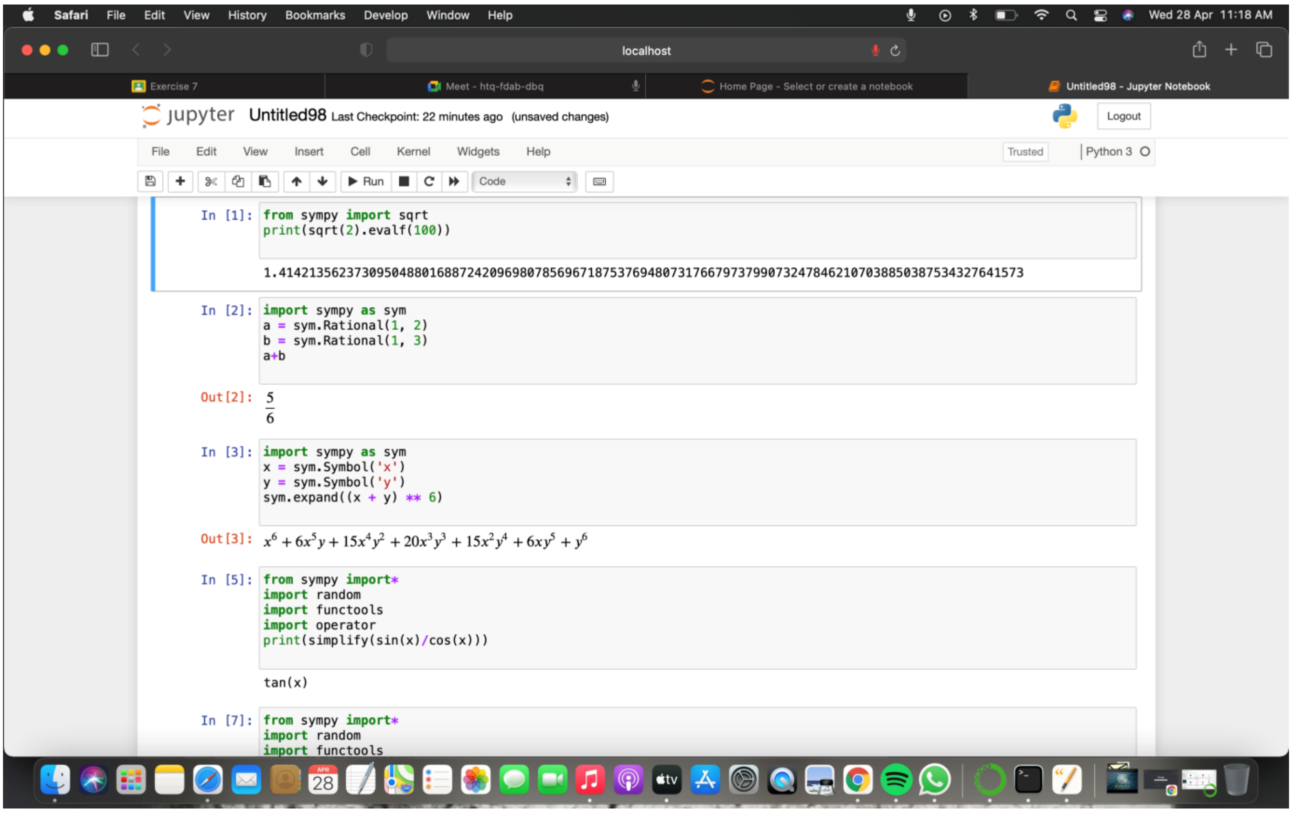
APP_Exercise_7

Name: Rahul Goel

Reg No: RA1911030010094

1. Solve the following using symbolic paradigm:

i. Calculate sqrt (2) with 100 decimal
ii. Calculate (1/2+1/3) in rational arithmetic.
iii. Calculate the expanded form of (x+y) ^ 6.
iv. Simplify the trigonometric expression sin (x) / cos (x) v. Calculate sin x -xx^3n


Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 11:18 AM
localhost
Exercise 7
Meet - htq-fdab-dbq
Home Page - Select or create a notebook
Untitled98 - Jupyter Notebook
jupyter Untitled98 Last Checkpoint: 22 minutes ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

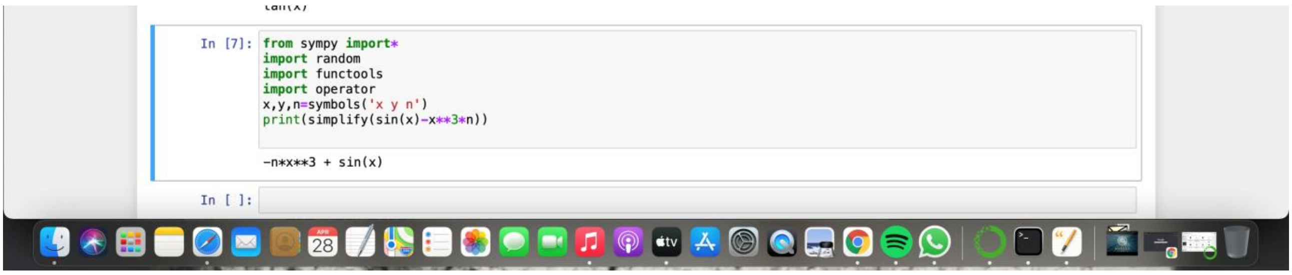
```
In [1]: from sympy import sqrt
        print(sqrt(2).evalf(100))

        1.414213562373095048801688724209698078569671875376948073176679737990732478462107038850387534327641573

In [2]: import sympy as sym
        a = sym.Rational(1, 2)
        b = sym.Rational(1, 3)
        a+b
```

Out[2]: $\frac{5}{6}$

```
In [3]: import sympy as sym
        x = sym.Symbol('x')
        y = sym.Symbol('y')
        sym.expand((x + y) ** 6)
```

Out[3]: $x^6 + 6x^5y + 15x^4y^2 + 20x^3y^3 + 15x^2y^4 + 6xy^5 + y^6$

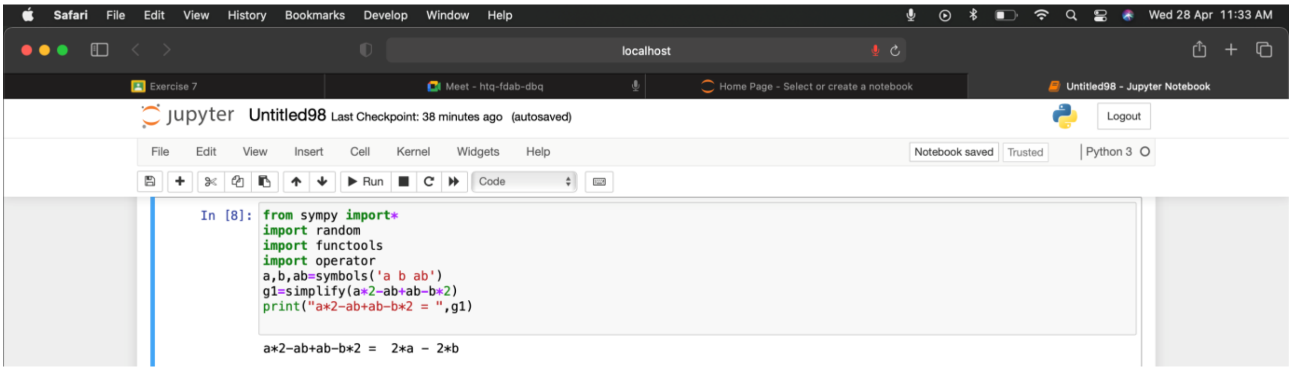
```
In [5]: from sympy import*
        import random
        import functools
        import operator
        print(simplify(sin(x)/cos(x)))

        tan(x)

In [7]: from sympy import*
        import random
        import functools
```



```
        tan(x)

In [7]: from sympy import*
        import random
        import functools
        import operator
        x,y,n=symbols('x y n')
        print(simplify(sin(x)-x**3*n))

        -n*x**3 + sin(x)

In [ ]:
```

2. Develop a python code for to carryout the operations on the given algebraic manipulation for

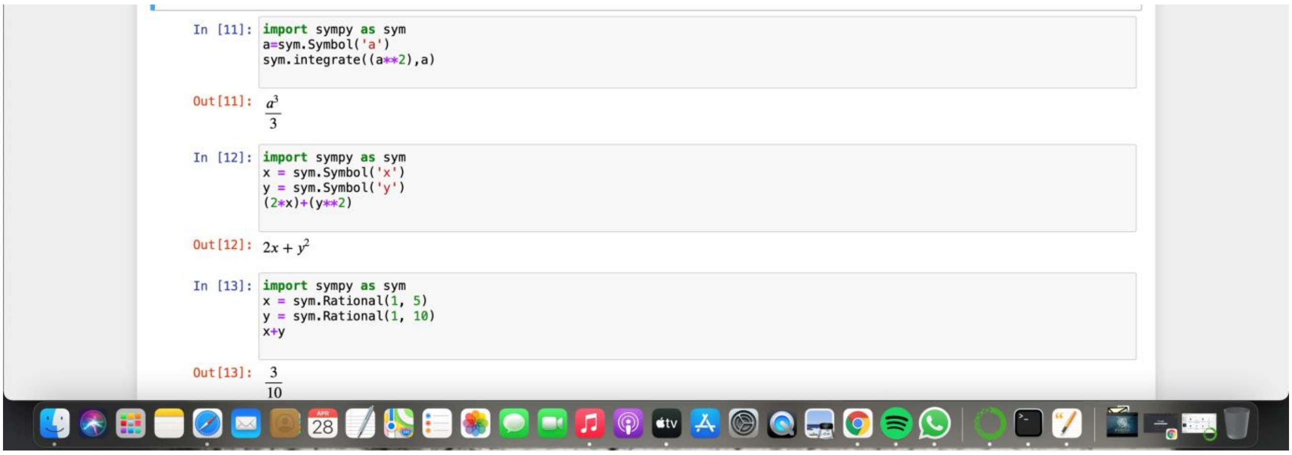
the given expression a 2 −ab+ab−b 2 =a 2 −b 2 by using the symbolic programming paradigms principles.


Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 11:33 AM
localhost
Exercise 7
Meet - htq-fdab-dbq
Home Page - Select or create a notebook
Untitled98 - Jupyter Notebook
jupyter Untitled98 Last Checkpoint: 38 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Notebook saved
Trusted
Python 3
Run
Code

```
In [8]: from sympy import*
        import random
        import functools
        import operator
        a,b,ab=symbols('a b ab')
        g1=simplify(a*2-ab+ab-b*2)
        print("a*2-ab+ab-b*2 = ",g1)

        a*2-ab+ab-b*2 =  2*a - 2*b
```

3. Give the Symbolic program for the expression given below: a. ∫∫a 2 da
b. 2x+y 2
c. 1/10 + 1/5

d. d/dx(sin(x))



```
In [11]: import sympy as sym
         a=sym.Symbol('a')
         sym.integrate((a**2),a)
```

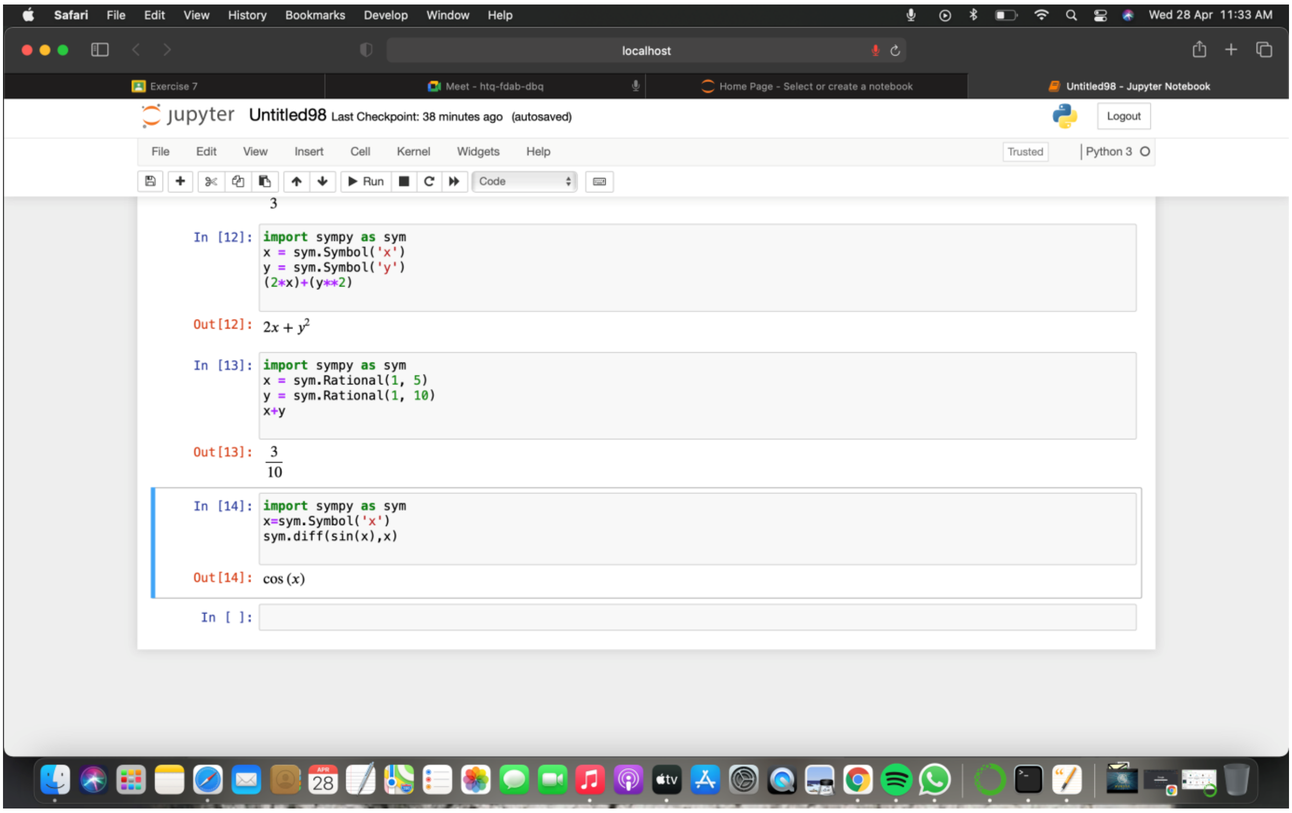
Out[11]: $\frac{a^3}{3}$

```
In [12]: import sympy as sym
         x = sym.Symbol('x')
         y = sym.Symbol('y')
         (2*x)+(y**2)
```

Out[12]: $2x + y^2$

```
In [13]: import sympy as sym
         x = sym.Rational(1, 5)
         y = sym.Rational(1, 10)
         x+y
```

Out[13]: $\frac{3}{10}$


Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 11:33 AM
localhost
Exercise 7
Meet - htq-fdab-dbq
Home Page - Select or create a notebook
Untitled98 - Jupyter Notebook
jupyter Untitled98 Last Checkpoint: 38 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
    3

In [12]: import sympy as sym
         x = sym.Symbol('x')
         y = sym.Symbol('y')
         (2*x)+(y**2)

Out[12]: 2x + y^2

In [13]: import sympy as sym
         x = sym.Rational(1, 5)
         y = sym.Rational(1, 10)
         x+y

Out[13]: 3/10

In [14]: import sympy as sym
         x=sym.Symbol('x')
         sym.diff(sin(x),x)

Out[14]: cos(x)

In [ ]:
```

Logic Programming:

1. Implement using pyDatalog:

Assume given a set of facts of the form father(name1,name2) (name1 is the father of name2).

a. Define a predicate brother(X,Y) which holds iff X and Y are brothers.

b. Define a predicate cousin(X,Y) which holds iff X and Y are cousins.

c. Define a predicate grandson(X,Y) which holds iff X is a grandson of Y.

d. Define a predicate descendent(X,Y) which holds iff X is a descendent of Y.

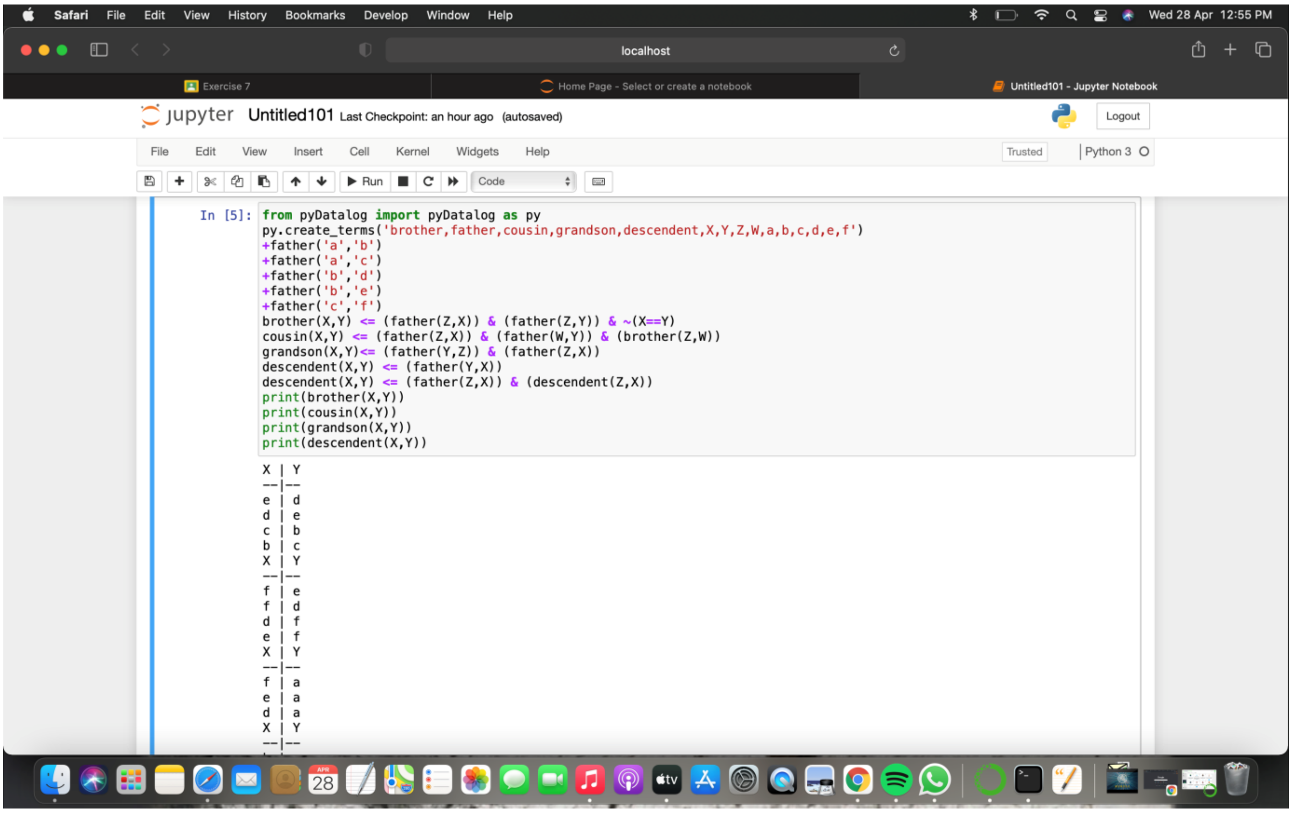
e. Consider the following genealogical tree:

a /\ bc /\| def

What are the answers generated by your definitions for the queries: brother(X,Y)

---

cousin(X,Y) grandson(X,Y) descendent(X,Y)



Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 12:55 PM
localhost
Exercise 7
Home Page - Select or create a notebook
Untitled101 - Jupyter Notebook
jupyter Untitled101 Last Checkpoint: an hour ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
In [5]: from pyDatalog import pyDatalog as py
        py.create_terms('brother,father,cousin,grandson,descendent,X,Y,Z,W,a,b,c,d,e,f')
        +father('a','b')
        +father('a','c')
        +father('b','d')
        +father('b','e')
        +father('c','f')
        brother(X,Y) <= (father(Z,X)) & (father(Z,Y)) & ~(X==Y)
        cousin(X,Y) <= (father(Z,X)) & (father(W,Y)) & (brother(Z,W))
        grandson(X,Y)<= (father(Y,Z)) & (father(Z,X))
        descendent(X,Y) <= (father(Y,X))
        descendent(X,Y) <= (father(Z,X)) & (descendent(Z,X))
        print(brother(X,Y))
        print(cousin(X,Y))
        print(grandson(X,Y))
        print(descendent(X,Y))
```

```
X | Y
--|--
e | d
d | e
c | b
b | c
X | Y
--|--
f | e
f | d
d | f
e | f
X | Y
--|--
f | a
e | a
d | a
X | Y
--|--
```

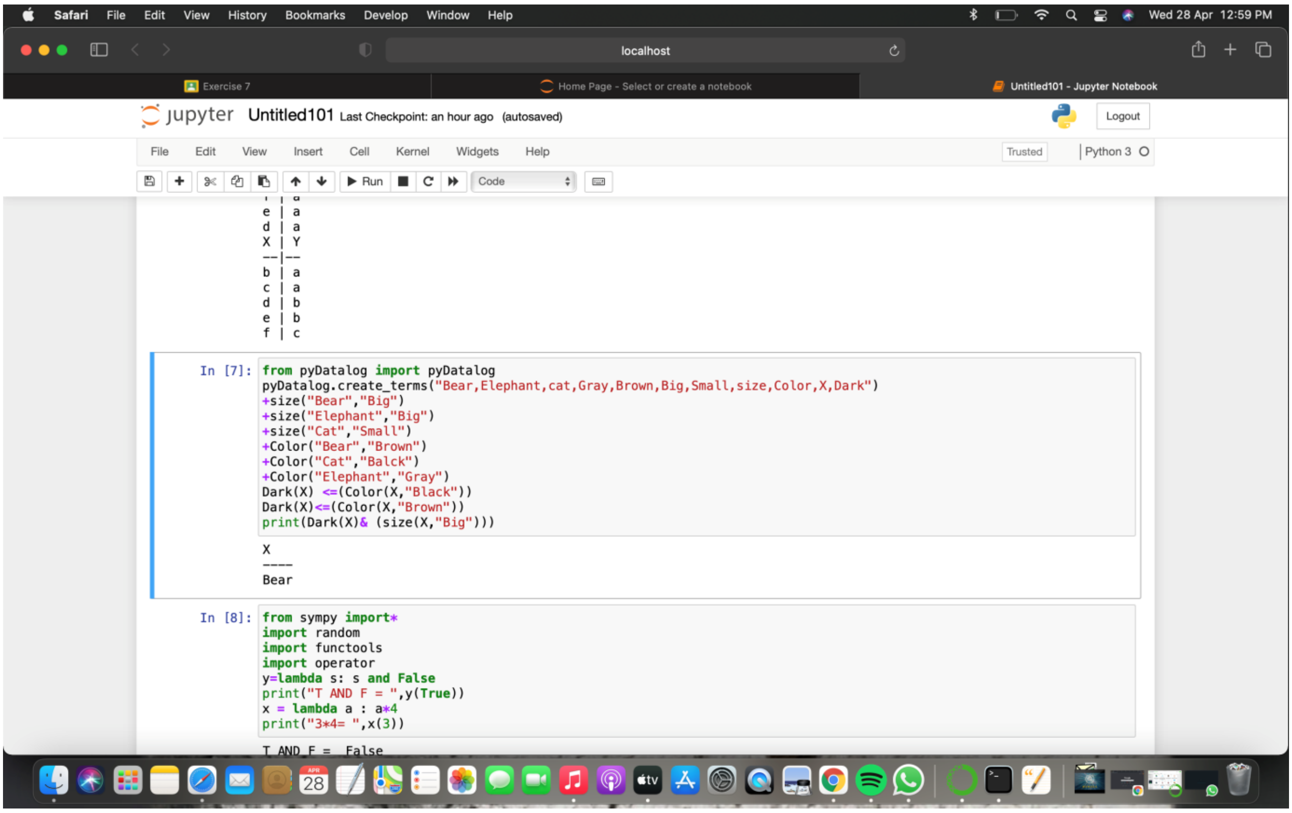
2. Encode the following facts and rules in pyDatalog: • Bear is big
• Elephant is big
• Cat is small

• Bear is brown

• Cat is black
• Elephant is gray

• An animal is dark if it is black • An animal is dark if it is brown

Write a query to find which animal is dark and big



Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 12:59 PM
localhost
Exercise 7
Home Page - Select or create a notebook
Untitled101 - Jupyter Notebook
jupyter Untitled101 Last Checkpoint: an hour ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
f | a
e | a
d | a
X | Y
--|--
b | a
c | a
d | b
e | b
f | c
```

```
In [7]: from pyDatalog import pyDatalog
        pyDatalog.create_terms("Bear,Elephant,cat,Gray,Brown,Big,Small,size,Color,X,Dark")
        +size("Bear","Big")
        +size("Elephant","Big")
        +size("Cat","Small")
        +Color("Bear","Brown")
        +Color("Cat","Balck")
        +Color("Elephant","Gray")
        Dark(X) <=(Color(X,"Black"))
        Dark(X)<=(Color(X,"Brown"))
        print(Dark(X)& (size(X,"Big")))

        X
        ----
        Bear
```

```
In [8]: from sympy import*
        import random
        import functools
        import operator
        y=lambda s: s and False
        print("T AND F = ",y(True))
        x = lambda a : a*4
        print("3*4= ",x(3))

        T AND F =  False
```

1. The following are the marks scored by 5 students.

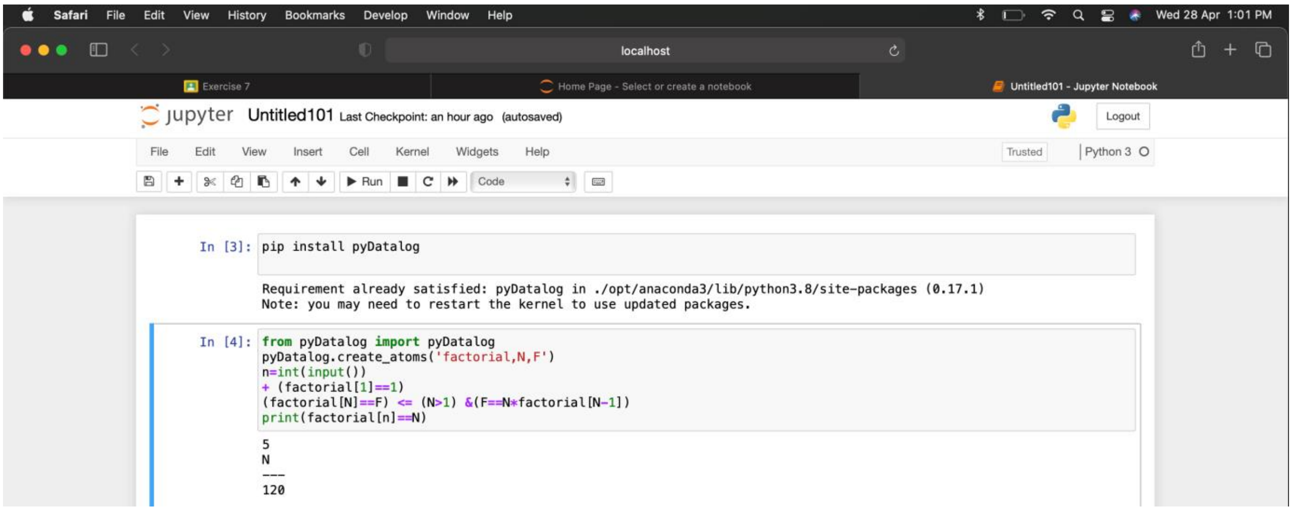
Student Name	Mark
Ram	90
Raju	45
Priya	85
Carol	70
Shyam	80

Enter the above data using pyDatalog.
Write queries for the following:
a. Print Student name and mark of all students.
b. Who has scored 80 marks?
c. What mark has been scored by Priya?
d. Write a rule ‘passm’ denoting that pass mark is greater than 50. Use the rule to print all students who failed.
e. Write rules for finding grade letters for a marks and use the rule to find the grade

letter of a given mark.

1. Solve the set of queries in the previous question using imperative programming paradigm in Python. Store the data in a dictionary.

1. Write a recursive program to find factorial of a number using pyDatalog.


Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 1:01 PM
localhost
Exercise 7
Home Page - Select or create a notebook
Untitled101 - Jupyter Notebook
jupyter Untitled101 Last Checkpoint: an hour ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
In [3]: pip install pyDatalog

        Requirement already satisfied: pyDatalog in ./opt/anaconda3/lib/python3.8/site-packages (0.17.1)
        Note: you may need to restart the kernel to use updated packages.

In [4]: from pyDatalog import pyDatalog
        pyDatalog.create_atoms('factorial,N,F')
        n=int(input())
        + (factorial[1]==1)
        (factorial[N]==F) <= (N>1) &(F==N*factorial[N-1])
        print(factorial[n]==N)

        5
        N
        ---
        120
```

Functional Programming:

1. Calculate the following using Lambda calculus: a. T AND F
b. 3 * 4
2. Lambda functions

a. Write a lambda function to convert measurements from meters to feet.
b. Write a lambda function in Python to implement the following lambda expression:

(. . (+))(. 2)()

Note: You need to write a nested lambda function for implementing f+m where f takes the square function (which takes argument x) passed as a parameter. The above expression calculates a 2 +b.

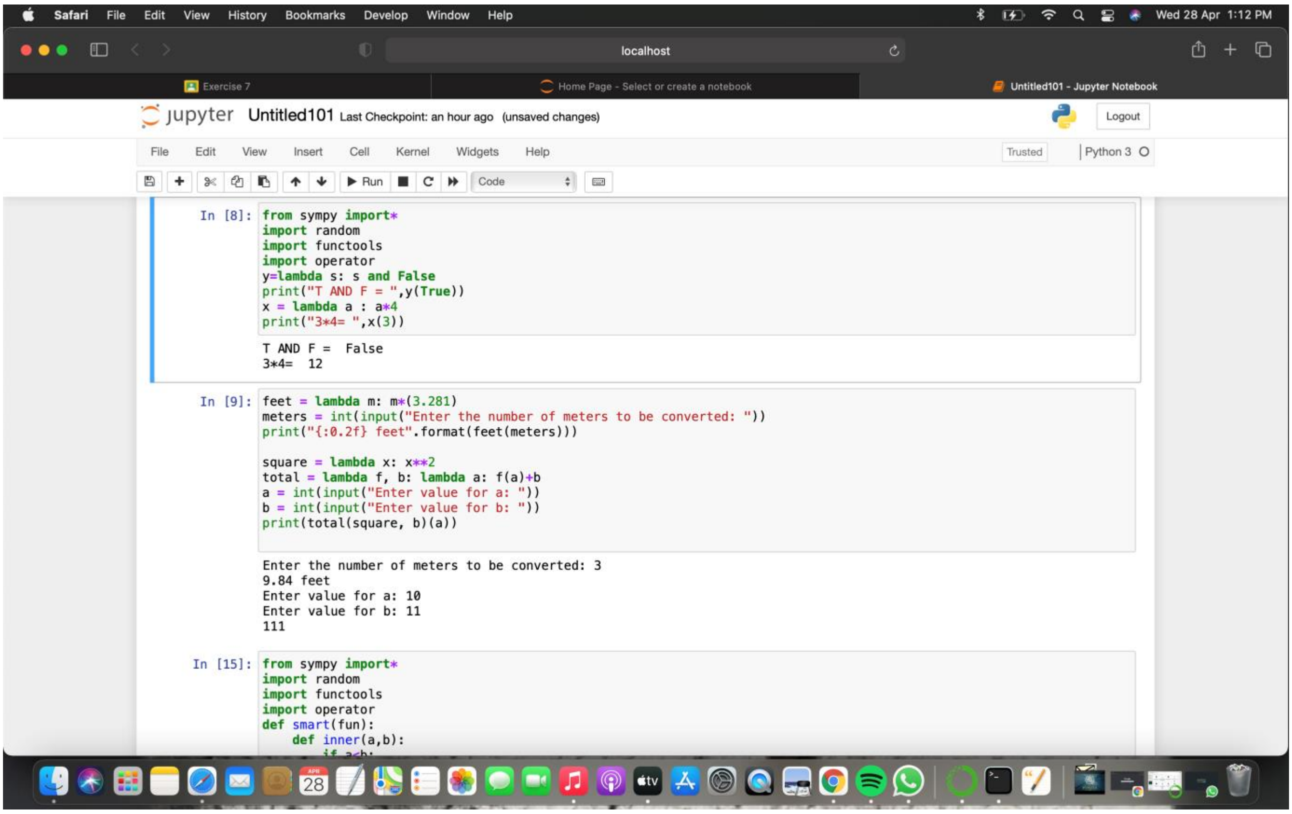
4. Closure
A Closure is a function object that remembers values in enclosing scopes even if they are not present in memory. We have a closure in Python when a nested function references a value in its enclosing scope.
a. Study the following program by executing it:

```
def multiplier_of(n):
def multiplier(number):
return number*n
return multiplier
multiplywith5 = multiplier_of(5) print(multiplywith5(9))
```

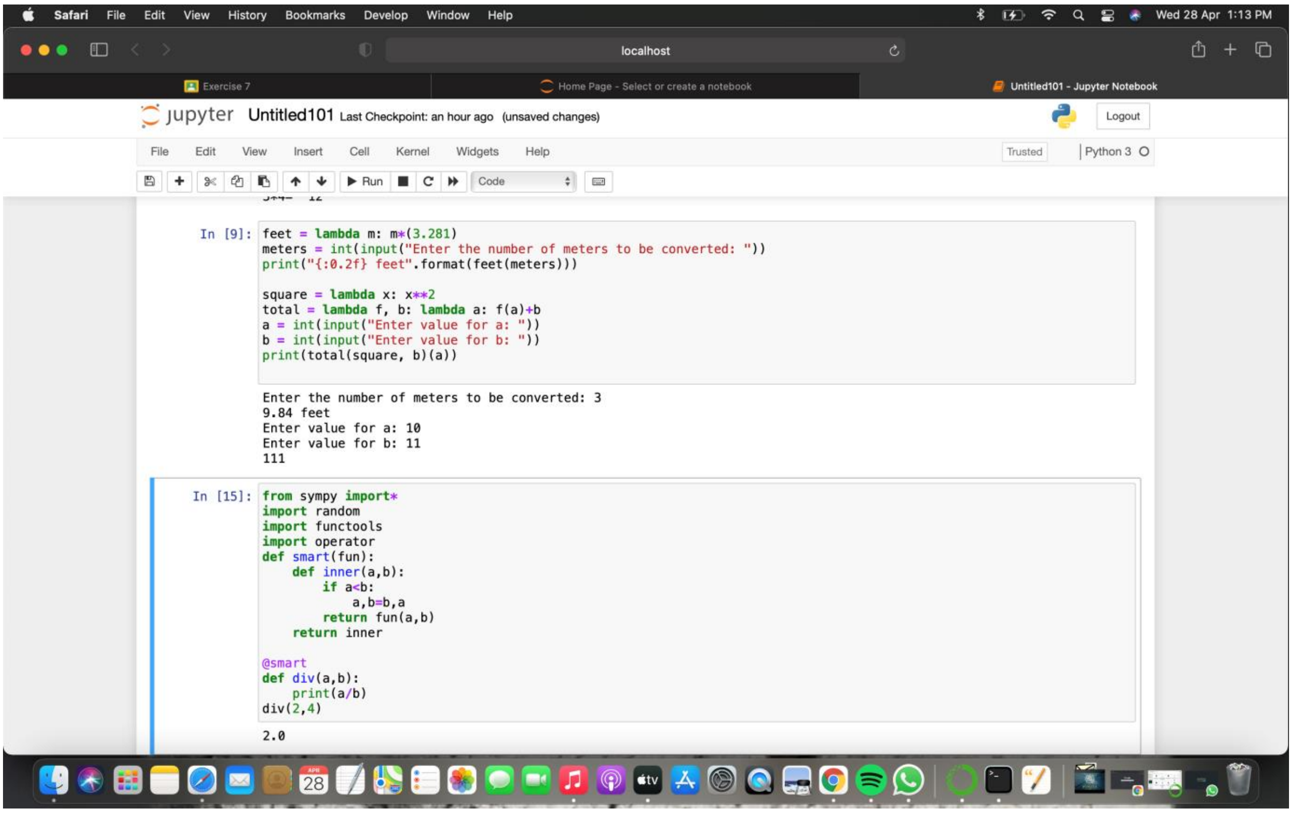
b. In a lottery system, random number is chosen by retrieving the number from a random index from a list of random numbers. Write a program to

choose a random number in this way. You must use nested functions – the inner function chooses a number from a random index and the outer function generates a random list of numbers. The outer function takes n as a parameter where is the maximum number that can be put in the random list. (Your code should be similar to the program in 5a)

6. Map
A secret message needs to be sent. Use the map function to encrypt the message using Caesar cipher.
7. Reduce
Given runs scored by 2 players in a series of matches, write a Python program using reduce function to
find who is the better player of the two in terms of maintaining consistency. (You need to find SD).
9. Map+reduce+filter
Given two trending topics and a bunch of tweets, write a Python program to count the number of tweets
that contain each topic. You need to do this by putting together map(), reduce() and filter() functions.


Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 1:12 PM
localhost
Exercise 7
Home Page - Select or create a notebook
Untitled101 - Jupyter Notebook
jupyter Untitled101 Last Checkpoint: an hour ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
In [8]: from sympy import*
        import random
        import functools
        import operator
        y=lambda s: s and False
        print("T AND F = ",y(True))
        x = lambda a : a*4
        print("3*4= ",x(3))

        T AND F =  False
        3*4=  12

In [9]: feet = lambda m: m*(3.281)
        meters = int(input("Enter the number of meters to be converted: "))
        print("{:0.2f} feet".format(feet(meters)))

        square = lambda x: x**2
        total = lambda f, b: lambda a: f(a)+b
        a = int(input("Enter value for a: "))
        b = int(input("Enter value for b: "))
        print(total(square, b)(a))

        Enter the number of meters to be converted: 3
        9.84 feet
        Enter value for a: 10
        Enter value for b: 11
        111

In [15]: from sympy import*
         import random
         import functools
         import operator
         def smart(fun):
             def inner(a,b):
                 if a<b:
```

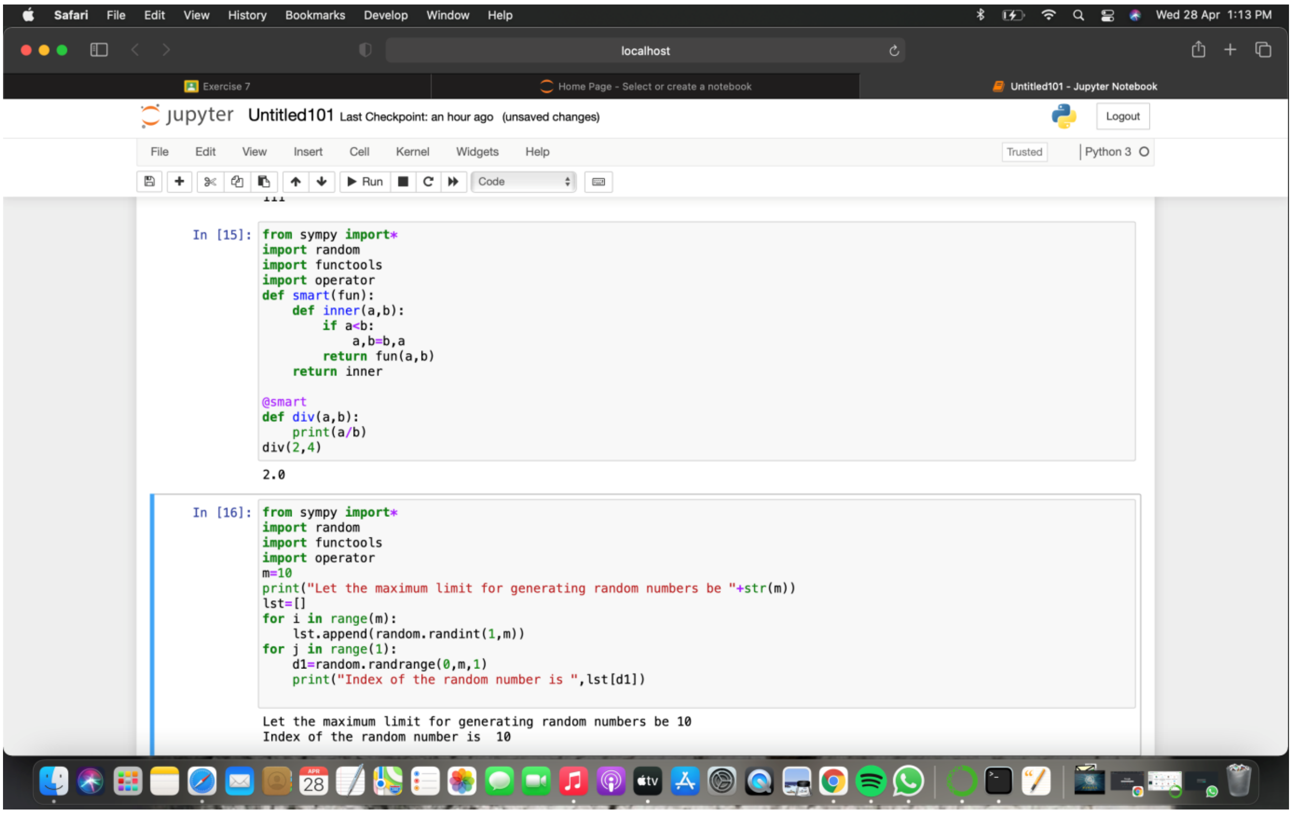

Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 1:13 PM
localhost
Exercise 7
Home Page - Select or create a notebook
Untitled101 - Jupyter Notebook
jupyter Untitled101 Last Checkpoint: an hour ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

In [9]:
```
feet = lambda m: m*(3.281)
meters = int(input("Enter the number of meters to be converted: "))
print("{:0.2f} feet".format(feet(meters)))

square = lambda x: x**2
total = lambda f, b: lambda a: f(a)+b
a = int(input("Enter value for a: "))
b = int(input("Enter value for b: "))
print(total(square, b)(a))
```

```
Enter the number of meters to be converted: 3
9.84 feet
Enter value for a: 10
Enter value for b: 11
111
```

In [15]:
```
from sympy import*
import random
import functools
import operator
def smart(fun):
    def inner(a,b):
        if a<b:
            a,b=b,a
        return fun(a,b)
    return inner

@smart
def div(a,b):
    print(a/b)
div(2,4)
```

```
2.0
```

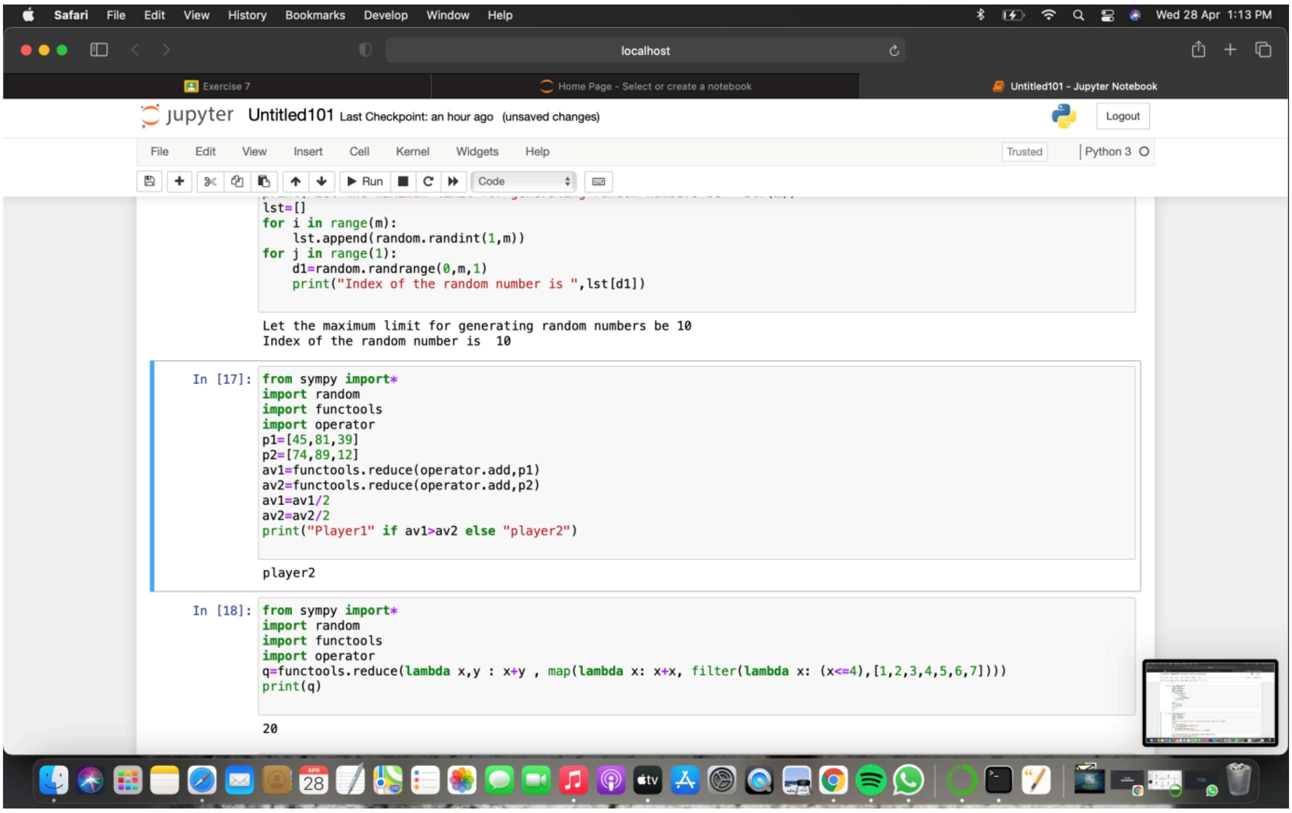

Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 1:13 PM
localhost
Exercise 7
Home Page - Select or create a notebook
Untitled101 - Jupyter Notebook
jupyter Untitled101 Last Checkpoint: an hour ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

In [15]:
```
from sympy import*
import random
import functools
import operator
def smart(fun):
    def inner(a,b):
        if a<b:
            a,b=b,a
        return fun(a,b)
    return inner

@smart
def div(a,b):
    print(a/b)
div(2,4)
```

```
2.0
```

In [16]:
```
from sympy import*
import random
import functools
import operator
m=10
print("Let the maximum limit for generating random numbers be "+str(m))
lst=[]
for i in range(m):
    lst.append(random.randint(1,m))
for j in range(1):
    d1=random.randrange(0,m,1)
    print("Index of the random number is ",lst[d1])
```

```
Let the maximum limit for generating random numbers be 10
Index of the random number is  10
```


Safari File Edit View History Bookmarks Develop Window Help
Wed 28 Apr 1:13 PM
localhost
Exercise 7
Home Page - Select or create a notebook
Untitled101 - Jupyter Notebook
jupyter Untitled101 Last Checkpoint: an hour ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
lst=[]
for i in range(m):
    lst.append(random.randint(1,m))
for j in range(1):
    d1=random.randrange(0,m,1)
    print("Index of the random number is ",lst[d1])
```

```
Let the maximum limit for generating random numbers be 10
Index of the random number is  10
```

In [17]:

```
from sympy import*
import random
import functools
import operator
p1=[45,81,39]
p2=[74,89,12]
av1=functools.reduce(operator.add,p1)
av2=functools.reduce(operator.add,p2)
av1=av1/2
av2=av2/2
print("Player1" if av1>av2 else "player2")
```

```
player2
```

In [18]:

```
from sympy import*
import random
import functools
import operator
q=functools.reduce(lambda x,y : x+y , map(lambda x: x+x, filter(lambda x: (x<=4),[1,2,3,4,5,6,7])))
print(q)
```

```
20
```