

"""

Name: Zainulabdin Bughio

Code name: weight\_converter.py

Description: Converts weight between kilograms and pounds based on user input.

"""

```
weight = float(input("enter weight: "))
unit = input(" in (K)g or (L)bs: ")
if unit.upper() == "L":
    weight_pounds = weight / 2.205
    print("weight: " + str(weight_pounds) + " KG")
elif unit.upper() == "K":
    weight_kilograms = weight * 2.205
    print("weight: " + str(weight_kilograms) + " Lbs")
```

"""

Name: Zainulabdin Bughio

Code name: basic\_calculator.py

Description: Performs basic arithmetic operations (+, -, \*, /) based on user input.

"""

```
num1 = float(input("enter your first number: "))
num2 = float(input("enter your second number: "))
operation = input(" enter the function you wish to use: ")
if operation.upper() == 'PLUS':
    Sum = num1 + num2
    print("SUM: " + str(Sum))
elif operation.upper() == 'MINUS' :
    diff = num1 - num2
    print("difference: " + str(diff))
elif operation.upper() == 'MULTIPLY':
    multiply = num1 * num2
    print("Answer:" + str(multiply))
elif operation.upper() == 'DIVIDE':
    divide = num1 / num2
```

```
print('divide: ' + str(divide))
```

```
"""
```

Name: Zainulabdin Bughio

Code name: even\_odd\_pattern.py

Description: Prints different star patterns depending on whether the input number is even or odd.

```
"""
```

```
num = float(input("enter the number")) # asks for the number
```

```
x = num % 2 # finds the remainder of the number when divided by 2
```

```
print(x) # just to check the remainder
```

```
if str(x) == "0.0": # makes the integer a string and checks if there is a  
remainder or not
```

```
    ast = ["*****", "      **", "*****"] # makes a list
```

```
    for y in ast: # assigns a variable to the values in the list and creates a  
loop where it goes through the list
```

```
        print(y) # prints every value in the list
```

```
else: # if the remainder is not 0 then it's odd and runs the following code
```

```
    ast = ["*", "**", "***", "****", "*****"] # makes this list
```

```
    for X in ast: # assigns variable again to every value
```

```
        print(X) # prints it
```