": "Волинська",
        "3": "Дніпропетровська",
        "4": "Донецька",
        "5": "Житомирська",
        "6": "Закарпатська",
        "7": "Запорізька",
        "8": "Івано-Франківська",
        "9": "Київська",

```

```

        "10": "Кіровоградська",
        "11": "Луганська",
        "12": "Львівська",
        "13": "Миколаївська",
        "14": "Одеська",
        "15": "Полтавська",
        "16": "Рівенська",
        "17": "Сумська",
        "18": "Тернопільська",
        "19": "Харківська",
        "20": "Херсонська",
        "21": "Хмельницька",
        "22": "Черкаська",
        "23": "Чернівецька",
        "24": "Чернігівська",
        "25": "Крим",
        "26": "м. Київ",
        "27": "Севастополь"
    }
    return region_map.get(region, "")

def getPlot(self, params):
    df = self.getData(params)
    data_type = params["data_type"]
    year = params["year"]
    region = params["region"]
    week_interval = params["week_interval"]
    y_label = data_type
    x_label = "Weeks"
    region_name = self.getAreaName(region)
    week_start, week_end = map(int, week_interval.split("-"))
    plt_obj = df.plot(x="Week", y=data_type, legend=False)
    plt_obj.set_ylabel(y_label)
    plt_obj.set_xlabel(x_label)
    plt_obj.set_title(f"{data_type} for {region_name}, {int(year)} year,
{week_start}-{week_end} weeks")
    fig = plt_obj.get_figure()
    return fig

if __name__ == "__main__":
    old_code_download()
    app = LabaApp()
    app.launch()

```

*\*old\_code\_download() та old\_code\_df() – код з попередньої лаби.*