

# SUN'IY INTELLEKT ASOSLARI



SIA1046

01

## Python tashqi kutubxonasi. PyPi.org bilan ishlash



SIA

```
def add5(x):  
    return x+5  
  
def dotwrite(nodename, label=sym_label):  
    print '%s.%s' % (nodename, label)  
    if isinstance(label, Node):  
        if ast.literal_eval(label.value):  
            print 'if %s' % label.value  
        else:  
            print 'else:  
                print children = []  
  
for name in children:  
    print '%s' % name,
```



Raxmanov Qurban Sodikovich  
t.f.n., raxmanov@gmail.com

# Reja

1. Data analysisda Pandas kutubxonasi.
2. Series Ma'lumotlar tuzilmasi.
3. DataFrame Ma'lumotlar tuzilmasi.
4. Indekslar. Qator va ustunlarni tashlab yuborish.
5. Elementlarni tanlash.
6. Arifmetik amallar.
7. Funksiyalarni qo'llash.
8. Tartiblash. Reytinglash
9. Dataset statistikasi: min, max, o'rta qiymat, summa va korrelyasiya.
10. Ma'lumotlarni filtrlash.

# Python tashqi kutubxonalar

Tashqi kutubxona (external library) — bu Python dasturlash tiliga dasturchilar tomonidan yaratilgan va til imkoniyatlarini kengaytiruvchi tayyor funksiyalar, sinflar va modullar to‘plamidir.

Python o‘zi bilan birga asosiy modullarni (masalan, math, os, datetime) o‘rnatadi. Ammo murakkab loyihalarda bizga qo‘srimcha imkoniyatlar kerak bo‘ladi

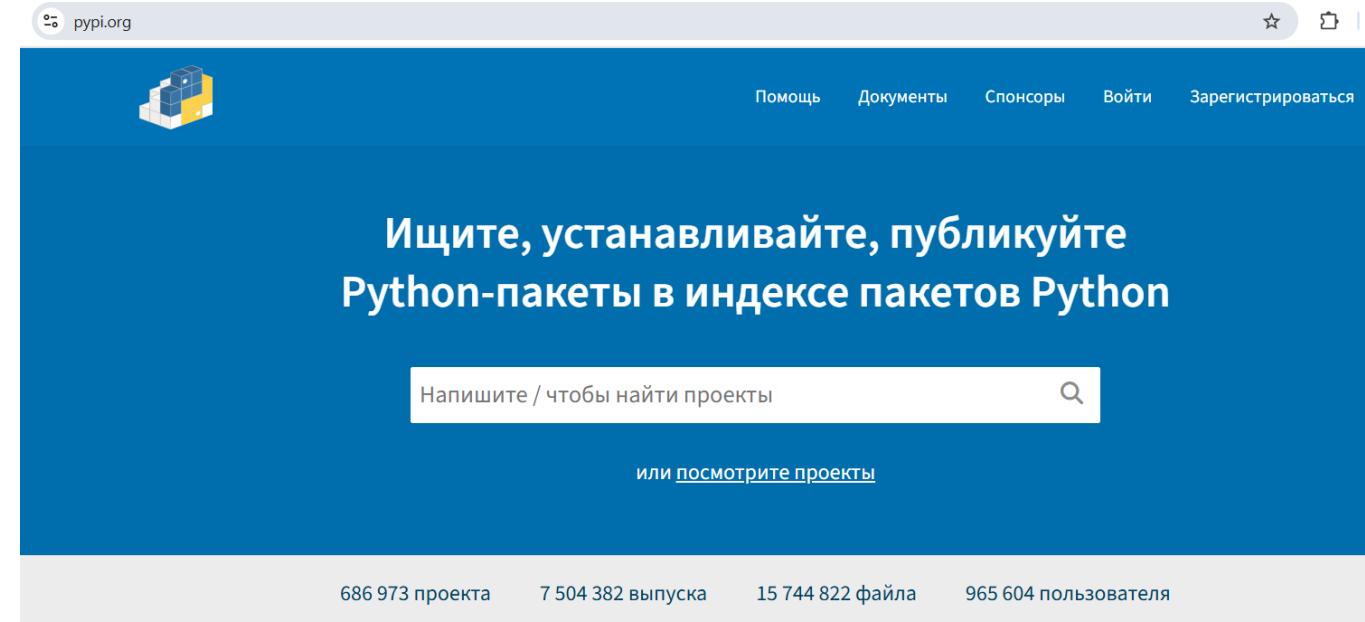
Yo‘nalish	Tashqi kutubxona	Vazifasi
Ma'lumotlar tahlili	NumPy, Pandas	Jadval, massiv va statistik tahlil
Grafik vizualizatsiya	Matplotlib, Seaborn, Plotly	Diagramma, grafiklar
Sun’iy intellekt va ML	TensorFlow, scikit-learn, PyTorch	Mashinaviy o‘rganish, neyron tarmoqlar
Veb dasturlash	Flask, Django, FastAPI	Veb ilovalar yaratish
Foydalanuvchi interfeysi	Tkinter, PyQt5, Kivy	Grafikli dasturlar (GUI)
Tarmoq ishlari	Requests, BeautifulSoup, Scrapy	Veb-so‘rovlар, ma'lumot yig‘ish

# PyPI.org nima?

**PyPI (Python Package Index)** — bu Python kutubxonalarini ombori (repository) bo'lib, butun dunyodagi dasturchilar tomonidan yaratilgan minglab paketlar shu yerda saqlanadi.

## PyPI ning vazifasi:

- Python kutubxonalarini **saqlash**, **tarqatish** va **yangilash**.
- Dasturchilarga o'z kutubxonalarini joylashtirish imkonini berish.
- pip vositasi orqali avtomatik o'rnatish imkoniyatini yaratadi.



# PyPI vs GitHub

**GitHub** - Kodlarni saqlash, versiya boshqaruv

**PyPI** - Kutubxonalarni tarqatish va o'rnatish platformasi

**Paketlar tarkibi.** Dasturchi yangi nashrni PyPI ga yuklaganda, u repozitoriydagi barcha fayllarni emas, balki faqat paketni o'rnatish uchun zarur bo'lgan fayllarni o'z ichiga oladi. Shuning uchun PyPI arxivlari kamroq.

**Paketlarni chiqarish.** Dasturchi GitHub'da paketning yangi versiyasini chiqarganda, u ko'pchilik foydalanuvchilar uchun mo'ljallanmagan bo'lishi mumkin. Nashr PyPIda paydo bo'lmaguncha, u umumiyl foydalanish uchun tayyor emas deb hisoblanishi kerak.

**Foydalanish maqsadlari** PyPI paketlar bilan ishlash uchun, GitHub esa birgalikda ishlab chiqish va kod almashish uchun ko'proq mos keladi.



# Kutubxona o‘rnatish

Terminal yoki PowerShell'da quyidagicha yoziladi:  
pip install numpy

```
Windows PowerShell
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Установите последнюю версию PowerShell для новых функций и улучшения! https://aka.ms/PSWindows

PS C:\Users\user> pip install numpy
Defaulting to user installation because normal site-packages is not writeable
Collecting numpy
  Downloading numpy-2.3.3-cp313-cp313-win_amd64.whl.metadata (60 kB)
  Downloading numpy-2.3.3-cp313-cp313-win_amd64.whl (12.8 MB)
    12.8/12.8 MB 8.2 MB/s eta 0:00:00
Installing collected packages: numpy
|
```

Bu buyruq PyPI.org dan **NumPy** kutubxonasini yuklab olib, tizimga o‘rnatadi.

# Bir necha kutubxonani bir vaqtning o‘zida o‘rnatish

pip install numpy pandas matplotlib

```
Windows PowerShell

Successfully installed pip-25.2
PS C:\Users\user> pip install numpy pandas matplotlib
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: numpy in c:\users\user\appdata\roaming\python\python313\site-packages (2.3.3)
Collecting pandas
  Downloading pandas-2.3.3-cp313-cp313-win_amd64.whl.metadata (19 kB)
Collecting matplotlib
  Downloading matplotlib-3.10.7-cp313-cp313-win_amd64.whl.metadata (11 kB)
Collecting python-dateutil>=2.8.2 (from pandas)
  Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)
Collecting pytz>=2020.1 (from pandas)
  Downloading pytz-2025.2-py2.py3-none-any.whl.metadata (22 kB)
Collecting tzdata>=2022.7 (from pandas)
  Downloading tzdata-2025.2-py2.py3-none-any.whl.metadata (1.4 kB)
Collecting contourpy>=1.0.1 (from matplotlib)
  Downloading contourpy-1.3.3-cp313-cp313-win_amd64.whl.metadata (5.5 kB)
Collecting cycler>=0.10 (from matplotlib)
  Downloading cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
```

# O'rnatilgan barcha kutubxonalarni ko'rish

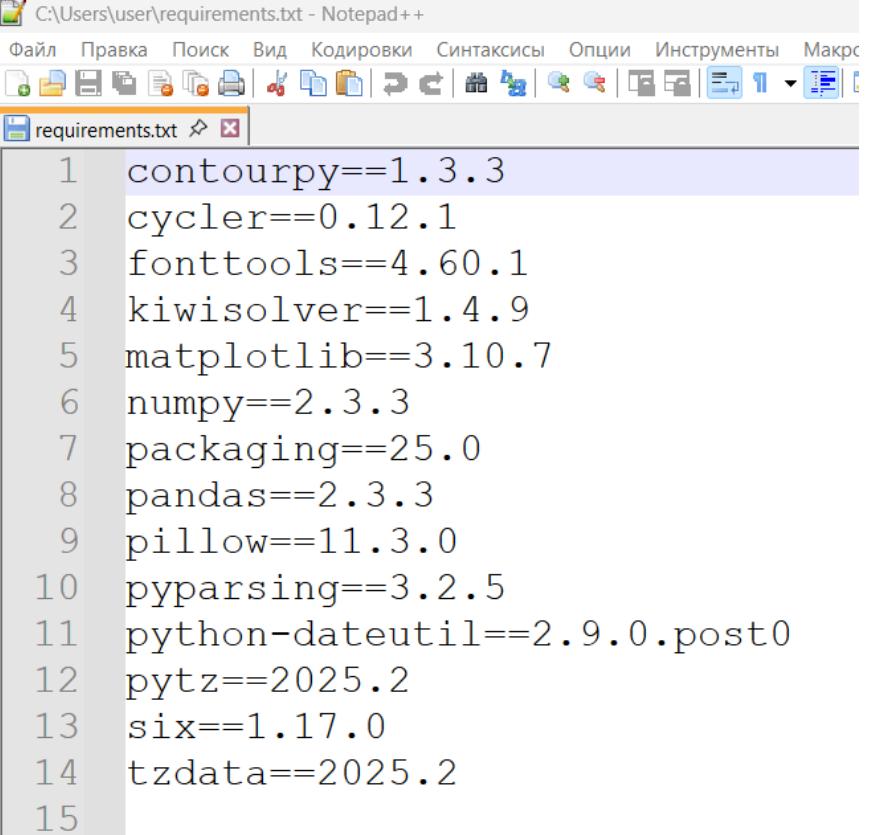
pip list

```
PS C:\Users\user> pip list
Package           Version
-----
contourpy        1.3.3
cycler           0.12.1
fonttools        4.60.1
kiwisolver       1.4.9
matplotlib       3.10.7
numpy            2.3.3
packaging        25.0
pandas           2.3.3
pillow           11.3.0
pip               25.2
pyparsing         3.2.5
python-dateutil  2.9.0.post0
pytz              2025.2
six               1.17.0
tzdata            2025.2
PS C:\Users\user>
```

Kutubxonani o'chirish: pip uninstall numpy

Foydalilanilgan barcha kutubxonalarni avtomatik ko'rish uchun:

pip freeze > requirements.txt



```
C:\Users\user\requirements.txt - Notepad++
Файл Правка Поиск Вид Кодировки Синтаксисы Опции Инструменты Макрос
requirements.txt
1 contourpy==1.3.3
2 cycler==0.12.1
3 fonttools==4.60.1
4 kiwisolver==1.4.9
5 matplotlib==3.10.7
6 numpy==2.3.3
7 packaging==25.0
8 pandas==2.3.3
9 pillow==11.3.0
10 pyparsing==3.2.5
11 python-dateutil==2.9.0.post0
12 pytz==2025.2
13 six==1.17.0
14 tzdata==2025.2
15
```



# PyPI'da kutubxonani izlash va o'r ganish

PyPI saytidan foydalanish:

1.Kirish: <https://pypi.org>

2.Qidiruv oynasiga kutubxona nomini yozing  
(masalan, “numpy”).

3.Kutubxona sahifasida:

- tavsif (description),
- o'rnatish ko'rsatmasi (pip install ...),
- versiyalar,
- manba kod (GitHub havolasi),
- muallif va litsenziya haqida ma'lumot beriladi.

Описание проекта

История выпусков

Загрузка файлов

Проверенные детали ✓  
Эти сведения были проверены PyPI

Owner

NumPy

Сопровождающие

charlesr.harris

matthew.brett

mattip

teoliphant

Непроверенные данные  
Эти сведения не проверены PyPI

Ссылки проекта

documentation

download

homepage

release notes

source

powered by NumFOCUS PyPI downloads 539M/month Conda downloads 112M stackoverflow Ask questions  
DOI 10.1038/s41586-020-2649-2 openssf scorecard 7.2 types typed

NumPy is the fundamental package for scientific computing with Python.

- Website: <https://numpy.org>
- Documentation: <https://numpy.org/doc>
- Mailing list: <https://mail.python.org/mailman/listinfo/numpy-discussion>
- Source code: <https://github.com/numpy/numpy>
- Contributing: <https://numpy.org/devdocs/dev/index.html>
- Bug reports: <https://github.com/numpy/numpy/issues>
- Report a security vulnerability: <https://tidelift.com/docs/security>

It provides:

- a powerful N-dimensional array object
- sophisticated (broadcasting) functions
- tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities

Testing:

NumPy requires `pytest` and `hypothesis`. Tests can then be run after installation with:

```
python -c "import numpy, sys; sys.exit(numpy.test() is False)"
```



# PyPI.org bilan ishlash

Maqsad	Buyruq	Izoh
Kutubxona o'rnatish	pip install package_name	Masalan: pip install requests
Versiyasini tekshirish	pip show package_name	Versiya, joylashuvi va muallif
Yangilash	pip install --upgrade package_name	Eng yangi versiyaga o'tkazish
O'chirish	pip uninstall package_name	
Hamma o'rnatilganlar ro'yxati	pip list	
Kutubxona ma'lumotini saqlash	pip freeze > requirements.txt	Loyihada ishlatilgan paketlarni saqlash

# Sun'iy intellekt uchun kutubxonalar o'rnatish

pip install numpy pandas scikit-learn matplotlib tensorflow

```
8.7/8.7 MB 8.3 MB/s 0:00:01
Downloading tensorflow-2.20.0-cp313-cp313-win_amd64.whl (332.0 MB)
332.0/332.0 MB 6.9 MB/s 0:00:44
Downloading grpcio-1.75.1-cp313-cp313-win_amd64.whl (4.6 MB)
4.6/4.6 MB 7.0 MB/s 0:00:00
Downloading ml_dtypes-0.5.3-cp313-cp313-win_amd64.whl (208 kB)
Downloading requests-2.32.5-py3-none-any.whl (64 kB)
Downloading charset_normalizer-3.4.3-cp313-cp313-win_amd64.whl (107 kB)
Downloading idna-3.10-py3-none-any.whl (70 kB)
Downloading tensorboard-2.20.0-py3-none-any.whl (5.5 MB)
5.5/5.5 MB 7.1 MB/s 0:00:00
Downloading tensorboard_data_server-0.7.2-py3-none-any.whl (2.4 kB)
Downloading typing_extensions-4.15.0-py3-none-any.whl (44 kB)
Downloading urllib3-2.5.0-py3-none-any.whl (129 kB)
Downloading absl_py-2.3.1-py3-none-any.whl (135 kB)
Downloading astunparse-1.6.3-py2.py3-none-any.whl (12 kB)
Downloading wheel-0.45.1-py3-none-any.whl (72 kB)
Downloading certifi-2025.10.5-py3-none-any.whl (163 kB)
Downloading flatbuffers-25.9.23-py2.py3-none-any.whl (30 kB)
Downloading gast-0.6.0-py3-none-any.whl (21 kB)
Downloading google_pasta-0.2.0-py3-none-any.whl (57 kB)
Downloading h5py-3.14.0-cp313-cp313-win_amd64.whl (2.9 MB)
2.9/2.9 MB 6.6 MB/s 0:00:00
Downloading joblib-1.5.2-py3-none-any.whl (308 kB)
Downloading keras-3.11.3-py3-none-any.whl (1.4 MB)
1.4/1.4 MB 6.0 MB/s 0:00:00
Downloading libclang-18.1.1-py2.py3-none-win_amd64.whl (26.4 MB)
26.4/26.4 MB 7.2 MB/s 0:00:03
Downloading markdown-3.9-py3-none-any.whl (107 kB)
Downloading opt_einsum-3.4.0-py3-none-any.whl (71 kB)
Downloading protobuf-6.32.1-cp310-abi3-win_amd64.whl (435 kB)
Downloading scipy-1.16.2-cp313-cp313-win_amd64.whl (38.5 MB)
38.5/38.5 MB 8.7 MB/s 0:00:04
Downloading setuptools-80.9.0-py3-none-any.whl (1.2 MB)
0.0/1.2 MB ? eta -:--:--
```

# Sun’iy intellekt uchun kutubxonalar o‘rnatish

```
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt

# Oddiy model
x = np.array([[1], [2], [3], [4]])
y = np.array([2, 4, 6, 8])
model = LinearRegression()
model.fit(x, y)
print(model.predict([[5]])) # natija: [10.]
```

1. IDLE
2. VS Code
3. Jupyter Notebook
4. Google Colab
5. ...

Shu dasturlarda ishlatib korish mumkin

# Sun'iy intellekt uchun kutubxonalar o'rnatish (VS CODE)

```
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt

# Oddiy model
x = np.array([[1], [2], [3], [4]])
y = np.array([2, 4, 6, 8])
model = LinearRegression()
model.fit(x, y)
print(model.predict([[5]]))
```

The screenshot shows the Microsoft Visual Studio Code (VS Code) interface. On the left, there's a sidebar with various icons for extensions, file operations, and user settings. The main area displays a Python script named `ai_test.py`. The code imports numpy, pandas, and LinearRegression from sklearn, and matplotlib.pyplot. It then defines a simple linear model with `x` and `y` arrays and prints the prediction for `x=5`. Below the code editor, the terminal tab is active, showing the command `$ python ai_test.py` followed by the output `[10.]`. The status bar at the bottom indicates the current user is `Qurban Raxmanov` on a `MINGW64` system.

```
File Edit Selection View ... ← → Python EXTENSIONS ... Welcome x ai_test.py x ai_test.py > ...
1 import numpy as np
2 import pandas as pd
3 from sklearn.linear_model import LinearRegression
4 import matplotlib.pyplot as plt
5
6 # Oddiy model
7 x = np.array([[1], [2], [3], [4]])
8 y = np.array([2, 4, 6, 8])
9 model = LinearRegression()
10 model.fit(x, y)
11 print(model.predict([[5]])) # natija: [10.]
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS bash + ...

Requirement already satisfied: six>=1.5 in c:\users\user\appdata\roaming\python\python313\site-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)

Qurban Raxmanov@Soccerstand MINGW64 /d/0 OXIA/0 2025-2026/0 SIA KI masofa/Python

\$ python ai\_test.py [10.]

# Nima uchun natija 10 chiqdi

**Sun'iy intellekt (aniqrog'i, mashinaviy o'qitish)** - jarayonidagi modelni o'qitish bosqichini ifodalaydi.

```
# Oddiy model
```

```
x = np.array([[1], [2], [3], [4]])
```

```
y = np.array([2, 4, 6, 8])
```

```
model = LinearRegression()
```

```
model.fit(x, y)
```

```
print(model.predict([[5]])) # natija: [10.]
```

**model.fit(x, y)**

Bu yerda **fit()** metodi modelni “o'qitadi”.

Yani:

- **x — kiruvchi ma'lumotlar** (input yoki mustaqil o'zgaruvchi),

- **y — chiqish natijalari** (output yoki bog'liq o'zgaruvchi).

**fit()** metodi shu x va y orasidagi **matematik munosabatni** topadi.

Misol uchun, bu holatda u **y = 2x** ekanini o'rganadi.

**moodel = LinearRegression()**

Bu satr bilan **chiziqli regressiya** modelini yaratiladi.

Bu model **ma'lumotlar orasidagi bog'liqlikni (munosabatni)** o'rganadi.

x	y
1	2
2	4
3	6
4	8

Jadvaldagi **x ortsa, y ham ortadi**.

Bunda **chiziqli bog'liqlik** mavjud.

**model.predict([[5]])**

Endi modeldan foydalanib, **x = 5 bo'lganda y qancha bo'ladi?** degan savolni beramiz.

Model ilgari o'rgangan bog'liqlikka ( $y = 2x$ ) asoslanib, **natija [10.]** ni qaytaradi.

# Uy maydoni asosida narxni aniqlash

```
import numpy as np
from sklearn.linear_model import LinearRegression

# Ma'lumotlar (x - maydon, y - narx)
x = np.array([[50], [60], [80], [100], [120]])    # kv.m
y = np.array([30000, 35000, 45000, 55000, 65000])  # dollar

# Model yaratamiz
model = LinearRegression()

# Modelni o'qitamiz
model.fit(x, y)

# Yangi qiymat uchun bashorat qilamiz
maydon = 90
narx = model.predict([[maydon]])

print(f"{maydon} kv.m uy taxminan {narx[0]:.0f} dollarga tushadi.")
```

# Uy maydoni asosida narxni aniqlash

```
import numpy as np
from sklearn.linear_model import LinearRegression

# Ma'lumotlar (x - maydon, y - narx)
x = np.array([[50], [60], [80], [100], [120]]) # kv.m
y = np.array([30000, 35000, 45000, 55000, 65000]) # dollar

# Model yaratamiz
model = LinearRegression()

# Modelni o'qitamiz
model.fit(x, y)

# Yangi qiymat uchun bashorat
maydon = 90
narx = model.predict([[maydon]])

print(f"{maydon} kv.m uy taxminan {narx[0]:.0f} dollarga tushadi.")
```

The screenshot shows a code editor interface with the file 'ai\_test.py' open. The code implements a linear regression model to predict house prices based on their area (maydon). The 'ai\_test.py' file contains the following code:

```
5  x = np.array([[50], [60], [80], [100], [120]]) # kv.m
6  y = np.array([30000, 35000, 45000, 55000, 65000]) # dollar
7  # Model yaratamiz
8  model = LinearRegression()
9  # Modelni o'qitamiz
10 model.fit(x, y)
11 # Yangi qiymat uchun bashorat qilamiz
12 maydon = 90
13 narx = model.predict([[maydon]])
14 print(f"{maydon} kv.m uy taxminan {narx[0]:.0f} dollarga tushadi.)
```

Below the code editor is a terminal window showing the execution of the script and its output:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS bash + ×
Qurban Raxmanov@Soccerstand MINGW64 /d/0 OXIA/0 2025-2026/0 SIA KI masofa/Python
$ python ai_test.py
Qurban Raxmanov@Soccerstand MINGW64 /d/0 OXIA/0 2025-2026/0 SIA KI masofa/Python
$ python ai_test.py
90 kv.m uy taxminan 50000 dollarga tushadi.
```

# SUN'IY INTELLEKT ASOSLARI



```
def add5(x):  
    return x+5  
  
def dictwrite(ast):  
    nodename = getNodeName()  
    label=symbol.sym_name.get(int(ast[0]),ast[0])  
    print '    %s [label="%s" % (nodename, label),  
  
else:  
    print '"];'  
    children = []  
    for n, child  
        children.append(child)  
    print '    %s -> %s;'.format(nodename,  
    for name in children:  
        print '%s' % name,
```

E'tiboringiz uchun rahmat!

Raxmanov Qurbon Sodikovich  
t.f.n., raxmanov@gmail.com