```
import matplotlib.pyplot as plt
import numpy as np
from matplotlib import pyplot


current_x = 2
rate = 0.01 precision = 0.000001  algorithm
delta_x = 1
max_iterations = 10000
iteration_counter = 0


# dy/dx of eqn = 2*(x+3)
def slope(x):
    return 2*(x+3)

def value_y(x):
    return (x+3)**2
y = []
x = []
y.append(value_y(current_x))
x.append(current_x)



xi = np.linspace(-8, 2, 100)  # Adjust the range as needed

yi = (xi + 3)**2

plt.plot(xi, yi)

# Add labels and a title
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.title("Curve: y = (x + 3)^2")

# Display the plot
plt.grid(True)  # Add grid lines if desired
plt.show()


# In[5]:


while delta_x > precision and iteration_counter < max_iterations:
    previous_x = current_x
    current_x = previous_x - rate * slope(previous_x)
    y.append(value_y(current_x))
    x.append(current_x)
    delta_x = abs(previous_x - current_x)
    print(f"Iteration {iteration_counter+1}")
    iteration_counter += 1
    print(f"X = {current_x}")

print(f"Local Minima occurs at: {current_x}")
```

```
# In[13]:


pyplot.plot(x,y,'.-')
plt.xlabel('x-values')
plt.ylabel('y-values')
plt.title('y=(x+3)^2')
plt.show()
```