In [13]:

```
def uppercase(str_data):
    result = ''
    for char in str_data:
        if ord(char) >= 65:
            result += chr(ord(char)-32)
        else:
            result += chr(ord(char))
    return result
print(uppercase('@dhd'))
```

```
@DHD
```

In [11]:

```
def lowercase(str_data):
    result = ''
    for char in str_data:
        if ord(char) >= 65:
            result += chr(ord(char)+32)
        else:
            result += chr(ord(char))
    return result
print(lowercase('@DHD'))
```

```
@dhd
```

In [38]:

```
def ifnum(str_data):

    for char in str_data:
        if ord(char) > 47 and ord(char) < 58 :
            return True
        else:
            return False
    pass
print(ifnum('9435'))
```

```
True
```

In [14]:

```
def ifalnum(str_data):

    for char in str_data:
        if ord(char) > 47 and ord(char) < 58 or ord(char) > 64 and ord(char) < 91 or ord(char) > 96 and ord(char) < 123:
            return True
        else:
            return False
    pass
print(ifalnum('gj01mp9435'))
```

```
True
```

In [19]:

```
def ifupper(str_data):

    for char in str_data:
        if ord(char) > 64 and ord(char) < 91 :
            return True
        else:
            return False
    pass
print(ifupper('HIGH'))
```

```
True
```

In [20]:

```
def iflower(str_data):

    for char in str_data:
        if ord(char) > 96 and ord(char) < 123 :
            return True
        else:
            return False
    pass
print(iflower('low'))
```

```
True
```

In [17]:

```
def ifalpha(str_data):

    for char in str_data:
        if ord(char) > 64 and ord(char) < 91 or ord(char) > 96 and ord(char) < 123:
            return True
        else:
            return False
    pass
print(ifalpha('Gj'))
```

```
True
```

In [19]:

```
def cap(x):
    a=""
    for i in x[0]:
        if ord(i) in range(ord("a"),ord("z")+1):
            a=a+chr(ord(i)-32)
        else:
            a+=chr(ord(i))
    for i in x[1:]:
        if ord(i) in range(ord("A"),ord("Z")+1):
            a+=chr(ord(i)+32)
        else:
            a+=chr(ord(i))
    return a
print(cap('rng8'))
```

```
Rng8
```

In [3]:

```
def swapin(x):
    y=""
    for i in x:
        if ord(i) in range(ord("a"),ord("z")+1):
            y+=chr(ord(i)-32)
        elif ord(i) in range(ord("A"),ord("Z")+1):
            y+=chr(ord(i)+32)
        else:
            y+=chr(ord(i))
    return y
print(swapin("@DrenaLIne"))
```

```
@dRENAliNE
```

In [36]:

```
import rng8
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

```

```

In [1]:

```
import rng8
```

In [6]:

```
rng8.ifnum("90")
```

Out[6]:

```
True
```

In [7]:

```
rng8.ifalnum("90aj")
```

Out[7]:

```
True
```

In [8]:

```
rng8.ifnum("gfd")
```

Out[8]:

```
False
```

In [9]:

```
rng8.ifupper("MNGF")
```

Out[9]:

```
True
```

In [10]:

```
rng8.ifupper("cvfg")
```

Out[10]:

```
False
```

In [11]:

```
rng8.iflower("cbvdg")
```

Out[11]:

```
True
```

In [12]:

```
rng8.iflower("LODP")
```

Out[12]:

```
False
```

In [14]:

```
rng8.ifalpha("AMBclop")
```

Out[14]:

```
True
```

In [21]:

```
rng8.cap("gvbdgf")
```

Out[21]:

```
'Gvbdgf'
```

In [16]:

```
rng8.swapin("MnOp")
```

Out[16]:

```
'mNoP'
```

In [20]:

```
rng8.uppercase("mcbdvgd")
```

Out[20]:

```
'MCBDVGD'
```

In [18]:

```
rng8.lowercase("VCYRPLK")
```

Out[18]:

```
'vcyrplk'
```

In [ ]:

```

```