Assignment 1

Question 1

In [1]:
```
import pulp
from pulp import LpVariable, LpProblem, LpMaximize, LpStatus, value, LpMinimize

prob = LpProblem("problem", LpMinimize)

#variables
Aw1 = LpVariable('Aw1', 0, None)
Bw1 = LpVariable('Bw1', 0, None)
Cw1 = LpVariable('Cw1', 0, None)
Aw2 = LpVariable('Aw2', 0, None)
Bw2 = LpVariable('Bw2', 0, None)
Cw2 = LpVariable('Cw2', 0, None)

#obj
prob += 10 * Aw1 + 9 * Aw2 + 11 * Bw1 + 10 * Bw2 + 12 * Cw1 + 13 * Cw2

#prod constraints
prob += Aw1 + Aw2 >= 60
prob += Bw1 + Bw2 >= 80
prob += Cw1 + Cw2 >= 50

#time constraints
prob += 2 * Aw1 + 4 * Bw1 + 3 * Cw1 <= 350

prob += 9 * Aw2 + 4 * Bw2 + 7 * Cw2 <= 250

#solve
status = prob.solve()
print("Status:", prob.status)
print("Optimal value of Z (minimum):", prob.objective.value())
print("Optimal values:")
for v in prob.variables():
    print(v.name, "=", v.varValue)
```

```
Status: 1
Optimal value of Z (minimum): 2017.5
Optimal values:
Aw1 = 60.0
Aw2 = 0.0
Bw1 = 17.5
Bw2 = 62.5
Cw1 = 50.0
Cw2 = 0.0
```

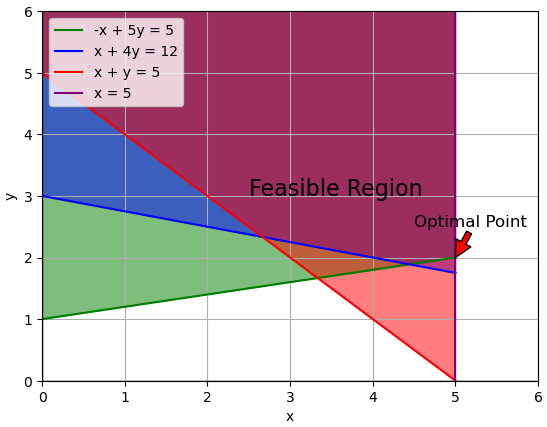
Question 2

In [2]:
```
import pulp
from pulp import LpVariable, LpProblem, LpMaximize, LpStatus, value, LpMinimize

prob = LpProblem("problem", LpMinimize)

#variables
x1 = LpVariable('x1', 0, 5)
x2 = LpVariable('x2', 0, None)

#obj
prob += -9 *x1 + 18 *x2

#constraints
prob += x1 - 5*x2 <= -5 #put negative values on right side
prob += x1 + 4*x2 >= 12
prob += x1 + x2 >= 5



#solve
status = prob.solve()
print("Status:", prob.status)
print("Optimal value of Z (minimum):", -prob.objective.value())
print("Optimal values:")
for v in prob.variables():
    print(v.name, "=", v.varValue)
```

```
Status: 1
Optimal value of Z (minimum): 9.0
Optimal values:
x1 = 5.0
x2 = 2.0
```

In [3]:
```
import numpy as np
import matplotlib.pyplot as plt


fig, ax = plt.subplots()


x = np.linspace(0, 5, 100)
y = (5 + x)/5
ax.plot(x, y, color='green', label='-x + 5y = 5')


ax.fill_between(x, y, 10, color='green', alpha=0.5)

y = (12 - x)/4
ax.plot(x, y, color='blue', label='x + 4y = 12')


ax.fill_between(x, y, 10, color='blue', alpha=0.5)

y = 5 - x
ax.plot(x, y, color='red', label='x + y = 5')


ax.fill_between(x, y, 10, color='red', alpha=0.5)
```

```
plt.axvline(x = 5, color = 'purple', label = 'x = 5')

#
ax.set_xlim(0, 6)
ax.set_ylim(0, 6)
ax.set_xlabel('x')
ax.set_ylabel('y')


plt.legend(loc='upper left')

#display
plt.grid(True)
plt.axhline(0, color='black', linewidth=1)
plt.axvline(0, color='black', linewidth=1)
plt.text(2.5, 3, 'Feasible Region', fontsize = 16)
plt.annotate('Optimal Point', xy = (5, 2),
             fontsize = 12, xytext = (4.5, 2.5),
             arrowprops = dict(facecolor = 'red'),
             color = 'black')
plt.show()
```



Question 4

In [3]:
```
#I fixed it!! Added parenthesis to the objective function.
import pulp
from pulp import LpVariable, LpProblem, LpMaximize, LpStatus, value, LpMinimize

prob = LpProblem("Maximize Profit", LpMaximize)

#variables
CB1 = LpVariable("CB1", 0, None)  # Columbian in R1
CB2 = LpVariable("CB2", 0, None)  # Columbian in R2
AB1 = LpVariable("AB1", 0, None)  # Arabian in R1
AB2 = LpVariable("AB2", 0, None)  # Arabian in R2

#obj
prob += (6 * (CB1 + AB1) + 5 * (CB2 + AB2)) - ((20/5 * (CB1 + CB2)) + (15/6 * (AB1+ AB2)))

#constraints
prob += CB1 >= .75 * (CB1 + AB1)
prob += CB2 >= .60 * (CB2 + AB2)
prob += CB1 + AB1 <= 40
prob += CB2 + AB2 <= 60

#solve
prob.solve()

# Print the results
print("Optimal Solution:")
print("Pounds of Roast 1 sold:", value(CB1 + AB1))
print("Pounds of Roast 2 sold:", value(CB2 + AB2))
print("Optimal values:")
for v in prob.variables():
    print(v.name, "=", v.varValue)
print("Maximized Profit: $", value(prob.objective))
```

```
Optimal Solution:
Pounds of Roast 1 sold: 40.0
Pounds of Roast 2 sold: 60.0
Optimal values:
AB1 = 10.0
AB2 = 24.0
CB1 = 30.0
CB2 = 36.0
Maximized Profit: $ 191.0
```

In [3]:
```
import pulp
from pulp import LpVariable, LpProblem, LpMaximize, LpStatus, value, LpMinimize


prob = LpProblem("problem", LpMaximize)

#variables
CB1 = LpVariable("CB1", 0, None)  #columbian beans in Roast 1
CB2 = LpVariable("CB2", 0, None)  #columbian beans in Roast 2
AB1 = LpVariable("AB1", 0, None)  #arabian beans in Roast 1
AB2 = LpVariable("AB2", 0, None)  #arabian beans in Roast 2

#costs
costCB = 20 / 5
costAB = 15 / 6

#revenue
```

```
revenue = 6 * (CB1 + AB1) + 5 * (CB2 + AB2)

# Cost part
total_cost = costCB * (CB1 + CB2) + costAB * (AB1 + AB2)

#obj function
prob += revenue - total_cost

#constraints
prob += CB1 >= 0.75 * (CB1 + AB1) #75 percent CB in R1
prob += CB2 >= 0.60 * (CB2 + AB2) #60 percent AB in R2
prob += CB1 + AB1 <= 40
prob += CB2 + AB2 <= 60

# Solve the problem
prob.solve()

# Print the results
print("Optimal Solution:")
print("Pounds of Roast 1 sold:", value(CB1 + AB1))
print("Pounds of Roast 2 sold:", value(CB2 + AB2))
print("Optimal values:")
for v in prob.variables():
    print(v.name, "=", v.varValue)
print("Maximized Profit: $", value(prob.objective))
```

```
Optimal Solution:
Pounds of Roast 1 sold: 40.0
Pounds of Roast 2 sold: 60.0
Optimal values:
AB1 = 10.0
AB2 = 24.0
CB1 = 30.0
CB2 = 36.0
Maximized Profit: $ 191.0
```

Question 5

In [6]:

```
import pulp
from pulp import LpVariable, LpProblem, LpMaximize, LpStatus, value, LpMinimize

prob = LpProblem("problem", LpMinimize)


x1 = LpVariable("x1", 0, None)  #12am-4am
x2 = LpVariable("x2", 0, None)  #4am-8am
x3 = LpVariable("x3", 0, None)  #8am-12pm
x4 = LpVariable("x4", 0, None) #12pm-4pm
x5 = LpVariable("x5", 0, None)  #4pm-8pm
x6 = LpVariable("x6", 0, None)  #8pm-12am

#obj
prob += x1 + x2 + x3 + x4 + x5 + x6

#constraints
prob += x6 + x1 >= 8
prob += x1 + x2 >= 7
prob += x2 + x3 >= 5
prob += x3 + x4 >= 4
prob += x4 + x5 >= 4
prob += x5 + x6 >= 7

#solve
status = prob.solve()
print("Status:", prob.status)
print("Optimal value of Z (minimum):", prob.objective.value())
print("Optimal values:")
for v in prob.variables():
    print(v.name, "=", v.varValue)
```

```
Status: 1
Optimal value of Z (minimum): 18.0
Optimal values:
x1 = 1.0
x2 = 6.0
x3 = 0.0
x4 = 4.0
x5 = 0.0
x6 = 7.0
```