1. SUM OF DIGITS

Program

```
n=int(input("Enter any number :"))
tot=0
while(n>0):
    dig=n%10
    tot=tot+dig
    n=n//10
print("Total sum of digits is",tot)
```

Output

```
Enter any number: 457
Total sum of digits is 16
```

2. REVERSE NUMBER

Program

```
number=int(input("Enter any number"))
reverse=0
while(number>0):
    remainder=number%10
    reverse=(reverse*10)+remainder
    number=number/10
print("Reverse of the number is=%d"%reverse)
```

Output

```
Enter any number 567
Reverse of the number is = 765
```

3. LARGEST NUMBER

Program

```
print("Enter three numbers")
n1=input()
n2=input()
n3=input()
if(n1>=n2)and(n1>=n3):
  largest=n1
elif(n2>n1)and(n2>=n3):
  largest=n2
else:
   largest=n3
print("The largest number is",largest)
```

Output

```
Enter three numbers
56
89
77
The largest number is 89
```

4. MATRIX MULTIPLICATION

Program

```
x=[[12,7,3],[4,5,6],[7,8,9]]
y=[[5,8,1,2],[6,7,3,0],[4,5,9,1]]
result=[[0,0,0,0],[0,0,0,0],[0,0,0,0]]
for i in range(len(x)):
    for j in range(len(y[0])):
        for k in range(len(y)):
            result[i][j]+=x[i][k]*y[k][j]
    for r in result:
            print(r)
```

Output

```
[114, 160, 60, 27]
[0, 0, 0, 0]
[0, 0, 0, 0]
[114, 160, 60, 27]
[74, 97, 73, 14]
[0, 0, 0, 0]
[114, 160, 60, 27]
[74, 97, 73, 14]
[119, 157, 112, 23]
```

5. USE OF LIST IN PYTHON

Program

```
#correcting mistake values in a list
odd=[2,4,6,8]
#change the 1st item
odd[0]=1
print(odd)
#change 2nd to 4th items
odd[1:4]=[3,4,7]
print(odd)
#update list
odd[3]=7.5
print(odd)
#delete list
del odd[2]
print("After deleting value at index 1:",odd)
```

Output

```
[1, 4, 6, 8]
[1, 3, 4, 7]
[1, 3, 4, 7.5]
After deleting value at index 1: [1, 3, 7.5]
```

6. FIBONACCI SERIES

Program

```
def fibonacci(n):
  if n==0:
     return 0
  elif n==1:
     return 1
  else:
     return fibonacci(n-1)+fibonacci(n-2)
print(fibonacci(6))
```

\\

Output

8

7. STRING PROCESSING

Program

```
#String Slicing
greet = 'Hello'
# access 1st index element
print(greet[1])
greet = 'Hello'
# access character from 1st index to 3rd index
print(greet[1:4])
#String Compare
str1 = "Python Programming"
str2 = "Information Technology"
str3 = "Python Programming"
# compare str1 and str2
print(str1 == str2)
# compare str1 and str3
print(str1 == str3)
#String concatenate
greet = "Good"
name = "Morning!"
# using + operator
result = greet + name
print(result)
greet = 'Python'
```

```
# iterating through greet string
for letter in greet:
    print(letter)
greet = 'Python'
# count length of greet string
print(len(greet))
#Membership Operator
print('a' in 'program')
print('at' not in 'battle')
```

Output

```
e
ell
False
True
GoodMorning!
P
y
t
h
o
n
6
True
False
```

8. DICTIONARIES AND SETS

Program

```
# Create a dictionary
my_dict = {
    'name': 'Alice',
    'age': 30,
    'city': 'New York'
}
# Print the original dictionary
print("Original dictionary:", my_dict)
# Add a new key-value pair
my_dict['job'] = 'Engineer'
print("After adding job:", my_dict)
# Update an existing key-value pair
my_dict['age'] = 31
print("After updating age:", my_dict)
# Remove a key-value pair
del my_dict['city']
print("After removing city:", my_dict)
# Check if a key exists
print("Is 'name' in dictionary?", 'name' in my_dict)
print("Is 'city' in dictionary?", 'city' in my_dict)
# Set operations
print("\nSet Operations:")
```

```
# Create a set
my_set = {1, 2, 3, 4, 5}
# Print the original set
print("Original set:", my_set)
# Add an element to the set
my_set.add(6)
print("After adding 6:", my_set)
# Remove an element from the set
my_set.discard(3)  # discard does not raise an error if the element is not present
print("After removing 3:", my_set)
# Check if an element is in the set
print("Is 4 in the set?", 4 in my_set)
print("Is 10 in the set?", 10 in my_set)
# Union and intersection of sets
another_set = {4, 5, 6, 7, 8}
print("Another set:", another_set)
union_set = my_set | another_set
print("Union of sets:", union_set)
intersection_set = my_set & another_set
print("Intersection of sets:", intersection_set)
```

Output

Original dictionary: {'name': 'Alice', 'age': 30, 'city': 'New York'}

After adding job: {'name': 'Alice', 'age': 30, 'city': 'New York', 'job': 'Engineer'}

After updating age: {'name': 'Alice', 'age': 31, 'city': 'New York', 'job': 'Engineer'}

After removing city: {'name': 'Alice', 'age': 31, 'job': 'Engineer'}

Is 'name' in dictionary? True

Is 'city' in dictionary? False

Set Operations

Original set: {1, 2, 3, 4, 5}

After adding 6: {1, 2, 3, 4, 5, 6}

After removing 3: {1, 2, 4, 5, 6}

Is 4 in the set? True

Is 10 in the set? False

Another set: {4, 5, 6, 7, 8}

Union of sets: {1, 2, 4, 5, 6, 7, 8}

Intersection of sets: {4, 5, 6}

9. CLASSES AND OBJECTS

Program

```
class Person:
  def __init__(self, name, age):
    self.name = name
    self.age = age
p1 = Person("John", '36');
print(p1.name)
print(p1.age)
```

Output

```
John
36
```

10. POLYMORPHISM

Program

```
class Car:
  def __init__(self, brand, model):
    self.brand = brand
    self.model = model
  def move(self):
    print("Drive!")
class Boat:
  def __init__(self, brand, model):
    self.brand = brand
    self.model = model
  def move(self):
    print("Sail!")
class Plane:
  def __init__(self, brand, model):
    self.brand = brand
    self.model = model
  def move(self):
    print("Fly!")
Car1=Car("Ford", "Mustang")       #Create a Car class
boat1=Boat("Ibiza", "Touring 20") #Create a Boat class
plane1=Plane("Boeing", "747")     #Create a Plane class
```

```
for x in (Car1, boat1, plane1):
  x.move()
\
\
```

Output

```
Drive!
Sail!
Fly!
```

11. INHERITANCE

Program

```
class Vehicle:
  def __init__(self, brand, model):
    self.brand = brand
    self.model = model
  def move(self):
    print("Move!")
class Car(Vehicle):
  pass
class Boat(Vehicle):
  def move(self):
    print("Sail!")
class Plane(Vehicle):
  def move(self):
    print("Fly!")
Car1 = Car("Ford", "Mustang") #Create a Car object
boat1 = Boat("Ibiza", "Touring 20") #Create a Boat object
plane1 = Plane("Boeing", "747") #Create a Plane object
for x in (Car1, boat1, plane1):
  print(x.brand)
  print(x.model)
  x.move()
```

Output

Ford

Mustang

Move!

Ibiza

Touring 20

Sail!

Boeing

747

Fly!