**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**
**Федеральное государственное автономное образовательное учреждение высшего образования «Северо-Кавказский федеральный университет»**

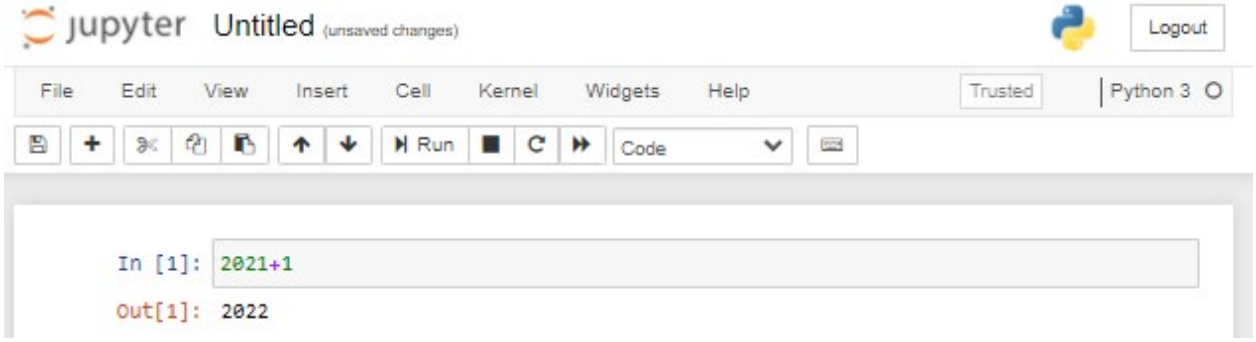
**Отчет по лабораторной работе №**1
***«Работа с IPython и Jupyter Notebook»***

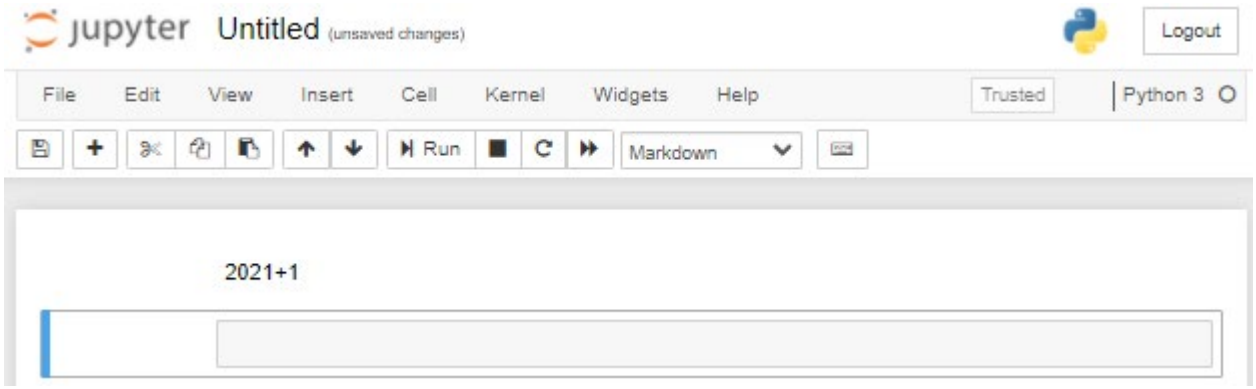
**по дисциплине «Технологии распознавания образов»**

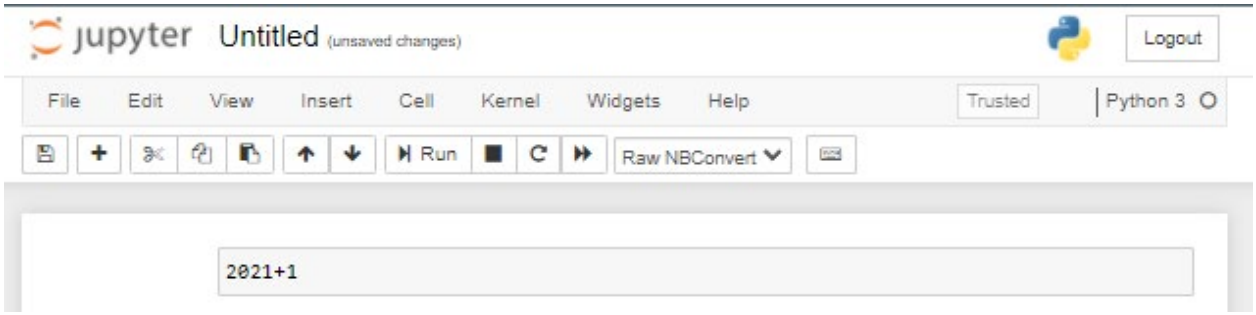
Выполнил студент группы ПИЖ-б-о-20-1
Симоненко А.С. «  » _____ 2022г.
Подпись студента _____
Работа защищена «  »_____2022г.
Проверил Воронкин Р.А. _____
(подпись)

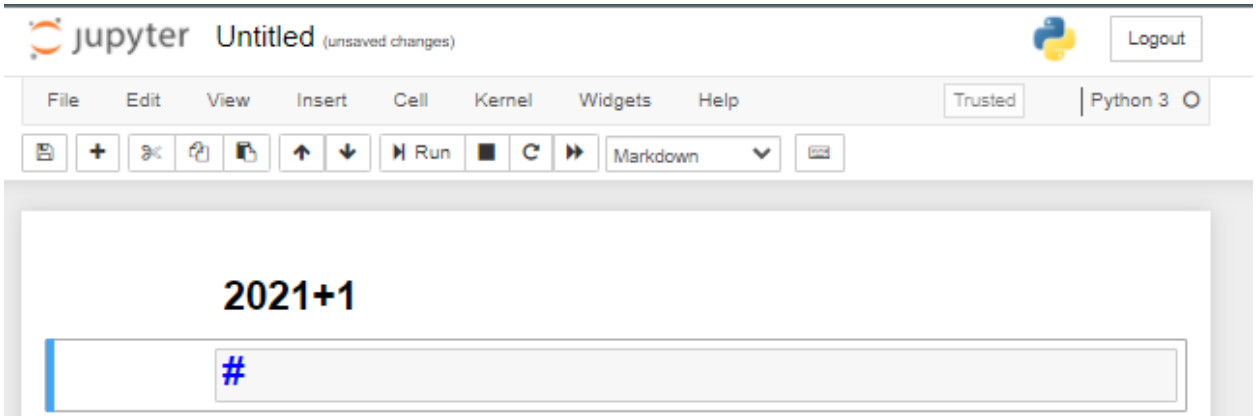
Ставрополь, 2022 г.

## Примеры


jupyter Untitled (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [1]: 2021+1
Out[1]: 2022


jupyter Untitled (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
2021+1


jupyter Untitled (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Raw NBConvert
2021+1


jupyter Untitled (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Markdown
2021+1
#

jupyter Untitled (unsaved changes)

Logout

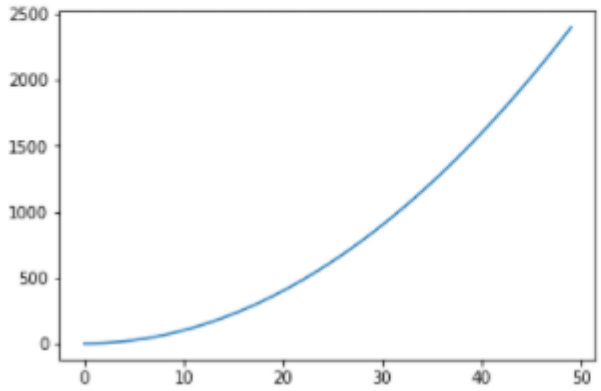
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

In [4]:
```
from matplotlib import pylab as plt
%matplotlib inline
```

In [5]:
```
x = [i for i in range(50)]
y = [i**2 for i in range(50)]
plt.plot(x, y)
```

Out[5]: [<matplotlib.lines.Line2D at 0xffff63216880>]




**Решить задания**

## 1) Счастливый билетик

```
def ticket_number(ticket):
    ticket = str(ticket)
    IS = int(ticket[0]) + int(ticket[1]) + int(ticket[2]) == int(ticket[3]) + int(ticket[4]) + int(tic
    if IS:
        print("Yes")
    else:
        print("No")

ticket_number(123456)
```

No

```
def ticket_number(ticket):
    ticket = str(ticket)
    IS = int(ticket[0]) + int(ticket[1]) + int(ticket[2]) == int(ticket[3]) + int(ticket[4]) + int(tic
    if IS:
        print("Yes")
    else:
        print("No")

ticket_number(123042)
```

Yes

## 2) Пароль

```
import string

def is_password(pswd) -> bool:
    if len(pswd) < 4:
        return False

    lm, tm, zm, sm = 0, -1, 0, 0

    for pswd_ch in pswd:
        if not(lm <= 2):
            return False

        if pswd_ch in string.ascii_lowercase:
            zm = 0

        elif pswd_ch in string.ascii_uppercase:
            zm = 1

        elif pswd_ch in (string.digits +  string.punctuation + " "):
            zm = 2
        else:
            print("Error: Only Latin letters, punctuation marks and numbers translation:\"" +  pswd_ch + "\"")
            return False

        if not(tm == zm):
            tm = zm
            lm = 0

        lm += 1

        if not ((sm & ~(2**zm)) == (2**zm)):
            sm = sm | (2**zm)

    if sm == 7:
        return True
    return False

if is_password("Aandrei123"):
    print("strong")
else:
    print("weak")
```

```
weak
```

```
import string

def is_password(pswd) -> bool:
    if len(pswd) < 4:
        return False

    lm, tm, zm, sm = 0, -1, 0, 0

    for pswd_ch in pswd:
        if not(lm <= 2):
            return False

        if pswd_ch in string.ascii_lowercase:
            zm = 0

        elif pswd_ch in string.ascii_uppercase:
            zm = 1

        elif pswd_ch in (string.digits +  string.punctuation + " "):
            zm = 2
        else:
            print("Error: Only Latin letters, punctuation marks and numbers translation:\"" +  pswd_ch + "\"")
            return False

        if not(tm == zm):
            tm = zm
            lm = 0

        lm += 1

        if not ((sm & ~(2**zm)) == (2**zm)):
            sm = sm | (2**zm)

    if sm == 7:
        return True
    return False

if is_password("an12dRei"):
    print("strong")
else:
    print("weak")
```

```
strong
```

## 3) Числа Фибоначчи

```
def fab(n):
    n = int(n)
    fab1 = fab2 = 1
    if(n > 0):
        print(fab1, end=" ")

    if(n > 1):
        print(fab2, end=" ")

    for i in range(2, n):
        fab1, fab2 = fab2, fab1 + fab2
        print(fab2, end=" ")

fab(3)
```

```
1 1 2
```

```python
def fab(n):
    n = int(n)
    fab1 = fab2 = 1
    if(n > 0):
        print(fab1, end=" ")

    if(n > 1):
        print(fab2, end=" ")

    for i in range(2, n):
        fab1, fab2 = fab2, fab1 + fab2
        print(fab2, end=" ")

fab(10)
```

```
1 1 2 3 5 8 13 21 34 55
```

## 4) Время исследований

Пусть таблица `history_weather_munich.csv` содержит данные по температура за все года:

- Termin - **дата**
- Max. Temp. - **Максимальная температура**
- Min. Temp. - **Минимальная Температура**

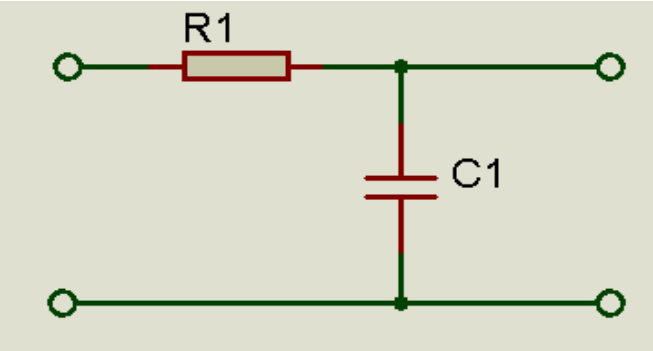
Одно из направлений исследования могло бы заключаться в проверке зависимости суммарного числа температуры.

In [43]:
```python
import os.path
import math
import csv

def mean(list_numb : list):
    return sum(list_numb) / len(list_numb)

def variance(list_numb : list):
    return sum([(x - mean(list_numb)) ** 2 for x in list_numb]) / (len(list_numb) - 1)

def stdev(list_numb : list):
    return math.sqrt(variance(list_numb))

def mnk(a : list, b : list):
    if len(a) != len(b):
        print("Error: a != b;")
        return

    ab = [ x * b[k]  for k, x in enumerate(a)]
    a2 = [ x ** 2  for x in a]
    am = sum(a) / len(a)
    bm = sum(b) / len(a)
    br = (len(a) * sum(ab) - sum(a) * sum(b)) / (len(a) * sum(a2) - (sum(a) ** 2))
    ar = bm - br * am
    return ar

def corrcoef(a : list, b : list):
    if len(a) != len(b):
        print("Error: a != b;")
        return

    an = [ x - (sum(a)/len(a)) for x in a ]
    bn = [ x - (sum(b)/len(a)) for x in b ]
    abp = sum([ x ** 2 for x in an ]) * sum([ x ** 2 for x in bn ])
    abq = math.sqrt(abp)

    return sum([ x * bn[k]  for k, x in enumerate(an)]) / abq

array_temp = { "date": [], "min": [], "max":[] }

with open('./history_weather_munich.csv', newline='') as csvfile:
        spamreader = csv.DictReader(csvfile, delimiter=";")
        for row in spamreader:
            array_temp["date"].append(row["Termin"])
            array_temp["min"].append(float(row["Min. Temp."].replace(",", ".")))
            array_temp["max"].append(float(row["Max. Temp."].replace(",", ".")))

print("Мин. темп.: ", "min:{", min(array_temp["min"]), "};    max:{", max(array_temp["min"]), "};" )
print("Мах. темп.: ", "min:{", min(array_temp["max"]), "};    max:{", max(array_temp["max"]), "};" )
print("Мин. темп. сред.: ", "%.2f" % mean(array_temp["min"]))
print("Мах. темп. сред.: ", "%.2f" % mean(array_temp["max"]))
print("МНК", "%.2f" % mnk(array_temp["min"], array_temp["max"]))
print("Коэффициент парной корреляции", "%.2f" % corrcoef(array_temp["min"], array_temp["max"]))
```

```
Мин. темп.:  min:{ -16.6 };    max:{ 22.1 };
Мах. темп.:  min:{ -10.7 };    max:{ 37.0 };
Мин. темп. сред.:  6.08
Мах. темп. сред.:  14.38
МНК 7.17
Коэффициент парной корреляции 0.91
```

## Самостоятельно задание

## Фильтр нижних частот




In [2]:
```
import numpy as np
import matplotlib.pyplot as plt

def Uin(t):
    res = -1
    if (int)(2*t)%2 == 0:
        res = 1
    return res

N = 101
time = np.linspace(0,10,N)
U = np.empty(N, float)
for i in range(N):
    U[i]=Uin(time[i])

plt.plot(time,U)
plt.xlabel('t')
plt.ylabel('U_in')
plt.show()
```