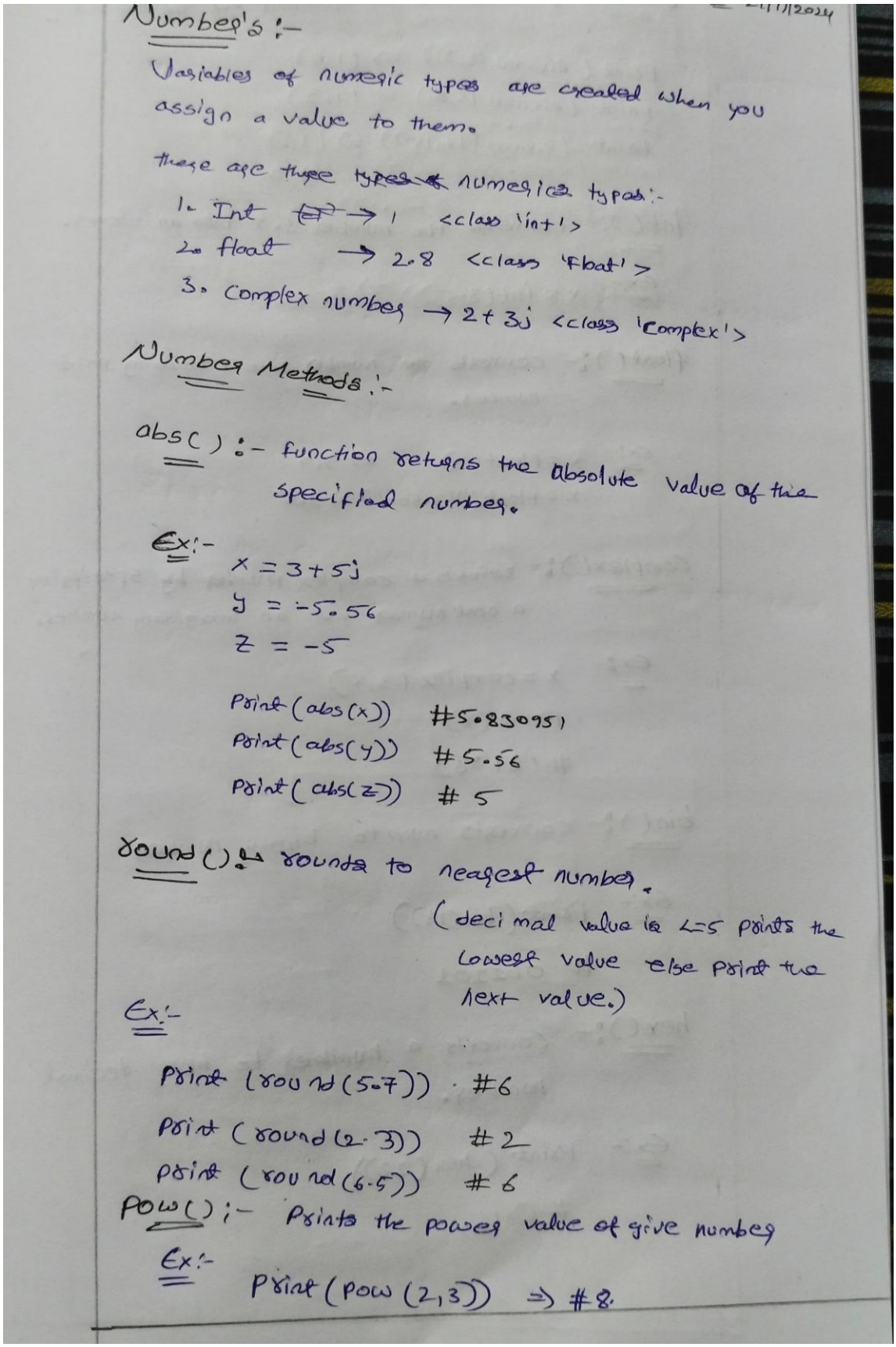
Number and Number methods:



Numbers :-

Variables of numeric types are created when you assign a value to them.

these are three types of numerical types :-

1. Int → 1 <class 'int'>
2. float → 2.8 <class 'float'>
3. Complex number → 2+3j <class 'complex'>

Number Methods :-

abs() :- function returns the absolute value of the specified number.

Ex :-

```
x = 3+5j
y = -5.56
z = -5

Print(abs(x))   #5.830951
Print(abs(y))   #5.56
Print(abs(z))   #5
```

round() :- rounds to neagest number.
(decimal value is <=5 prints the lowest value else print the next value.)

Ex :-

```
Print(round(5.7))   #6
Print(round(2.3))   #2
print(round(6.5))   #6
```

pow() :- Prints the powers value of give number

Ex :-

```
Print(pow(2,3))  => #8.
```

divmod():- it returns the quotient and the remainder.

print (divmod (3,3)) => (1,0)
print (divmod (5,3)) => (1, 2)
print (divmod (17,15)) => (1,2)

int():- converts the number 3.5 into an integer.

Ex:- x = int(3.5) => 3

float():- converts the number into a floating point number.

Ex:- x = float(3) => 3.0
x = float("3.500") => 3.5

Complex():- returns a complex number by specifying a real number and an imaginary number.

Ex:-
x = complex(3,5)
print(x)
#(3+5j)

bin():- converts num to binary number.

Ex:- Print (bin(13))
0b1101

hex():- Converts a number to hexa decimal number.

Ex:- print (hex(22))
#0x16

oct():- converts a number into the oct number

Ex:-

```
print (oct(16))
#0020
```

Aggregate Functions:-

max():- returns the max value of list or tuple

Ex:-

```
a = (2,3,4,5,7,8,9)
print (max(a))
#9
```

Min():- returns the minimum value

```
a = (2,3,4,5,6,7,8)
print (min(a))
#2
```

Sum():- returns the sum value.

```
print (sum(2,10,8,3,7))
#30
```

Avg():-

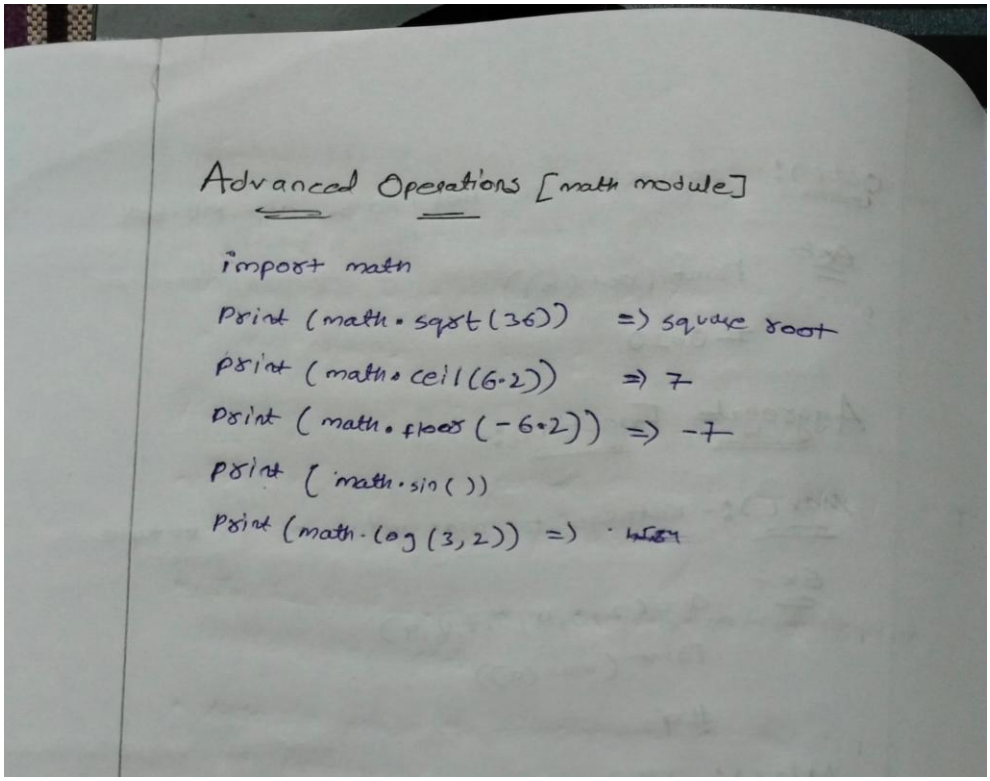
Example:-(without methods)

```
tup = (1,2,3,4)
max = 0

for i in tup:
    if i > max:
        max = i
print (max)
```

(or)

```
max = tup[0]
for i in tup:
    if i > max:
        max = i
print (max)
```



Advanced Operations [math module]

```
import math
Print (math.sqrt(36))      => square root
Print (math.ceil(6.2))     => 7
Print (math.floor(-6.2))   => -7
Print (math.sin())
Print (math.log(3,2))      =>
```