HOMEWORK 2

1.Design vacuum cleaner with any 4 different shapes of you choice ,Each code should atleast accept “start”, “stop”, “right”, “left” and “dock”. Rest is up to your imagination. Look each shape offers unique advantage ,now that’s something you need extract for your advantage and also justify why a particular shape is better

INPUT

```
# Vacuum cleaner program with 4 shapes

def vacuum(shape, cmd):
    print("\n", shape, "Vacuum Cleaner")

    # actions
    if cmd == "start":
        print(shape, "vacuum started cleaning.")
    elif cmd == "stop":
        print(shape, "vacuum stopped.")
    elif cmd == "left":
        print(shape, "vacuum turned left.")
    elif cmd == "right":
        print(shape, "vacuum turned right.")
    elif cmd == "dock":
        print(shape, "vacuum returned to dock.")
    else:
        print("Invalid command!")

    # advantages of shapes
    if shape.lower() == "circle":
        print("Advantage: Rotates smoothly and avoids stuck.")
    elif shape.lower() == "square":
        print("Advantage: Cleans edges and corners well.")
    elif shape.lower() == "triangle":
        print("Advantage: Can reach into sharp corners.")
    elif shape.lower() == "hexagon":
        print("Advantage: Covers more area with balance.")
    else:
        print("Choose only Circle, Square, Triangle or Hexagon")

# main program
print("Shapes available: Circle, Square, Triangle, Hexagon")
shape = input("Enter shape: ")
cmd = input("Enter command (start/stop/left/right/dock): ")
vacuum(shape, cmd)
```

OUTPUT

```
File  Edit  Shell  Debug  Options  Window  Help
Python 3.13.4 (tags/v3.13.4:8a526ec, Jun  3 2025, 17:46:04) [MSC v.1943 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
==== RESTART: C:/Users/varsh/AppData/Local/Programs/Python/Python313/dghf.py ===
Shapes available: Circle, Square, Triangle, Hexagon
Enter shape: circle
Enter command (start/stop/left/right/dock): start

 circle Vacuum Cleaner
circle vacuum started cleaning.
Advantage: Rotates smoothly and avoids stuck.
>>>
= RESTART: C:/Users/varsh/AppData/Local/Programs/Python/Python313/dghf.py
Shapes available: Circle, Square, Triangle, Hexagon
Enter shape: square
Enter command (start/stop/left/right/dock): left

 square Vacuum Cleaner
square vacuum turned left.
Advantage: Cleans edges and corners well.
>>>
= RESTART: C:/Users/varsh/AppData/Local/Programs/Python/Python313/dghf.py
Shapes available: Circle, Square, Triangle, Hexagon
Enter shape: trianngle
Enter command (start/stop/left/right/dock): right

 trianngle Vacuum Cleaner
trianngle vacuum turned right.
Choose only Circle, Square, Triangle or Hexagon
>>>
= RESTART: C:/Users/varsh/AppData/Local/Programs/Python/Python313/dghf.py
Shapes available: Circle, Square, Triangle, Hexagon
Enter shape: hexagon
Enter command (start/stop/left/right/dock): dock

 hexagon Vacuum Cleaner
hexagon vacuum returned to dock.
Advantage: Covers more area with balance.
>>>
```

CODE

```
# Vacuum cleaner program with 4 shapes
def vacuum(shape, cmd):
    print("\n---", shape, "Vacuum Cleaner ---")
 # actions
    if cmd == "start":
        print(shape, "vacuum started cleaning.")
    elif cmd == "stop":
        print(shape, "vacuum stopped.")
    elif cmd == "left":
        print(shape, "vacuum turned left.")
    elif cmd == "right":
        print(shape, "vacuum turned right.")
    elif cmd == "dock":
        print(shape, "vacuum returned to dock.")
```

```
    else:
        print("Invalid command!")
# advantages of shapes
    if shape.lower() == "circle":
        print("Advantage: Rotates smoothly and avoids stuck.")
    elif shape.lower() == "square":
        print("Advantage: Cleans edges and corners well.")
    elif shape.lower() == "triangle":
        print("Advantage: Can reach into sharp corners.")
    elif shape.lower() == "hexagon":
        print("Advantage: Covers more area with balance.")
    else:
        print("Choose only Circle, Square, Triangle or Hexagon")
# main program
print("Shapes available: Circle, Square, Triangle, Hexagon")
shape = input("Enter shape: ")
cmd = input("Enter command (start/stop/left/right/dock): ")
vacuum(shape, cmd)
```