Name:-Tej Sutar Batch:-H Roll no:-176 Date:-25/01/25 Practical no:-9 & 10 Practical name:- Study of Operation Research in Python(LPP)

Q1) Write a Python program to solve the following LPP: Max Z = 150x + 75y subject to 4x + 6y ≤ 24 5x + 3y ≤ 15 x ≥ 0, y ≥ 0

In [13]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMaximize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
lpp+=(4*x+6*y<=24)
lpp+=(5*x+3*y<=15)
lpp+=(150*x+75*y)
lpp
```

Out[13]:
```
LPP:
MAXIMIZE
150*x + 75*y + 0
SUBJECT TO
_C1: 4 x + 6 y <= 24

_C2: 5 x + 3 y <= 15

VARIABLES
x Continuous
y Continuous
```

In [14]:
```
lpp.solve()
```

Out[14]:
```
1
```

In [15]:
```
lpp.objective.value()
```

Out[15]:
```
450.0
```

In [16]:
```
x.value()
```

Out[16]:
```
3.0
```

In [17]:
```
y.value()
```

Out[17]:
```
0.0
```

Q2) Write a Python program to solve the following LPP: Max Z = 5x + 3y subject to x + y ≤ 7 2x + 5y ≤ 1 x ≥ 0, y ≥ 0.

In [18]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMaximize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
lpp+=(x+y<=7)
lpp+=(2*x+5*y<=1)
lpp+=(5*x+3*y)
lpp
```

Out[18]:
```
LPP:
MAXIMIZE
5*x + 3*y + 0
SUBJECT TO
_C1: x + y <= 7

_C2: 2 x + 5 y <= 1

VARIABLES
x Continuous
y Continuous
```

In [19]:
```
lpp.solve()
```

Out[19]:
```
1
```

In [20]:
```
lpp.objective.value()
```

Out[20]:
```
2.5
```

In [21]:
```
x.value()
```

Out[21]:
```
0.5
```

In [22]:
```
y.value()
```

Out[22]:
```
0.0
```

Q3) Write a Python program to solve the following LPP: Max Z = x + y subject to 2x − 2y ≥ 1 x + y ≥ 2 x ≥ 0, y ≥ 0

In [23]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMaximize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
lpp+=(2*x-2*y>=1)
lpp+=(x+y>=2)
lpp+=(x+y)
lpp
```

Out[23]:
```
LPP:
MAXIMIZE
1*x + 1*y + 0
SUBJECT TO
_C1: 2 x - 2 y >= 1

_C2: x + y >= 2

VARIABLES
x Continuous
y Continuous
```

In [24]:
```
lpp.solve()
```

Out[24]:
```
-2
```

In [25]:
```
lpp.objective.value()
```

Out[25]:
```
0.0
```

In [26]:
```
x.value()
```

Out[26]:
```
0.0
```

In [27]:
```
y.value()
```

Out[27]:
```
0.0
```

Q4) Write a Python program to solve the following LPP: Min Z = 3.5x + 2y subject to x + y ≥ 5 x ≥ 4 y ≤ 2 x ≥ 0, y ≥ 0.

In [28]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMinimize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
lpp+=(x+y>=5)
lpp+=(x>=4)
lpp+=(y<=2)
lpp+=(3.5*x+2*y)
lpp
```

Out[28]:
```
LPP:
MINIMIZE
3.5*x + 2*y + 0.0
SUBJECT TO
_C1: x + y >= 5

_C2: x >= 4

_C3: y <= 2

VARIABLES
x Continuous
y Continuous
```

In [29]:
```
lpp.solve()
```

Out[29]:
```
1
```

In [30]:
```
lpp.objective.value()
```

Out[30]:
```
16.0
```

In [31]:
```
x.value()
```

Out[31]:
```
4.0
```

In [32]:
```
y.value()
```

Out[32]:
```
1.0
```

Q5) Solve LPP by using python: Min Z = x + y subject to x ≥ 6 y ≥ 6 x + y ≥ 11 x ≥ 0, y ≥ 0

In [38]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMinimize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
lpp+=(x>=6)
lpp+=(y>=6)
lpp+=(x+y>=11)
lpp+=(x+y)
lpp
```

Out[38]:
```
LPP:
MINIMIZE
1*x + 1*y + 0
SUBJECT TO
_C1: x >= 6

_C2: y >= 6

_C3: x + y >= 11

VARIABLES
x Continuous
y Continuous
```

In [39]:
```
lpp.solve()
```

Out[39]:
```
1
```

In [40]:
```
lpp.objective.value()
```

Out[40]:
```
12.0
```

In [41]:
```
x.value()
```

Out[41]:
```
6.0
```

In [42]:
```
y.value()
```

Out[42]:
```
6.0
```

Q6) Write a Python program to solve the following LPP: Max Z = 4x + y + 3z + 5w subject to 4x + 6y−5z−4w ≥−20 −8x−3y + 3z + 2w ≤ 20 x ≥ 0,y ≥ 0.

In [44]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMaximize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
z=LpVariable("z",lowBound=0)
w=LpVariable("w",lowBound=0)
lpp+=(4*x+6*y-5*z-4*w>=(-20))
lpp+=((-8)*x-3*y+3*z+2*w<=20)
lpp+=(4*x+y+3*z+5*w)
lpp
```

Out[44]:
```
LPP:
MAXIMIZE
5*w + 4*x + 1*y + 3*z + 0
SUBJECT TO
_C1: - 4 w + 4 x + 6 y - 5 z >= -20

_C2: 2 w - 8 x - 3 y + 3 z <= 20

VARIABLES
w Continuous
x Continuous
y Continuous
z Continuous
```

In [45]:
```
lpp.solve()
```

Out[45]:
```
-2
```

In [46]:
```
lpp.objective.value()
```

Out[46]:
```
25.0
```

In [47]:
```
x.value()
```

Out[47]:
```
0.0
```

In [48]:
```
y.value()
```

Out[48]:
```
0.0
```