2347265-lab-exercise-2

August 7, 2023

VISHNU SWAROOP P S - 2347265 - MCA B LAB EXERCISE - 2

[1]:
```
#1. Create a LIST with your domain attributes, insert the elements using the␣ ↪append (),
insert(), extend() and add any iterables (tuples, sets,␣ ↪dictionaries etc.) to the list (Use all
the methods ).
```

```
pharmacy_attributes = []

pharmacy_attributes.append("Inventory Management")
pharmacy_attributes.append("Prescription Processing")
print(pharmacy_attributes)


pharmacy_attributes.insert(2, "Drug Interactions Checking")
pharmacy_attributes.insert(4, "Pharmacist Dashboard")
print(pharmacy_attributes)

tuple = ("Pharmacy Analytics", "Barcode Scanning", "e-Prescriptions") print(tuple)
pharmacy_attributes.extend(tuple)
print(pharmacy_attributes)

set = {"Drug Pricing", "Pharmacy Billing"}
print(set)
pharmacy_attributes.extend(set)
print(pharmacy_attributes)


dict = {"Robotic Dispensing": True, "Electronic Health Records": True} print(dict)
pharmacy_attributes.extend(dict.keys())

print(pharmacy_attributes)
```

```
['Inventory Management', 'Prescription Processing']
['Inventory Management', 'Prescription Processing', 'Drug Interactions Checking',
'Pharmacist Dashboard']
```



```
('Pharmacy Analytics', 'Barcode Scanning', 'e-Prescriptions') ['Inventory Management',
'Prescription Processing', 'Drug Interactions Checking', 'Pharmacist Dashboard',
'Pharmacy Analytics', 'Barcode Scanning', 'e-Prescriptions']
{'Pharmacy Billing', 'Drug Pricing'}
```

['Inventory Management', 'Prescription Processing', 'Drug Interactions Checking', 'Pharmacist Dashboard', 'Pharmacy Analytics', 'Barcode Scanning', 'e-Prescriptions', 'Pharmacy Billing', 'Drug Pricing']
{'Robotic Dispensing': True, 'Electronic Health Records': True} ['Inventory Management', 'Prescription Processing', 'Drug Interactions Checking', 'Pharmacist Dashboard', 'Pharmacy Analytics', 'Barcode Scanning', 'e-Prescriptions', 'Pharmacy Billing', 'Drug Pricing', 'Robotic Dispensing', 'Electronic Health Records']

[6]:
```
#2 Create a list with numeric and perform the following operations. # a. Write a program to swap the first and last elements in a list. # b. Write a program to find the sum of the digits in a list. # c. Write a program to find the smallest element in a list.

num_list = [10, 3, 75, 28, 23, 45]
num_list[0], num_list[-1] = num_list[-1], num_list[0]
print("After swapping:", num_list)
added = sum(num_list)
print("Total sum:", added)
smallest = min(num_list)
print("Smallest :", smallest)
```

```
After swapping: [45, 3, 75, 28, 23, 10]
Total sum: 184
Smallest : 3
```

[1]:
```
# ii) Dictionaries
# a. Sort the dictionaries in ascending order based on the Key of the dictionary.


pharmacy_inventory = {'Aspirin': 100,'Paracetamol': 50,'Cough Syrup': 75,'Antibiotics': 30,'Vitamin C': 60}

sorted_inventory = dict(sorted(pharmacy_inventory.items()))

print("Pharmacy Inventory (Sorted by Key):")
for item, quantity in sorted_inventory.items():
    print(f"{item}: {quantity}")
```

```
Pharmacy Inventory (Sorted by Key):
Antibiotics: 30
Aspirin: 100
```



```
Cough Syrup: 75
Paracetamol: 50
Vitamin C: 60
```

[7]:
```
# b. Create the dictionary with Numeric as Value in Key – Value pair and find the sum of all the values in the Dictionary.
```

```
pharmacy_dict = {
        'paracetamol': 100,
        'aspirin': 75,
        'ibuprofen': 50,
        'antacid': 121,
        'antibiotic': 200
}

total_stock = sum(pharmacy_dict.values())

print("Pharmacy Dictionary:")
for key, value in pharmacy_dict.items():
        print(key, ":", value)

print("Sum of all values in the dictionary:", total_stock)
```

```
Pharmacy Dictionary:
paracetamol : 100
aspirin : 75
ibuprofen : 50
antacid : 121
antibiotic : 200
Sum of all values in the dictionary: 546
```

[8]:
```
# c. Write a Python code to demonstrate the sorting in descending order of␣
↪values with
  lambda function.
pharmacy_dict = {
        'paracetamol': 100,
        'aspirin': 75,
        'nocold': 50,
        'antacid': 120,
        'antibiotic': 200
}

sorted_pharmacy_dict = dict(sorted(pharmacy_dict.items(), key=lambda item:␣
↪item[1], reverse=True))

print("Sorted pharmacy dictionary in descending order of stock quantities:") for medicine,
quantity in sorted_pharmacy_dict.items():
        print(medicine, ":", quantity)
```



```
Sorted pharmacy dictionary in descending order of stock quantities: antibiotic :
200
antacid : 120
paracetamol : 100
```

aspirin : 75
nocold : 50