

IDE : Integrated Development Environment

PyCharm

PyZoo

PyScripter

Spyder

Anaconda

Atom

Visual studio

Eclipse

i) File → New

```
print ("Hello")
```

File → Save → pl.py ↴

▶ [F5]

Shell

Hello

2] File → New

```
print (" Welcome to Infopark ")
```

File → Save → p2.py ↴

▶ [F5]

Shell

```
Welcome to Infopark
```

3] File · New

```
print (" Hello \n World ")
# \n : newlines
```

File → Save → p3.py ↴

▶ [F5]

Shell

```
Hello
World
```

4) [P4.py]

print ("Hello")

print ("World")

print adds \n after o/p

O/p

Hello

World

[P5.py]

print ("Hello", end = "+")

print ("World")

O/p

Hello + World

[P6.py]

```
print ("Hello", end = " ")  
print ("World")
```

O/P

Hello World

-

[P7.py]

```
print ("Hello", end = "\t") # \t : tab  
print ("World")
```

O/P

Hello World

[P8.py]

```
print ("Hello", end = ":")  
print ("World")
```

O/P

Hello : World

[P9. py]

```
print ("Hello")
```

```
print ('World')
```

O/P

Hello
World

[P10. py]

```
s1 = "New"
```

```
print (s1)
```

```
s2 = 'York'
```

```
print (s2)
```

O/P

New
York

[P11. py]

```
s1 = "New"
```

```
s2 = "York"
```

```
print (s1, s2) # by default separated by space
```

O/P

New York

p12.py

S1 = "New"

S2 = "York"

print (S1, S2, Sep = "+")

O/P

New + York

p13.py

S1, S2 = "New", "York"

print (S1, S2, Sep = "\t")

O/P

New York.

p14.py

S1, S2 = "Raj", "Amit"

print (S1, S2, Sep = ":")

O/P

Raj : Amit

[P15.py]

`s1, s2, s3 = "Raj", "Amit", "Joe"`

`print(s1, s2, s3, sep = "\n")`

[P16.py]

`s1, s2 = "Raj", "Amit"`

`s3, s4 = "Joe", "Jack"`

`print(s1, s2, sep = "+", end = "\t")`

`print(s3, s4, sep = ":")`

O/P

Raj + Amit

Joe : Jack

[P17.py]

- `Print("Hello") # error`

- `Print("Raj")
print # error`

P19. PY

$s_1 = "New"$
 $s_2 = " York"$
 $s_3 = s_1 + s_2$

+ : join (concatenation)
 $\text{print}(s_1, s_2, s_3, \text{sep} = "\n")$

O/P

New
 York
 New York

P20. PY

$s_1 = "Raj"$
 $s_2 = " More"$
 $s_3 = s_1 + " " + s_2$

$\text{print}(s_1, s_2, s_3, \text{sep} = "\n")$

O/P

Raj
 More
 Raj More

[P21.py]

name = "Raj"

msg = "Hello" + name
print (msg)

O/P

Hello Raj

[P22.py]

name = "Raj"

print ("Hello" + name)

O/P

Hello Raj

[P23.py]

s1 = "New"

s2 = "York"

print (s1 + s2)

O/P

New York

[P24. py]

name = "Amit"
city = "Pune"

print ("name : " + "lives in" + city)
o/p
Amit lives in Pune.

[P25. py]

name = "Raj"

print ("Hello {name}")
print (f "Hello {name}")

o/p
Hello Raj.

[P26. py]

name = "Amit"
city = "Pune"

print (f "{name} lives in {city}")

o/p

Amit lives in city. Pune.

There must be no space between
F & " ✓ F" X F "

M	T	W	T	F	S	S
Page No.:	YOUVA					
Date:						

[P27.py]

```
name = "Raj"  
print ( f "Hello {name}")
```

```
print (F "Hello {name}")
```

O/P

Hello Raj
Hello Raj

[P28.py]

```
print (" enter name : ", end = " ")
```

```
name = input()
```

```
print ( f "Hello {name}")
```

O/P

enter name : Ravi ↴
Hello Ravi

* variable $\xrightarrow{\text{o/p}}$ monitor
variable $\xleftarrow{\text{i/p}}$ keyboard

\downarrow = input()
str

[P29.py]

```
name = input ("enter name :")  
print (* ("Hello {name}"))
```

O/P

Enter name : Amit
Hello Amit.

[P30.py]

```
name = input ("enter name :")  
city = input ("enter city :")  
print (* "{name} lives in {city}")
```

O/P

enter name : Amit ↴
enter city : Pune ↴
Amit lives in Pune.

[P31.py]

```
# str : " " . . .  
s1 = "roj" # lowercase  
print (s1)  
s2 = "AMIT" # uppercase  
print (s2)  
s3 = "New York"  
print (s3)  
s4 = "45" # digits
```

M	T	W	F	S	S
Page No.:	YOUVA				
Date:					

```

print (s4)
S5 = " Room No 45" # uc LC Ø
print (s5)
S6 = " Room # 45 1000
print (s6)

```

[P36. py]

```

S1 = "20"    # str
S2 = "30"    # str

```

```

S3 = S1 + S2      →      S3
# S3 = "20" + "30"

```

```

print (S1, S2, S3) →      o/p

```

```

x = int (S1)          20

```

```

y = int (S2)          30

```

```

z = x + y

```

```

print (x, y, z) →      o/p

```

20 30 50

[P37. py]

```

x = 10

```

```

y = 20

```

```

z = x + y

```

```

print (f"addition is {z}")

```

o/p

addition is 30

P38.py

$x, y = 10, 20$

print ("addition is " + {x+y})

O/P
addition is 30.

P39.py

s1 = input ("enter no 1 : ")

s2 = input ("enter no 2 : ")

x = int(s1)

y = int(s2)

z = x + y

print ("addition is " + {z})

O/P

20, 30

addition is 50

P40.py

x = int(input ("enter no 1 : "))

y = int(input ("enter no 2 : "))

print ("addition is " + {x+y})

O/P

enter no 1 : 100
 enter no 2 : 60
 addition is 160.

P41.py

```
print ("enter 2 no.s :")
x,y = int(input()), int(input())
S = x - y
m = x * y
```

```
print ("Sub is {S} \n mul is {m}")
```

x	y	S	m
20	5	15	100

enter 2 no.s:

20 ↴

5 ↴

Subtraction is 15

Mult is 100.

P42.py

```
x,y = int(input("enter 2 no.s : \n")), int(input())
p = x ** y # ** : raise to : xy
```

```
print ("power : {p}")
```

O/P
enter two no.s:
5 ↴
4 ↴
power : 625

p43.py

```
x = int(input('enter x:'))
s = x ** 2
c = x ** 3
print(f"square : {s} \n cube : {c}")
```

x	s	c
10	100	1000

enter a no. : 10 ↴
square : 100
cube : 1000

p44.py

```
name = input("enter name :")
age = int(input("enter age :"))
```

print(f"{name} is {age} years old")

O/P

enter name : Raj ↴
enter age : 20 ↴

Raj is 20 years old

$l \cdot b = \text{int}(\text{input}("enter len : ")), \text{int}(\text{input}("enter bre : "))$

$$a = l * b$$

$$p = 2 * (l+b)$$

`print(f" area : {a} \n parameter : {p}")`

l	b	a	p
20	8	160	56

enter len : 20 ↴

enter bre : 8 ↴

area : 160

parameter : 56

put.py

$$x = 14 // 3$$

`print(x)`

o/p

4

y
8

$$y = .62 // 7$$

`print(y)`

o/p

8

M	T	W	T	F	S	S
Page No.:						
Date:						

Quo //
Rem % Modulus
(Mod)

✓ P48.py

a = 14 % 3
print (a)

O/P

2

b = 62 % 7
print (b)

O/P

6

✓ P49.py

i/p 2 no.s & cal quo & rem

print ("enter 2 no.s :")

x,y = int(input()), int(input())

x,y = int(input()), int(input())

q = x // y

r = x % y

print(f"quo:{q} \nrem:{r}")

7
5 | 39

35
04

x	y	q	r
39	5	7	4

enter 2 nos :

39 ↴

5 ↴

quo : 7

rem : 4

- P50-PY1

`x, y = int(input("enter 2 nos :\n")), int(input())`

`q, r = x // y, x % y`

`print(f"quo : {q}\nrem : {r}")`

cal rem of x div by without using %

i/p 2 digit no. ϕ cal Σ of digits

eg.

$$\begin{array}{c} \alpha \\ 48 \\ \downarrow \quad \downarrow \\ 4 + 8 \\ 12 \text{ o/p} \end{array}$$

[P51.py]

`x = int(input("enter 2 digit no. :"))`

`d1 = x // 10`

`d2 = x % 10`

`sum = d1 + d2`

`print(f"sum of digits : {sum}")`

[P52.py]

`x = int(input("enter 2 digit no. :"))`

`d1, d2 = x // 10, x % 10`

`sum = d1 + d2`

`print(f"sum of digits : {sum}")`

[P53.py]

```
x = int(input("enter a digit no.:"))
```

```
sum = x // 10 + x % 10
```

```
print(f"sum of digits: {sum}")
```

i/p 2 digit no. & calculate sum of original & reverse no.

x	ON : 48 (i/p)
48	RN : <u>+84</u> calculate.
	132 → o/p

i/p 2 nos. in x & y & swap them

[P54.py]

```
print("enter 2 nos.:")
```

```
x, y = int(input()), int(input())
```

```
print(f"before swap x: {x} y: {y}")
```

z = x

x = y

y = z

x	y	z
50	70	50
70	50	

print (" enter after swap x:{x} y:{y}")

enter 2 no.s:

50 ↴

70 ↴

before swap x: 50 y: 70
after swap x: 70 y: 50

[p55.py]

print (" enter 2 no.s : ")

x,y = int(input()), int(input())

print (" before swap x: {x} y: {y}")

x,y = y, x

print (" after swap x: {x} y: {y}")

without using 3rd var

enter 2 no.s :

50 ↴

70 ↴

before swap x: 50 y: 70
after swap x: 70 y: 50

[P56.py]

```
print ("Hello")
```

O/P
Hello

```
print ('Hello')
```

Hello
Hello

```
print ("   " "Hello" "   ")
```

Hello .

```
print (" " "Hello" " ")
```

[P57.py]

"""
This is a demo
pgm to represent
diff string
"""

print (" " This is a pgm.
To show multiline string.
This is allowed " ")

O/P

This is a pgm
To show multiline string
This is allowed .

[P 58.py]

```
print ('Hello')  
print ('World')  
print ('Asia')
```

o/p
Hello
World
Asia

[P 59.py]

```
print ("Hello  
World  
Asia")
```

o/p
Hello
World
Asia

[P 60.py]

```
print ('Hello'  
'World'  
'Asia')
```

o/p
helloworldAsia

P61.py

```
print ("Hello \n"
      "world \n"
      "asia")
```

O/P
Hello
World
asia

P62.py

x
10.11

O/P

```
print(x)
```

10.11

y
20.22

O/P

y = 20.22

20.22

z
30.33

$z = x + y$

```
print(f" add is {z}")
```

O/P

30.33

[P63.py]

x
10.11

$x, y = 10.11, 20.22$

y
20.22

`print(x, y, sep = "\n")`

`print(f"add is {x+y}")`

O/P

10.11

20.22

add is 30.33.

[P64.py]

x
10.1111

$x = 10.1111$

`print(f"x is {x}")`

y
10.11

$y = \text{round}(x, 2)$

z
10.1111

`print(f"y is {y}")`

$z = x$

`print(f"z is {z:.2f}")`

O/P

x is 10.1111

y is 10.11

z is 10.11

P65.py

x = 10.6666
print(f" x is {x}")

y = round(x, 2)

print(f" y is {y}")

z = x

print(f" z is {z} :. 2f")

O/P

x is 10.6666

y is 10.67

z is 10.67

P66.py

s = input("enter a no.: ")

x = float(s)

Square = x * x

print(f" square is {square :. 2f}")

"1.3" 1.3 1.69

enter a no.: 1.3 ↴

square is 1.69

[P67.py]

```
x = float ( input ("enter a no.:"))  
square = x * * 2  
print ("square is", square, "2f")
```

[P68.py]

```
print ("enter 2 nos.:")  
x, y = float (input ()), float (input ())
```

```
print ("addition is", x+y, "2f")
```

x y

10.11 20.22

enter 2 nos.:

10.11 ↴

20.22 ↴

addition is 30.33.

[P69.py]

x y
2 8

```
x = 10 // 5    # int : quo  
y = 34 // 4    # int : quo
```

```
print (x, y)
```

o/p

2 8

a = 10 / 5 # float : quo

print (f" {a:.2f}")

%/p

2.00

a

2.0 b = 34 / 4 # float : quo

b
8.5

print (f" {b:.2f}")

%/p

8.50

[PYO.PY]

i/p 2 no.s of cal. mean on avg.

print ("enter 2 no.s : ")

x, y = int(input()), int(input())

z = (x+y) / 2

print (f" Avg : {z:.2f}", z)

x	y	z
10	15	12.5

enter 2 no.s :

10 ↴

15 ↴

Avg : 12.50

176.625

M	T	W	T	F	S
Page No.:					

Date:

$$3.14 * 7.5 * 7.5 = 176.625$$

[PI, PY]

i/p radius of circle and cal. area

PI

3.14

$$\text{PI} = 3.14$$

r = float (input ("enter radius :"))

$$a = \text{PI} * r * r$$

$$\# a = \text{PI} * r * r * 2$$

print (f "area is { a:.2f }")

%p

enter radius : 7.5 ↵

area is 176.625

i/p base & height of Δ & cal. area

$$a = 0.5 * b * h$$

[PT2.py]

bill = float(input("enter bill :"))

discount = bill * 10 / 100

total_bill = bill - discount

print(f" discount : {discount:.2f} Rs")

print(f" total bill : {total_bill:.2f} Rs")

enter bill : 300.0

discount : 30.00 Rs

total bill : 270.00 Rs

Q/P bill : 10% disc

bill

200.0

$$dis : 200.0 * \frac{10}{100} : 20.0$$

$$\text{total bill} : 200.0 - 20.0 : 180.0$$

O/P {Hello} → print(f" ")

PT3.py

S1

"Hello"

"NewYork"

S2

"New"

"New York"

S1 = "New"
S1 = S1 + "York"

print (S1) O/P
 NewYork

S2 = "New"

S2 += "York" ↓

S2 = S2 + "York"

print (S2) O/P
 NewYork

PT4.py

S1

"New"

S2

"York"

S1 = "New"

S2 = "York"

S3 = S1

S3 += S2

print (S1, S2, S3, sep = "\n")

S3

"New"

"NewYork"

O/P

New

York

NewYork

PT5.py

s1, s2 = "Raj", "More"

s1

"Raj"

s2

"More"

s3 = s1

s3 s3+= " "

s3 += s2

s3

" Raj "

" Raj "

" Raj More "

print(s1, s2, s3, sep = "\n")

O/P

Raj

More

Raj More.

PT6.py

s1 s1 = "\$" * 5

"\$\$\$\$\$"

print(s1)

O/P

\$ \$ \$ \$ \$

s2

s2 = "Raj" * 3

"RajRajRaj"

print(s2)

O/P

RajRajRaj

P77.py

... \$
... \$ \$ \$
\$ \$ \$ \$ \$
\$ \$ \$ \$ \$ \$

print(" " * 3 + "\$")

print(" " * 2 + "\$" * 3)

print(" " + "\$" * 5)

print("\$" * 7)

P78.py

name = input("enter name : ") + "\n"

c = int(input("enter count : "))

print(name * c)

print("Sam\n" * 3)

name
"Sam\n" c
3

enter name : sam ↴

enter count : 3 ↴

Sam

Sam

Sam

Decision Making statement :

if st.

if - else st

Nested if - else st

if - elif - else . st

single line if-else st.

i) if statement

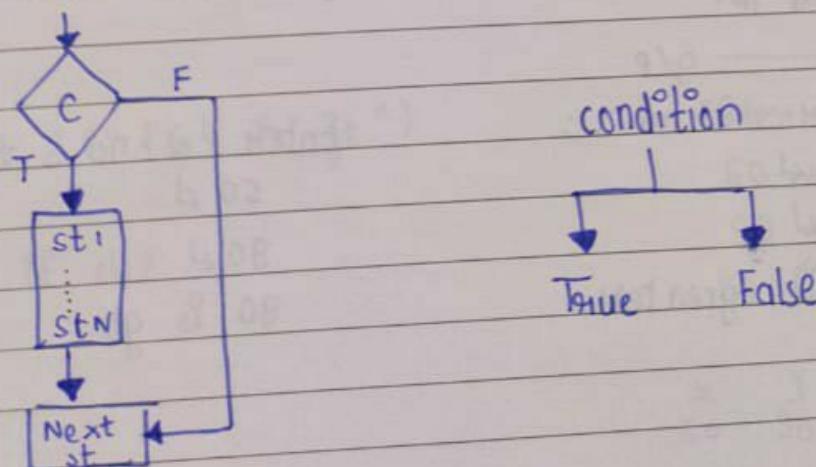
syntax

```
if condition :  
    st 1  
    st 2  
    :  
    st N  
    # end if  
    next statement
```

bool : boolean

True False
↑ ↑

flow chart



T	$10 > 5$	$20 > 80$	F	$10 > 10$	F
T	$10 >= 5$	$20 >= 80$	F	$10 >= 10$	T
F	$10 < 5$	$20 < 80$	T	$10 < 10$	F
F	$10 <= 5$	$20 <= 80$	T	$10 <= 10$	T
T	$10 != 5$	$20 != 80$	T	$10 != 10$	F
F	$10 == 5$	$20 == 80$	F	$10 == 10$	T

IF : 2 condition

Relational operator.

>	\geq	$<$	\leq	$!=$	$==$
Gt	Gt	LT	LT	Nt	Ea
ea		Ea		Ea	

[p79 . py]

```

print ("Enter 2 no.s")
x, y = int(input()), int(input())
if x > y:
    print ("{} is gt".format(x))
# end if
if x < y:
    print ("{} is gt".format(y))
# end if

```

O/P

Enter 2 no.s :

40 ↴

10 ↴

40 is greater

Enter 2 no.s :

60 ↴

90 ↴

90 is gt.

a) if - else statement

Syntax :-

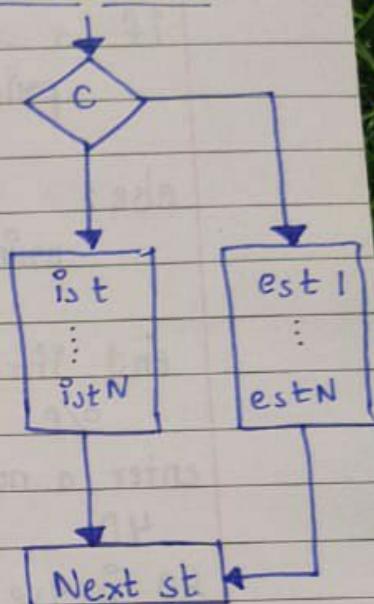
```

if condition:
    ist1
    ist2
    s:
    istN
else:
    est1
    est2
    :
    estN
# end of if - else
Next_st
  
```

T: if
F: else

if - else - 1 condition

Flow chart



P80.py

```

print ("Enter 2 no.s")
x,y = int(input()),int(input())
if x > y:
    print(f"{x} is gt")
  
```

```

else:
    print(f"{y} is gt")
  
```

end if else.

O/P x y
 20 10

Enter 2 no.s
40
10
40 is gtr

Enter 2 no.s
60
90
90 is gtr

x y
60 90

[P81.py]

i/p and o/p whether it is +ve or -ve
(do not consider zero)

```
x = int(input("Enter a no.:"))
if x > 0:
    print(f"{x} is +ve")
```

```
else:
    print(f"{x} is -ve")
```

end if-else

o/p

enter a no: 40	enter a no: -10
40	-10
40 is +ve	-10 is -ve

i/p Age of a person o/p whether it is senior citizen or not

age above 60 → senior citizen

otherwise → not senior citizen.

Nested if - else :

One if - else st is written inside another if - else st.

Syntax

→ if condition - one :

— block - one

else :

— if condition two :

— — block - two

— else :

— — block three

— # end : if - else (inner)

— # end : if - else (outer)

next st

T if

F else

block:
collection of python sts.

st1
st2
st3
:
stN

Flow chart:

