அலகு III

பாடம் 10

பைத்தான் இனக்குழுக்கள் மற்றும் பொருள்கள்

4LSLLU

கற்றலின் நோக்கங்கள்

இந்த பாடப்பகுதியைக் கற்றபின், மாணவர்கள்

- பொருள் நோக்கு நிரலாக்கத்தின் அடிப்படை கருத்துருக்களாகிய இனக்குழு, பொருள் ஆக்கி மற்றும் அழிப்பி ஆகியவற்றைப் புரிந்து கொள்ளுதல்.
- பைத்தானில் இனக்குழுக்களையும், பொருள்களையும் உருவாக்குவதைப் பற்றிய அறிவைப் பெறுதல்.
- ஆக்கிகளுடன் கூடிய இனக்குழுக்களை உருவாக்குதல்.
- இனக்குழுவைப் பயன்படுத்தி **பைத்தானில்** சிக்கலான நிரல்களை எழுதுதல்.

10.1 அறிமுகம்

பைத்தான் ஒரு பொருள் நோக்கு நிரலாக்க மொழி (Object Oriented Programming Language) ஆகும். இனக்குழுக்களும் (Class), பொருள்களும் (Object) பொருள் நோக்கு நிரலாக்கத்தின் முக்கிய அம்சமாகும். இதன் இனக்குழு மற்றும் பொருள்களின் கோட்பாடுகள், C++ கோட்பாடுகளை ஒத்திருக்கும். ஆனால் C++ உடன் ஒப்பிடும் போது, பைத்தானில் இனக்குழு மற்றும் பொருள்களை உருவாக்குவதும், செயல்படுத்துவதும் மிக எளிது.

பைத்தானில் மிக முக்கிய கட்டமைப்பு கூறுகளாகத் திகழ்பவை இனக்குழுவாகும். பொருள் என்பது தரவுகளின் மீது செயல்படும் செயற்கூறு மற்றும் தரவுகளின் தொகுப்பாகும். இனக்குழு என்பது பொருளின் வார்ப்புரு (template) ஆகும்.

பொருள்நோக்கு நிரலாக்கத்தின்படி பொருள்கள் இனக்குழுவின் சான்றுரு (instance) (அல்லது) இனக்குழு மாறி (class variables) என்று அழைக்கப்படும். பைத்தானில் அனைத்துமே பொருள்களாகும். எடுத்துக்காட்டாக, நிரலில் பயன்படும் அனைத்து முழு எண் மாறிகளும் “int” இனக்குழுவின் பொருளாகும். அதே போல அனைத்து சரங்களின் மாறிகளும் string இனக்குழுவின் பொருளாகும்.

10.2 இனக்குழுவை வரையறுத்தல்

பைத்தானில், இனக்குழுவை வரையறுக்க “class” என்னும் சிறப்புச் சொல் பயன்படுத்தப்படுகிறது. ஒவ்வொரு இனக்குழுவும் தனித்த பெயருடன் முக்காற்புள்ளி (:) யைக் கொண்டு முடியும்.

Chapter 10.indd 174 30-01-2020 19:40:01

தொடரியல்

```
class class_name:
    statement_1
    statement_2
    …………..
    …………..
    statement_n
```

இனக்குழு வரையறுப்பில், கூற்று (statement) என்பது மாறி அறிவிப்பாகவோ, தேர்ந்தெடுப்பு கூற்றாகவோ, மடக்காகவோ அல்லது செயற்கூறு வரையறையாகவோ இருக்கலாம். இனக்குழுக்குள்ளே வரையறுக்கப்படும் மாறிகள் இனக்குழு மாறிகள் (Class variables) என்றும் செயற்கூறுகள் வழிமுறைகள் (methods) என்றும் அழைக்கப்படும். இனக்குழு மாறிகள் மற்றும் வழிமுறைகள் சேர்ந்து இனக்குழுவின் உறுப்புகள் (members) எனப்படும். இனக்குழுவின் உறுப்புகளை இனக்குழுவின் பொருள்கள் அல்லது சான்றுருக்கள் மூலமாகவே அணுகுதல் வேண்டும். ஒரு இனக்குழுவை பைத்தான் நிரலில் எங்கு வேண்டுமானாலும் வரையறுக்கலாம்.

எடுத்துக்காட்டு: இனக்குழுவை வரையறுக்கும் நிரல்

```
class Sample:
        x, y = 10, 20          # class variables
```

மேலே குறிப்பிட்டுள்ள இனக்குழு வரையறையில், "sample" என்பது இனக்குழுவின் பெயர். இதில் x மற்றும் y என இரு மாறிகள் உள்ளன. இதன் தொடக்க மதிப்பு முறையே 10 மற்றும் 20 ஆகும். இனக்குழுவின் உள்ளே வரையறுக்கப்பட்டுள்ள மதிப்புகளை அணுகுவதற்கு இனக்குழுவின் பொருள் அல்லது சான்றுரு தேவை.

10.3 பொருள்களை உருவாக்குதல்

இனக்குழு உருவாக்கப்பட்ட பின் அந்த இனக்குழுவின் பொருள் அல்லது சான்றுரு உருவாக்கப்பட வேண்டும். பொருளை உருவாக்கும் இந்த முறைக்கு "சான்றுருவாக்கல்" (instantiation) என்று பெயர்.

தொடரியல்

```
Object_name = class_name( )
```

குறிப்பு- இனக்குழு சான்றுருவாக்கல், செயற்கூறு குறியீட்டைப் பயன்படுத்துகிறது. அதாவது, இனக்குழு பெயரானது அடைப்புக்குறியைக் கொண்டிருக்கும்.

WTPD8W

10.4 இனக்குழு உறுப்புகளை அணுகுதல்

இனக்குழு உறுப்புகளை (அதாவது இனக்குழு மாறி அல்லது வழிமுறை (செயற்கூறு)) புள்ளி செயற்குறி மூலம் அணுக முடியும்.

தொடரியல்

```
Object_name . class_member
```

எடுத்துக்காட்டு 10.1 : இனக்குழுவை வரையறுத்து அதன் உறுப்பு மாறிகளை அணுகும் நிரல்

```
class Sample:
#class variables
        x, y = 10, 20
S=Sample( )             # class instantiation
print("Value of x = ", S.x)
print("Value of y = ", S.y)
print("Value of x and y = ", S.x+S.y)
```

வெளியீடு

```
Value of x =  10
Value of y =  20
Value of x and y =  30
```

மேலே உள்ள நிரலில் இனக்குழுவின் பெயர் sample இனக்குழுவிற்குள்ளே, x மற்றும் y மாறிகள் முறையே 10 மற்றும் 20 என்ற தொடக்க மதிப்புகளை இருத்துகிறது. இந்த இரண்டு மாறிகளும் இனக்குழு மாறிகள் அல்லது இனக்குழுவின் உறுப்பு மாறிகள் (member variable) என்றும் அழைக்கப்படுகிறது. இந்த இனக்குழு சான்றுருவாக்கல் முறையில் இனக்குழுவின் உறுப்புகளை அணுகுவதற்கு S என்ற பொருள் உருவாக்கப்பட்டுள்ளது. முதல் இரண்டு print கூற்றுகள் இனக்குழுவின் மாறிகளான x மற்றும் y ன் மதிப்புகளை அச்சிடுகிறது. கடைசி print கூற்று இரண்டு இனக்குழு மாறிகளான x மற்றும் y மதிப்புகளின் கூட்டுத் தொகையை அச்சிடுகிறது.

10.5 இனக்குழு வழிமுறைகள்

பைத்தானில் இனக்குழு செயற்கூறு அல்லது வழிமுறை சாதாரண செயற்கூறுவைப்போல் ஒத்திருந்தாலும், ஒரு சிறிய வேறுபாடு உள்ளது. இனக்குழு வழிமுறையின் முதல் அளபுரு கண்டிப்பாக self என இருக்க வேண்டும். வழிமுறைகளை அழைக்கும் போது இந்த அளபுருவுக்கு மதிப்பை அனுப்ப தேவையில்லை. பைத்தான் தானாகவே இதற்கு மதிப்பை வழங்கும். வழிமுறையில் அளபுருக்களை எடுத்துக் கொள்ளாத போதும், முதல் அளபுருவாக self வரையறுக்கப்பட வேண்டும். வழிமுறை ஒரே ஒரு அளபுருவை ஏற்க வரையறுக்கப்பட்டால் அது இரண்டு அளபுருவாக எடுத்துக் கொள்ளும். அதாவது, self மற்றும் வரையறுக்கப்பட்ட அளபுரு.

இனக்குழுவிற்குள் மாறிகளை அறிவிக்கும் பொழுது, வழிமுறைகள் முன்னொட்டாக இனக்குழுவின் பெயர் மற்றும் புள்ளி(.) செயற்குறியைக் கொண்டு இருக்க வேண்டும்.

குறிப்பு

இனக்குழுவின் உள்ளே வரையறுக்கப்படும் கூற்றுகள் முறையாக உள்தள்ளப்பட்டிருக்க வேண்டும்.

Chapter 10.indd 176   30-01-2020 19:40:01

எடுத்துக்காட்டு 10.2 : இனக்குழுவைப் பயன்படுத்தி மொத்தம் மற்றும் சராசரி மதிப்பெண்ணைக் கணக்கிடும் நிரல்.

```
class Student:
        mark1, mark2, mark3 = 45, 91, 71              #class variable
        def process(self):                             #class method
                sum = Student.mark1 + Student.mark2 + Student.mark3
                avg = sum/3
                print("Total Marks = ", sum)
                print("Average Marks = ", avg)
                return
S=Student()
S.process()
```

மேலே உள்ள நிரலில் இனக்குழு வரையறுக்கப்பட்டபின் S என்ற பொருள் உருவாக்கப்பட்டுள்ளது. S.process( ) என்ற கூற்று தேவையான வெளியீட்டைப் பெறுவதற்காக செயற்கூறுவை அழைக்கிறது.

mark1, mark2 மற்றும் mark3 ஆகிய மூன்று மாறிகளின் மதிப்பு முறையே 45, 91, 71 என அறிவிக்கப்பட்டுள்ளது. process() என்னும் வழிமுறை self அளபுருவுடன் வரையறுக்கப்பட்டுள்ளது. அதாவது, எந்த மதிப்பையும் அனுப்பப்போவதில்லை. process வழிமுறை முதலில் இனக்குழு மாறிகளின் மதிப்பை கூட்டுகிறது. Sum என்ற மாறியில் விடையைச் சேமிக்கிறது. சராசரியைக் கணக்கிட்டு காண்பிக்கிறது.

மேலே உள்ள நிரலின் படி, பின்வரும் வெளியீடு தோன்றும்.

வெளியீடு

```
Total Marks =  207
Average Marks =  69.0
```

எடுத்துக்காட்டு 10.3: இனக்குழுவைப் பயன்படுத்தி கொடுக்கப்பட்ட எண் ஒற்றைப்படை அல்லது இரட்டைப்படை எண்ணா என்று சோதித்து அச்சிடும் நிரல்

```
class Odd_Even:
        even = 0                        #class varaiable
def check(self, num):
        if num%2==0:
                print(num," is Even number")
        else:
                print(num," is Odd number")
n=Odd_Even()
x = int(input("Enter a value: "))
n.check(x)
```

இந்த நிரலை இயக்கும் போது பைத்தான், பயனர் கொடுக்கும் மதிப்பை உள்ளீடாகப் பெற்று, அதை வழிமுறைக்கு அனுப்பி, பொருள் மூலம் சரிபார்க்கிறது.

வெளியீடு 1

```
Enter a value: 4
4  is Even number
```

வெளியீடு 2

```
Enter a value: 5
5  is Odd number
```

10.6 பைத்தானில் ஆக்கிகள் மற்றும் அழிப்பிகள்

ஓர் இனக்குழுவின் சான்றுரு பயன்பாட்டிற்கு வரும்பொழுது ஆக்கி என்னும் சிறப்புச் செயற்கூறு தானாகவே இயக்கப்படுகிறது. பைத்தானில், "init" என்னும் சிறப்பு செயற்கூறு ஆக்கியாக செயல்படுகிறது. இது இரட்டை அடிக்கீறில் (Under Score) தொடங்கி இரட்டை அடிக்கீறலுடன் முடிய வேண்டும்.

இந்த செயற்கூறு சாதாரண செயற்கூறுவைப் போல செயல்படும். ஆனால் பொருள் உருவாக்கப்பட்டவுடன் தானாகவே இயக்கப்படும். இந்த ஆக்கி செயற்கூறை அளபுருக்களுடனோ அல்லது இல்லாமலோ வரையறுக்கலாம்.

__init__ **வழிமுறையின் *(ஆக்கி)* பொதுவடிவம்**

def __init__(self, [args ....]):

<statements>

எடுத்துக்காட்டு 10. 4: ஆக்கியை விளக்கும் நிரல்

```
class Sample:
        def __init__(self, num):
                print("Constructor of class Sample...")
                self.num=num
                print("The value is :", num)
S=Sample(10)
```

மேலே உள்ள இனக்குழு "Sample", "num" என்னும் ஒரே ஒரு அளபுருவுடன் கூடிய ஆக்கியைக் கொண்டுள்ளது. ஆக்கி இயக்கப்படும் போது முதல் "print" கூற்று, "Constructor of class Sample...." என்பதை அச்சிடும். பிறகு, ஆக்கிக்கு அனுப்பப்படும் மதிப்பு self.num ல் இருத்தப்படுகிறது. இறுதியாக, கொடுக்கப்பட்ட சரத்துடன் அனுப்பட்ட மதிப்பையும் அச்சிடுகிறது. S என்ற பொருள் 10 என்ற மெய்யான அளபுருவுடன் உருவாக்கப்படும் போது மேலே உள்ள ஆக்கி தானாகவே இயக்கப்படும். பின்வரும் வெளியீடு கிடைக்கிறது.

Constructor of class Sample...

The value is : 10

ஆக்கியின் உள்ளே வரையறுக்கப்படும் இனக்குழு மாறியானது, இனக்குழுவில் உருவாக்கப்படும் பொருள்களின் எண்ணிக்கையைக் கொண்டிருக்கும்:

எடுத்துக்காட்டு 10.5 : உருவாக்கப்பட்ட பொருள்களின் எண்ணிக்கையை கொண்டிருக்கும் இனக்குழு மாறியை விளக்கும் நிரல்

```
class Sample:
        num=0
        def __init__(self, var):
                Sample.num+=1
                self.var=var
                print("The object value is = ", var)
                print("The count of object created = ", Sample.num)

S1=Sample(15)
S2=Sample(35)
S3=Sample(45)
```

மேலே உள்ள நிரலில், num என்னும் இனக்குழு மாறியானது Sample என்னும் இனக்குழுவின் மூன்று பொருள்களுடன் பகிரப்படுகிறது. இதில் 0 என்ற தொடக்க மதிப்பு இருத்தப்பட்டு, ஒவ்வொரு முறையும் பொருள் உருவாக்கப்படும் போது num ன் மதிப்பு 1 அதிகரிக்கப்படுகிறது. அனைத்துப் பொருள்களும் மாறியை பகிர்வதால் ஒரு பொருளில் num மாறியில் செய்யப்படும் மாற்றமானது மற்ற பிற பொருள்களில் பிரதிபலிப்பதால், பின்வரும் வெளியீடு கிடைக்கும்.

வெளியீடு

```
The object value is            =    15
The count of object created    =    1
The object value is            =    35
The count of object created    =    2
The object value is            =    45
The count of object created    =    3
```

குறிப்பு: இனக்குழு மாறியானது *C++*ல் உள்ள *Static* வகையை ஒத்ததாகும்.

இனக்குழுவில் உருவாக்கப்பட்ட பொருளின் பயன்பாடு முடிவுக்கு வரும் போது அழிப்பி என்னும் சிறப்பு செயற்கூறு தானாகவே இயக்கப்படும். இது ஆக்கிக்கு முரணானது. பைத்தானில், __del__( ) செயற்கூறு அழிப்பியாகப் பயன்படுத்தப்படுகிறது.

Chapter 10.indd 179 30-01-2020 19:40:01

எடுத்துக்காட்டு 10.6 : __del__( ) வழிமுறையை விளக்கும் நிரல்

```
class Sample:
        num=0
        def __init__(self, var):
                Sample.num+=1
                self.var=var
                print("The object value is = ", var)
                print("The value of class variable is= ", Sample.num)
        def __del__(self):
                Sample.num-=1
                print("Object with value %d is exit from the scope"%self.var)
S1=Sample(15)
S2=Sample(35)
S3=Sample(45)
```

10.7 Public மற்றும் Private தரவு உறுப்புகள்

இனக்குழுவின் உள்ளே வரையறுக்கப்படும் மாறி கொடாநிலையாக public மாறி ஆகும். இந்த மாறிகளை புள்ளி செயற்குறி(.) பயன்படுத்தி நிரலில் எங்கு வேண்டுமானாலும் அணுகலாம். இரட்டை அடிக்கீறலை முன்னொட்டாக கொண்ட மாறிகள் private ஆகும். இந்த மாறிகளை இனக்குழுவின் உள்ளே மட்டும் தான் அணுக முடியும்.

எடுத்துக்காட்டு 10.7 : private மற்றும் public மாறிகளை விளக்கும் நிரல்

```
class Sample:
        def __init__(self, n1, n2):
                self.n1=n1
                self.__n2=n2
        def display(self):
                print("Class variable 1 = ", self.n1)
                print("Class variable 2 = ", self.__n2)
S=Sample(12, 14)
S.display()
print("Value 1 = ", S.n1)
print("Value 2 = ", S.__n2)
```

மேலே உள்ள நிரலில் n1 மற்றும் n2 என இரண்டு இனக்குழு மாறிகள் அறிவிக்கப்பட்டுள்ளது. n1 என்பது public மாறி n2 என்பது Private மாறி ஆகும். Display ( ) உறுப்பு செயற்கூறு இந்த இரண்டு மாறிகளுக்கும் அனுப்பப்பட்ட மதிப்புகளைக் காண்பிப்பதற்காக வரையறுக்கப்பட்டுள்ளது.

இனக்குழு மாறி n2 private ஆக இருந்தபோதிலும் இனக்குழுவின் உள்ளே உள்ள print கூற்றுகள் வெற்றிகரமாக n1 மற்றும் n2 மாறிகளின் மதிப்புகளைக் காண்பிக்கிறது. ஏனென்றால், இந்த நிரலில் இனக்குழுவின் உள்ளே வரையறுக்கப்பட்ட வழிமுறையின் மூலம் n2 அழைக்கப்பட்டுள்ளது. ஆனால், இனக்குழுவின் வெளியில் இருந்து n2 வின் மதிப்பை அணுக முயற்சித்தால் பிழை செய்தியைக் காண்பிக்கும். ஏனென்றால் private மாறிகள் இனக்குழுவின் வெளியில் இருந்து அணுக முடியாது.

Chapter 10.indd 180 30-01-2020 19:40:01

வெளியீடு

```
Class variable 1 =  12
Class variable 2 =  14
Value 1 =  12

Traceback (most recent call last):
        File "D:/Python/Class-Test-04.py", line 12, in <module>
                print("Value 2 = ", S.__n2)
AttributeError: 'Sample' object has no attribute '__n2'
```

10.8 இனக்குழு மற்றும் பொருள் ஆகியவற்றை விளக்கும் மாதிரி நிரல்கள்

நிரல் 1: வட்டத்தின் பரப்பையும், சுற்றளவையும் கணக்கிடும் பைத்தான் நிரலை எழுதுக.

```
class Circle:
        pi=3.14
        def __init__(self,radius):
                self.radius=radius
        def area(self):
                return Circle.pi*(self.radius**2)
        def circumference(self):
                return 2*Circle.pi*self.radius
r=int(input("Enter Radius: "))
C=Circle(r)
print("The Area =",C.area())
print("The Circumference =", C.circumference())
```

வெளியீடு

```
Enter Radius: 5
The Area = 78.5
The Circumference = 31.400000000000002
```

நிரல் 2: உன்னுடைய பள்ளி நூலகத்தில் உள்ள புத்தகங்களை பதிவு செய்து, புத்தகங்களின் பட்டியலை காட்டும் பைத்தான் நிரலை எழுதுக

```
class Library:
    def __init__(self):
        self.bookname=""
        self.author=""

    def getdata(self):
        self.bookname = input("Enter Name of the Book: ")
        self.author = input("Enter Author of the Book: ")

    def display(self):
        print("Name of the Book: ",self.bookname)
        print("Author of the Book: ",self.author)
        print("\n")

book=[] #empty list
ch = 'y'
while(ch=='y'):
    print("1. Add New Book \n 2.Display Books")
    resp = int(input("Enter your choice : "))
if(resp==1):
    L=Library()
    L.getdata()
    book.append(L)
elif(resp==2):
    for x in book:
        x.display()
else:
    print("Invalid input....")
ch = input("Do you want continue....")
```

வெளியீடு:

```
1. Add New Book
2.Display Books
Enter your choice : 1
Enter Name of the Book: Programming in C++
Enter Author of the Book: K. Kannan
Do you want continue....y
1. Add New Book
2.Display Books
Enter your choice : 1
Enter Name of the Book: Learn Python
Enter Author of the Book: V.G.Ramakrishnan
Do you want continue....y
```


Chapter 10.indd 182 30-01-2020 19:40:01

```
1. Add New Book
 2.Display Books
Enter your choice : 1
Enter Name of the Book: Advanced Python
Enter Author of the Book: Dr. Vidhya
Do you want continue....y
1. Add New Book
 2.Display Books
Enter your choice : 1
Enter Name of the Book: Working with OpenOffice
Enter Author of the Book: N.V.Gowrisankar
Do you want continue....y
1. Add New Book
 2.Display Books
Enter your choice : 1
Enter Name of the Book: Data Structure
Enter Author of the Book: K.Lenin
Do you want continue....y
1. Add New Book
 2.Display Books
Enter your choice : 1
Enter Name of the Book: An Introduction to Database System
Enter Author of the Book: R.Sreenivasan
Do you want continue....y
1. Add New Book
 2.Display Books
Enter your choice : 2
Name of the Book:  Programming in C++
Author of the Book:  K. Kannan

Name of the Book:  Learn Python
Author of the Book:  V.G.Ramakrishnan

Name of the Book:  Advanced Python
Author of the Book:  Dr. Vidhya

Name of the Book:  Working with OpenOffice
Author of the Book:  N.V.Gowrisankar

Name of the Book:  Data Structure
Author of the Book:  K.Lenin

Name of the Book:  An Introduction to Database System
Author of the Book:  R.Sreenivasan

Do you want continue....n
```

நிரல் 3: சரத்தை உள்ளீடாகப் பெற்று, அதில் எத்தனை பெரிய எழுத்துக்கள், சிறிய எழுத்துக்கள், உயிரெழுத்துகள், மெய்யெழுத்துகள் மற்றும் இடைவெளிகள் உள்ளன என்று கண்டறியும் பைத்தான் நிரலை எழுதுக.

```
class String:
    def __init__(self):
        self.uppercase=0
        self.lowercase=0
        self.vowels=0
        self.consonants=0
        self.spaces=0
        self.string=""
    def getstr(self):
        self.string=str(input("Enter a String: "))
    def count_upper(self):
            for ch in self.string:
                if (ch.isupper()):
                    self.uppercase+=1
    def count_lower(self):
            for ch in self.string:
                if (ch.islower()):
                    self.lowercase+=1
    def count_vowels(self):
        for ch in self.string:
            if (ch in ('A', 'a', 'e', 'E', 'i', 'I', 'o', 'O', 'u','U')):
                self.vowels+=1
    def count_consonants(self):
        for ch in self.string:
            if (ch not in ('A', 'a', 'e', 'E', 'i', 'I', 'o', 'O', 'u','U')):
                self.consonants+=1
    def count_space(self):
        for ch in self.string:
            if (ch==" "):
                self.spaces+=1
    def execute(self):
        self.count_upper()
        self.count_lower()
        self.count_vowels()
        self.count_consonants()
        self.count_space()
```


Chapter 10.indd 184   30-01-2020 19:40:01

```
        def display(self):
                print("The given string contains...")
                print("%d Uppercase letters"%self.uppercase)
                print("%d Lowercase letters"%self.lowercase)
                print("%d Vowels"%self.vowels)
                print("%d Consonants"%self.consonants)
                print("%d Spaces"%self.spaces)
S = String()
S.getstr()
S.execute()
S.display()
```

வெளியீடு

```
        Enter a String:Welcome To Learn Computer Science
        The given string contains...
        5 Uppercase letters
        24 Lowercase letters
        12 Vowels
        17 Consonants
        3 Spaces
```

நிரல் 4: பொருட்களையும், அதன் அடக்க விலையையும் சேமிக்கவும், சேமிக்கப்பட்ட பொருட்களை திரையில் தோன்றவும் செய்ய வேண்டும். அப்போது அனைத்து பொருட்களையும் அளவுகளை உள்ளீடாக பெற்று இறுதியில் விலைப்பட்டியலை அச்சிடும் பைத்தான் நிரல் ஒன்றை எழுதுக.

```
class MyStore:
        __prod_code=[]
        __prod_name=[]
        __cost_price=[]
        __prod_quant=[]
        def getdata(self):
                self.p = int(input("Enter no. of products you need to store: "))
                for x in range(self.p):
                        self.__prod_code.append(int(input("Enter Product Code: ")))
                        self.__prod_name.append(str(input("Enter Product Name: ")))
                        self.__cost_price.append(int(input("Enter Cost price: ")))
```

```
def display(self):
        print("Stock in Stores")
        print("----------------------------------------------------------")
        print("Product Code \t Product Name \t Cost Price")
        print("----------------------------------------------------------")
        for x in range(self.p):
                print(self.__prod_code[x], "\t\t", self.__prod_name[x], "\t\t", self.__
                                                                        cost_price[x])
        print("----------------------------------------------------------")

def print_bill(self):

        total_price = 0

        for x in range(self.p):

                q=int(input("Enter the quantify for the product code %d : "%self.__

                                                                        prod_code[x]))

self.__prod_quant.append(q)

total_price = total_price +self.__cost_price[x]*self.__prod_quant[x]

        print("           Invoice Receipt                    ")
        print("---------------------------------------------------------------------------")
        print("Product Code\t Product Name\t Cost Price\t Quantity \t Total
                                                                        Amount")
        print("---------------------------------------------------------------------------")
        for x in range(self.p):
                print(self.__prod_code[x], "\t\t", self.__prod_name[x], "\t\t",
                self.__cost_price[x], "\t\t", self.__prod_quant[x], "\t\t",
                self.__prod_quant[x]*self.__cost_price[x])
        print("---------------------------------------------------------------------------")
        print("                                      Total Amount = ", total_price)
S=MyStore()
S.getdata()
S.display()
S.print_bill()
```

வெளியீடு:

```
Enter no. of products you need to store: 5
Enter Product Code: 101
Enter Product Name: Product-A
Enter Cost price: 25
Enter Product Code: 201
Enter Product Name: Product-B
Enter Cost price: 35
Enter Product Code: 301
Enter Product Name: Product-C
Enter Cost price: 35
Enter Product Code: 401
Enter Product Name: Product-D
Enter Cost price: 50
Enter Product Code: 501
Enter Product Name: Product-E
Enter Cost price: 120

                    Stock in Stores
---------------------------------------------------------------------------------
    Product Code         Product Name              Cost Price
---------------------------------------------------------------------------------
    101                        Product-A               25
    201                        Product-B               35
    301                        Product-C               35
    401                        Product-D               50
    501                        Product-E               120
---------------------------------------------------------------------------------
```


Chapter 10.indd 187 30-01-2020 19:40:01

```
Enter the quantify for the product code 101 : 10
Enter the quantify for the product code 201 : 15
Enter the quantify for the product code 301 : 10
Enter the quantify for the product code 401 : 20
Enter the quantify for the product code 501 : 10

                        Invoice Receipt
------------------------------------------------------------------------------
   Product Code  Product Name       Cost Price    Quantity   Total Amount
------------------------------------------------------------------------------
   101                Product-A          25          10           250
   201                Product-B          35          15           525
   301                Product-C          35          10           350
   401                Product-D          50          20          1000
   501                Product-E         120          10          1200
------------------------------------------------------------------------------
                                             Total Amount = 3325
------------------------------------------------------------------------------
```

நினைவில் கொள்க

- பைத்தான் ஒரு பொருள் நோக்கு நிரலாக்க மொழி ஆகும்.
- இனக்குழுக்களும், பொருள்களும் பொருள் நோக்கு நிரலாக்கத்தின் முக்கிய அம்சங்களாகும்.
- பைத்தானில், class என்னும் சிறப்புச் சொல்லைப் பயன்படுத்தி இனக்குழு வரையறுக்கப்பட்டுள்ளது.
- இனக்குழுக்குள் வரையறுக்கப்படும் மாறிகள், இனக்குழு மாறிகள் எனவும் செயற்கூறுகள் வழிமுறைகள் எனவும் அழைக்கப்படுகிறது.
- ஒரு பொருளை உருவாக்கும் செயல்முறையே "சான்றுருவாக்கல்" எனப்படும்.
- ஒரு இனக்குழுவின் சான்றுரு பயன்பாட்டிற்கு வரும் பொழுது ஆக்கி என்னும் சிறப்புச் செயற்கூறு, தானாகவே இயக்கப்படுகிறது.
- இனக்குழுவில் உருவாக்கப்பட்ட பொருளின் பயன்பாடு முடிவுக்கு வரும் போது அழிப்பி என்னும் சிறப்பு செயற்கூறு தானாகவே இயக்கப்படும்.
- பைத்தானில், __del__( ) செயற்கூறு அழிப்பியாகப் பயன்படுத்தப்படுகிறது.
- இரட்டை அடிக்கீறினை முன்னொட்டாக கொண்ட மாறிகள் private மாறிகள் ஆகும்.

Chapter 10.indd 188 30-01-2020 19:40:01

செய்முறைப் பயிற்சி

1. இனக்குழுவைப் பயன்படுத்தி மாணவர்களின் பெயர் மற்றும் மதிப்பெண்களைப் பட்டியலாக சேமித்து மொத்த மதிப்பெண்ணை அச்சிடும் பைத்தான் நிரலை எழுதுக.

2. இனக்குழுவைப் பயன்படுத்தி முக்கோணத்தின் மூன்று பக்கங்களின் அளவை உள்ளீடாகப் பெற்று பரப்பளவைக் கண்டறியும் பைத்தான் நிரலை எழுதுக.

3. இரண்டு தூரங்களின் அளவை படிக்க, காண்பிக்க, கூட்ட மற்றும் கழிக்க உதவும் பட்டியல் முறை (Menu driven) நிரலை எழுதுக.

81GNU5

மதிப்பீடு

பகுதி–அ

சரியான விடையை தேர்ந்தெடுத்து எழுதுக **(1 மதிப்பெண்)**

1. பின்வருவனவற்றுள் எவை பொருள் நோக்கு நிரலாக்கத்தின் முக்கிய அம்சம் ஆகும்?
   (அ) ஆக்கி மற்றும் இனக்குழு (ஆ) ஆக்கி மற்றும் பொருள்
   (இ) இனக்குழு மற்றும் பொருள் (ஈ) ஆக்கி மற்றும் அழிப்பி

2. இனக்குழுவின் உள்ளே வரையறுக்கப்படும் செயற்கூறு எது:
   (அ) செயற்கூறு (ஆ) கூறு
   (இ) வழிமுறை (ஈ) பிரிவு

3. இனக்குழு உறுப்புகள் எந்த செயற்குறியின் மூலம் அணுகப்படுகிறது?
   (அ) & (ஆ) .
   (இ) # (ஈ) %

4. பொருள் உருவாக்கப்படும் போது தானாகவே இயக்கப்படும் செயற்கூறு எது?
   (அ) __object__( ) (ஆ) __del__( )
   (இ) __func__( ) (ஈ) __init__( )

5. private இனக்குழு மாறியின் முன்னொட்டு எது
   (அ) __ (ஆ) &&
   (இ) ## (ஈ) **

6. பின்வரும் வழிமுறைகளில் எது அழிப்பியாகப் பயன்படுகிறது?
   (அ) __init__( ) (ஆ) __dest__( )
   (இ) __rem__( ) (ஈ) __del__( )

Chapter 10.indd 189 30-01-2020 19:40:02

7. பின்வருவனவற்றுள் எந்த இனக்குழு அறிவிப்பு சரியானது?

(அ) class class_name (ஆ) class class_name<>

(இ) class class_name: (ஈ) class class_name[ ]

8. பின்வரும் நிரலின் வெளியீடு என்ன?

```
class Student:
        def __init__(self, name):
                self.name=name
                print(self.name)
S=Student("Tamil")
```

(அ) Error (ஆ) Tamil

(இ) name (ஈ) self

9. பின்வருவனவற்றுள் எது private இனக்குழு மாறி?

(அ) __num (ஆ) ##num

(இ) $$num (ஈ) &&num

10. பொருளை உருவாக்கும் செயல்முறை எது:

(அ) ஆக்கி (ஆ) அழிப்பு

(இ) மதிப்பிருத்தல் (ஈ) சான்றுருவாக்கல்

பகுதி–ஆ

அனைத்து வினாக்களுக்கும் விடையளி **(2 மதிப்பெண்)**

1. இனக்குழு என்றால் என்ன?
2. சான்றுருவாக்கல் என்றால் என்ன?
3. பின்வரும் நிரலின் வெளியீடு என்ன?

```
class Sample:
        __num=10
        def disp(self):
                print(self.__num)
S=Sample()
S.disp()
print(S.__num)
```

4. பைத்தானில் ஆக்கியை எவ்வாறு உருவாக்குவாய்?
5. அழிப்பியின் நோக்கம் என்ன?

Chapter 10.indd 190 30-01-2020 19:40:02

பகுதி–இ

அனைத்து வினாக்களுக்கும் விடையளி **(3 மதிப்பெண்)**

1. இனக்குழு உறுப்புகள் என்றால் என்ன? அதனை எவ்வாறு வரையறுப்பாய்?
2. இரண்டு private இனக்குழு மாறிகளுடன், வழிமுறையை பயன்படுத்தி கூட்டுத் தொகை sum அச்சிடும் இனக்குழுவை வரையறுக்கவும்.
3. கொடுக்கப்பட்ட வெளியீட்டை பெற பின்வரும் நிரலில் உள்ள பிழைகளை காண்க.

```
class Fruits:
        def __init__(self, f1, f2):
                self.f1=f1
                self.f2=f2
        def display(self):
                print("Fruit 1 = %s, Fruit 2 = %s" %(self.f1, self.f2))
F = Fruits ('Apple', 'Mango')
del F.display
F.display()
```

வெளியீடு

```
        Fruit 1 = Apple,   Fruit 2 = Mango
```

4. பின்வரும் நிரலின் வெளியீடு என்ன?

```
class Greeting:
        def __init__(self, name):
                self.__name = name
        def display(self):
                print("Good Morning ", self.__name)
obj=Greeting('Bindu Madhavan')
obj.display()
```

5. பைத்தானில் ஆக்கி மற்றும் அழிப்பிகளை எவ்வாறு வரையறுப்பாய்?

பகுதி–ஈ

அனைத்து வினாக்களுக்கும் விடையளி **(5 மதிப்பெண்)**

1. எழுது பொருட்களை (stationary) சேர்க்க அல்லது நீக்கும் பட்டியல் முறை நிரல் ஒன்றை எழுதுக. பொருட்களின் பெயர் மற்றும் பிராண்ட்-யை ஒரு dictionary -யில் சேமிக்க வேண்டும்.

References

1. *https://docs.python.org/3/tutorial/index.html*
2. *https://www.techbeamers.com/python-tutorial-step-by-step/#tutorial-list*
3. *Python programming using problem solving approach – Reema Thareja – Oxford University press.*
4. *Python Crash Course – Eric Matthes – No starch press, San Francisco.*