

data structures

lists

```
a=[1,2,3,4,5,6,7,8,9,10]
print(a)

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

b=[1,2.5,'pavan']
print(b)

[1, 2.5, 'pavan']
```

list operations *accessing items*

```
print(a[0]) #output:1
print(a[-1]) #output:10

1
10
```

modifying items

```
numbers = a # Assuming you want 'numbers' to refer to the list 'a'
numbers[0] = 10
print(numbers) #output:[10,2,3,4,5]

[10, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

adding items

insert

```
numbers.insert(1,15)
print(numbers) # Changed 'prin' to 'print' and passed 'numbers' as
argument

[10, 15, 15, 15, 15, 15, 15, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

append

```
numbers.append(11) # Changed 'number' to 'numbers'
print(numbers)

[10, 15, 15, 15, 15, 15, 15, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

removing items

remove()

```
numbers.remove(10) # Changed 'number' to 'numbers' to refer to the
defined list
print(numbers)
```

```
[15, 15, 15, 15, 15, 15, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

pop()

```
numbers.pop(2)
print(numbers)
```

```
[15, 15, 15, 15, 15, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

iterating through a list

```
for num in numbers:
    print(num)
```

```
15
15
15
15
15
2
3
4
5
6
7
8
9
10
11
```

tuples

creating a tuple

```
for num in numbers:
    print(num) # Corrected indentation (4 spaces)
```

```
15
15
15
15
15
2
3
4
```

```
5
6
7
8
9
10
11
```

dictionary

creating a dictionary

```
student={
    "name": "pavan",
    "age": 18,
    "marks": 85
}
print(student["name"])
pavan
```

modifying

```
student["age"] = 18
sets
```

creating a set

```
number={1,2,3,4,5}
```

adding items

```
if 6 in numbers:
    numbers.remove(6)
print(number)
{1, 2, 3, 4, 5}
```

removing items

```
# Check if the element is in the list before attempting to remove it.
if 3 in numbers:
    numbers.remove(3)
```

```
else:
    print("3 is not in the list")
3 is not in the list
```

set operations

```
manipulating list_
fruits=["apple","banana","cherry"]
fruits.append("orange")
fruits.remove("banana")
print(fruits)

['apple', 'cherry', 'orange']
```

creating a dictionary

```
book={
    "title":"python basics",
    "author":"pavan",
    "year":2021
}
print(book["title"])
book["year"]=2022
print(book)

python basics
{'title': 'python basics', 'author': 'pavan', 'year': 2022}
```

merge two lists

```
list=[1,2,3]
list2=[4,5,6]
merged_list=list+list2
print("merged list:",merged_list)

merged list: [1, 2, 3, 4, 5, 6]
```

dictionary operations

```
student={"name":"pavan","age":18,"marks":85}
print("name:",student["name"])
student["marks"]=90
print("updated marks:",student["marks"])

name: pavan
updated marks: 90
```

find the maximum and minimum in a list

```

number=[10,20,30,40,50]
print("maximum:",max(numbers))
print("manimum:",min(numbers))

```

```

maximum: 15
manimum: 2

```

count frequency of elements in a list

```

num=[1,2,2,3,3,3,4,4,4,4]
frequency={}
for num in numbers:
    if num in frequency:
        frequency[num]+=1
    else:
        frequency[num]=1
print("frequency:",frequency)

frequency: {15: 5, 2: 1, 4: 1, 5: 1, 7: 1, 8: 1, 9: 1, 10: 1, 11: 1}

```

sort a list of tuples by the second element

```

tuples=[(1,'apple'),(2,'banana'),(3,'cherry')]
sorted_tuples=sorted(tuples,key=lambda x:x[1])
print("sorted tuples:",sorted_tuples)

sorted tuples: [(1, 'apple'), (2, 'banana'), (3, 'cherry')]

```

car class with methods

```

class car:
    def __init__(self, brand, model): # Corrected __inti__ to __init__
        self.brand = brand
        self.model = model

    def display_info(self): # Corrected display__info to display_info
        print(f"brand:{self.brand},model:{self.model}")

# Create an instance of the car class
my_car = car("toyota", "camry")
my_car.display_info()

brand:toyota,model:camry


number=int(input("enter a number"))
reverse_number=0

```

```
temp=number
while temp>0:
    digit=temp%10
    reverse_number=reverse_number*10+digit
    temp=temp//10
```

```
if number==reverse_number:
    print(f"{number} palli")
else:
    print(f"{number} not pali")
```

```
enter a number121
121 palli
```

```
number=input("enter a number:")
if number == number[::-1]:
    print("palindrome")
else:
    print("not palindrome")
```

```
enter a number:sos
palindrome
```