**CODE:**

prime = int(input("Enter the prime number: "))

primitive = int(input("Enter the primiitve root: "))

xa = int(input("Enter the value of Xa: "))

xb = int(input("Enter the value of Xb: "))

def deffie\_hellman(prime, primitive, xa, xb):

    ya = pow(primitive, xa, prime)

    yb = pow(primitive, xb, prime)

    priv\_a = pow(yb, xa, prime)

    priv\_b = pow(ya, xb, prime)

    return priv\_a, priv\_b, ya, yb

keys = deffie\_hellman(prime, primitive, xa, xb)

print(f"Public Key of A = {keys[2]}")

print(f"Public Key of B = {keys[3]}")

print(f"Private Key of A = {keys[0]}")

print(f"Private Key of B = {keys[1]}")

if keys[0] == keys[1]:

    print("This contains shared private keys")

else:

    print("This doesn't contain shared private keys")

**OUTPUT:**

