# Final Exam

January 16, 2024

Engineering Faculty Academic year: 2023/2024 Assessment Period 1

# 1 Module: Python for Data Analytics (M-227-04)

## 1.1 Final examination

### 1.1.1 Instructions to Students:

- Examination Duration: **2 hours** (120 minutes)
- Maximum amount of points is **100**. Minimum amount of points to pass examination is **50**
- Open book examination: **Yes**
- Internet usage is allowed: **Yes**
- Exam solution form: **Jupiter notebook** (single .ipynb file)

## 1.2 Problem 1 (20 points)

Develop a data structure **ds** for storing information about a sequence of user sessions, where every session has the following attributes: * session ID * session start datetime * ordered sequence of visited pages (e.g., "index", "productlist", "products/product1") * unordered sequence of unique session flags (e.g., "logged in", "order completed", etc.) * user's geo-coordinates (longitude and latitute)

1. Fill the structure with any 3 example sessions.
2. Print the sessions, sorting them by number of visited pages in descending order.

- *Hint: always use appropriate(!) Python data types*

## 1.3 Problem 2 (20 points)

Develop a function **tf(text:str)-> dict** for calculating case-insensitive word frequency in the **text** argument. E.g.,

```
tf("Live is life, Na, na, na, na-na")
```

returns

```
{"life": 0.125,"live": 0.125,"is": 0.125,"na": 0.625}
```

- *Hint 1*: There are 14 punctuation marks that are used in the English language: the period, question mark, exclamation point, comma, colon, semicolon, dash, hyphen, brackets, braces, parentheses, apostrophe, quotation mark, and ellipsis.
- *Hint 2*: Bonus point for a solution with a regular expression

### 1.4 Problem 3 (20 points)

The famous **ordinary least squares** (OLS) formula for linear regression coefficients is:

$$\beta = (X^T X)^{-1} X^T y$$

where - **y** is a vector of the depenedent variable values, - **X** is the matrix of explanatory variables, - $\beta$ are the coefficients.

Your task is to calculate OLS estimates using this formula and **NumPy** library.

1. Load the Iris data set from CSV file available at https://raw.githubusercontent.com/DmitryPavlyuk/python-da/main/week4/iris.csv into a Numpy array. Do not forget to skip the first row - the headers
2. Assign the first column (sepal.length) to the **y** vector and second, third and forth columns (sepal.width, petal.length, petal.width) to the **X** matrix.
3. Calculate OLS coefficients using the formula above

*Hint*: You can validate your solution using the the **statsmodels** library and the foloowing command:

*import statsmodels.api as sm;sm.OLS(y, X).fit().params*

### 1.5 Problem 4 (20 points)

**Target encoding** is the process of replacing a categorical value with the mean of the target variable.

Your task to to encode categorical **variety** variable of the Iris data set using the **pandas** library.

1. Load the Iris data set from CSV file available at https://raw.githubusercontent.com/DmitryPavlyuk/python-da/main/week4/iris.csv into a Pandas data frame.
2. Add a column **mean_sepal_length** that contains **variety**-specific mean **sepal.length** values. Note that the resulting data frame should contain the same number of rows.
3. Print 10 random rows from the resulting data frame.

*Hint*: use pandas grouping and merging functions.

### 1.6 Problem 5 (20 points)

Open-Meteo is an open-source weather API with free access for non-commercial use.

1. Using the Open-Meteo REST API, obtain **hourly forecast of temperature** (*air temperature at 2 meters above ground*) for the next 14 days in Riga. API URL is https://api.open-meteo.com/v1/forecast; all further information is available at https://open-meteo.com/en/docs
2. Plot the obtained values using matplotlib/seaborn.

## 2 Good luck!