```
1+1
```

```
2
```

```
'adas'
```

```
'adas'
```

```
pwd
```

```
'/content'
```

```
import this
```

```
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!
```

```
print("this is my 1st program")
```

```
this is my 1st program
```

```
1+2
```

```
3
```

```
3/7
```

```
0.42857142857142855
```

Not declare as datatype in python

```
m=670
m
```

```
670
```

```
type(m)
```

```
int
```

```
d=5+7j
d
```

```
(5+7j)
```

```
type(d)
```

```
complex
```

```
e='m'
e
```

```
'm'
```

coding is 20% of coding & 80% of debugging

```
g=True
type(g)
```

```
bool
```

```
_a=45
_a
```

```
45
```

```
a=34
a
```

```
34
```

```
a=5464
b='m'
c=6+7j
d=True
e=546.7
```

```
a
```

```
5464
```

```
a1=90
```

```
a+a1
```

```
5554
```

All this variable into Single line and phsase those sequentially

```
a,b,c,d,e="adas",3,8.9,7+9j,False
```

```
a
```

```
'm'
```

```
b
```

```
3
```

```
c
```

```
8.9
```

Fetch Real & imaginary data

```
d.real
```

```
7.0
```

```
d.imag
```

```
9.0
```

we can get error until and unless 4 to convert string

```
a+4
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-52-36b7dce91f81> in <module>
----> 1 a+'4'

TypeError: can only concatenate str (not "int") to str

SEARCH STACK OVERFLOW
```

typecasing in another way

```
a+str(4)
```

```
'adas4'
```

True-->1

True-->1

```
True+True
```

```
2
```

```
True-False
```

```
1
```

input by default take string function

```
a=input()
```

```
9
```

```
a
```

```
'9'
```

```
a+9
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-59-dbb0651420c8> in <module>
----> 1 a+9

TypeError: can only concatenate str (not "int") to str

SEARCH STACK OVERFLOW
```

```
int(a)+9
```

```
18
```

find Memory variable

```
id(a)
```

```
139975256122928
```

✓ 0s completed at 9:29 PM