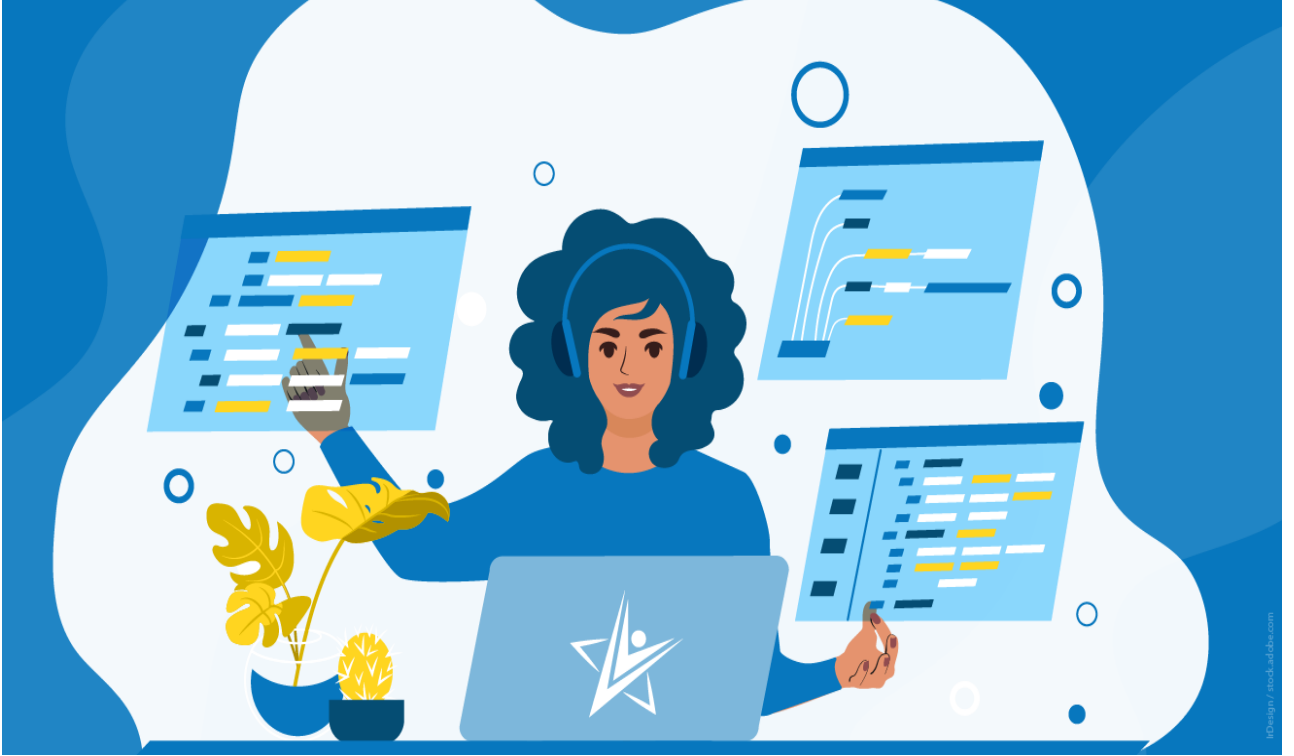


மேல்நிலை இரண்டாம் ஆண்டு

கணினி அறிவியல்

அலகு - III கூறுநிலை மற்றும் பொருள்நோக்கு நிரலாக்கம்
வினா - விடை தொகுப்பு

2023 - 24



ஜெ. கவிதா B.Sc, B.Ed, M.C.A, M.Phil.,

கணினி பயிற்றுநர் நிலை - I

அரசு மேல்நிலைப்பள்ளி,

சர்க்கார்சாமக்குளம்,

கோயம்புத்தூர் - 641107.

அலகு - III கூறுநிலை மற்றும் பொருள்நோக்கு நிரலாக்கம்

பாடம் 9. (List, Tuples, Set மற்றும் Dictionary) தொகுப்பு தரவினங்கள்

1 மதிப்பெண் வினாக்கள்:

சரியான விடையை தேர்ந்தெடுத்து எழுதுக.

1. தரவினத் தொகுதியின் தொடர்பில்லாத ஒன்றைத் தேர்வு செய்க.
அ) list ஆ) tuple இ) dictionary ஈ) **Loop**
2. Let list1 = [2,4,6,8,10], எனில் print(list1[-2]) ன் விடை
அ) 10 ஆ) **8** இ) 4 ஈ) 6
3. பின்வரும் எந்த செயற்கூறு list - ல் உள்ள உறுப்புகளின் எண்ணிக்கையைக் கணக்கிட பயன்படுகிறது?
அ) count() ஆ) find() இ) **len()** ஈ) index()
4. If list=[10,20,30,40,50] எனில் list[2]=35 ன் விடை
அ) [35,10,20,30,40,50] ஆ) [10,20,30,40,50,35] இ) **[10,20,35,40,50]** ஈ) [10,35,30,40,50]
5. if list=[17,23,41,10] எனில் list.append (32) ன் விடை
அ) [32,17,23,41,10] ஆ) **[17,23,41,10,32]** இ) [10,17,23,32,41] ஈ) [41,32,23,17,10]
6. பின்வரும் எந்த பைத்தான் செயற்கூறு ஏற்கனவே உள்ள list-ல் ஒன்றுக்கும் மேற்பட்ட உறுப்புகளை சேர்க்கப் பயன்படுகிறது?
அ) Append() ஆ) append_more() இ) **extend()** ஈ) more()
7. பின்வரும் பைத்தான் குறிமுறையின் விடை என்ன?
S=[x**2 for x in range(5)]
Print(S)
அ) [0,1,2,4,5] ஆ) **[0,1,4,9,16]** இ) [0,1,4,9,16,25] ஈ) [1,4,9,16,25]
8. பைத்தானில் type() செயற்கூறின் பயன் என்ன?
அ) tuple உருவாக்க
ஆ) tuple உள்ள உறுப்புகளின் வகையைக் கண்டறிய
இ) **பைத்தான் பொருளின் தரவினத்தை கண்டறிய**
ஈ) பட்டியலை உருவாக்க
9. பின்வரும் எந்த கூற்று சரியானது அல்ல?
அ) list மாற்றம் செய்யலாம் ஆ) tuples மாற்றம் செய்யலாம்
இ) append() செயற்கூறு, ஒரு உறுப்பை சேர்க்கப் பயன்படுகிறது
ஈ) **extend() செயற்கூறு list-ல் உறுப்புகளை சேர்க்க tuples-ல் பயன்படுகிறது.**
10. Set A={3,6,9}, set B={1,3,9} எனில், பின்வரும் நிரலின் வெளியீடு என்ன?
Print(set A|set B)
அ) {3,6,9,1,3,9} ஆ) {3,9} இ) {1} ஈ) **{1,3,6,9}**
11. பின்வரும் எந்த set செயல்பாடு, இரண்டு set-களுக்கும் பொதுவான உறுப்புகள் நீங்கலாக மற்ற அனைத்து உறுப்புகளையும் உள்ளடக்கியது?
அ) **சமச்சீரான வேறுபாடு** ஆ) வேறுபாடு இ) வெட்டு ஈ) சேர்ப்பு
12. பைத்தான், dictionary- ல் திறவுகோல்கள் எதனால் குறிப்பிடப்படுகின்றன.
அ) = ஆ) ; இ) + ஈ) :

2 மதிப்பெண் வினாக்கள்:

1. பைத்தானில் List என்றால் என்ன?

- பைத்தானில் List என்பது சதுர அடைப்புக்குறிக்குள் அடைக்கப்பட்ட மதிப்புகளின் வரிசைப்படுத்தப்பட்ட தொகுப்பு தரவினமாகும்..
- List - ல் உள்ள ஒவ்வொரு மதிப்பும் உறுப்பு என்று அழைக்கப்படுகிறது.
- இது எண்கள், எழுத்துக்கள், சரநிலையருக்கள் மற்றும் பின்னலான List போன்ற எந்த வகையாகவும் இருக்கலாம்.

2. List உறுப்புகளை பின்னோக்கு வரிசையில் தலைகீழாக எவ்வாறு அணுகுவாய்?

- List - ல் உறுப்புகளை எதிர்மறை சுட்டெண் மதிப்பைக் கொண்டு இறுதியிலிருந்து பின்னோக்கு வரிசையில் தலைகீழாக அணுக முடியும்.

3. பின்வரும் பைத்தான் குறிமுறையில் x - ன் மதிப்பு என்ன?

```
List1 = [2,4,6,[1,3,5]]
```

```
x = len(List1)
```

- x - ன் மதிப்பு - 4

4. List - ன் del மற்றும் remove() செயற்கூறின் வேறுபாடுகள் யாவை?

del கூற்று	remove() செயற்கூறு
del() கூற்று சுட்டெண் தெரிந்த உறுப்புகளை list - லிருந்து நீக்குவதற்கு பயன்படுகிறது.	remove() செயற்கூறு சுட்டெண் தெரியாத உறுப்புகளை list - லிருந்து நீக்குவதற்கு பயன்படுகிறது.

5. ஒரு Tuples n எண்ணிக்கை உறுப்புகளுடன் உருவாக்குவதற்கான தொடரியலை எழுதுக.

- n எண்ணிக்கை உறுப்புகளுடன் Tuples உருவாக்குவதற்கான தொடரியல்:
Tuples_Name = (E1, E2, E3, . . . En) (அல்லது) Tuples_Name = E1, E2, E3, . . . En

6. பைத்தானில் set என்றால் என்ன?

- பைத்தானில் set என்பது தரவின தொகுப்பின் மற்றொரு வகையாகும்.
- set என்பது மாறக்கூடிய மற்றும் நகல்கள் இல்லாத வரிசைப்படுத்தப்படாத உறுப்புகளின் தொகுப்பாகும்.

3 மதிப்பெண் வினாக்கள்:

1. List மற்றும் Tuples - ஒப்பிடுக.

- List - ன் உறுப்புகளை மாற்றலாம். ஆனால் Tuples - ன் உறுப்புகளை மாற்ற முடியாது. இதுவே List மற்றும் Tuples-க்கு இடையே உள்ள முக்கியமான வேறுபாடு ஆகும்.
- List - ன் உறுப்புகள் சதுர அடைப்புக்குறிக்குள் அடைக்கப்பட்டிருக்கும். ஆனால் Tuples வளைவு அடைப்புக் குறிக்குள் அடைக்கப்பட்டிருக்கும்.
- Tuples - ன் மடக்கு செயல் List - ஐ காட்டிலும் விரைவானது.

2. Sort() பற்றி சிறுகுறிப்பு எழுதுக.

- Sort() செயற்கூறு List - ல் உள்ள உறுப்புகளை வரிசையாக்கம் செய்ய பயன்படுகிறது.

தொடரியல்: List.sort(reverse=True/False,key=myFunc)

- Reverse - ஐ True என பொருத்தினால் இறங்கு வரிசையில் List வரிசையாக்கம் செய்யும். ஏறுவரிசை தானமைவு வரிசையாக்கம் ஆகும்.

3. பின்வரும் குறிமுறையின் வெளியீடு என்ன?

```
list = [2*x for x in range(5)]
print(list)
```

வெளியீடு: [1,2,4,8,16]

4. del மற்றும் clear() செயற்கூறுகளுக்கு இடையேயான வேறுபாடுகளை எடுத்துக்காட்டுடன் வெளக்குக.

del() கூற்று	clear() செயற்கூறு
del கூற்று சுட்டெண் தெரிந்த உறுப்புகளை நீக்குவதற்கும், முழு List - ஐ நீக்குவதற்கும் பயன்படுகிறது.	clear() செயற்கூறு List - ன் அனைத்து உறுப்புகளையும் நீக்கி List - ஐ தொடர்ந்து வைத்திருக்க பயன்படுகிறது.
எ.கா: >>> MySubjects = ['Tamil', 'English'] >>> del MySubjects[1] >>> print (MySubjects) வெளியீடு: ['Tamil']	எ.கா: >>> MySubjects = ['Tamil', 'English'] >>> MySubjects.clear() >>> print (MySubjects) வெளியீடு: []

5. பைத்தானின் set செயல்பாடுகளை பட்டியலிடுக.

- **சேர்ப்பு:** இரண்டு அல்லது அதற்கு மேற்பட்ட set களின் அனைத்து உறுப்புகளையும் உள்ளடக்கும்.
- **வெட்டு:** இது இரண்டு set களின் பொதுவான உறுப்புகளை உள்ளடக்கியது.
- **வேறுபாடு:** இது முதல் set-ல் உள்ள உறுப்புகள் இரண்டாவது set-ல் இருந்தால் அதை தவிர்த்து மற்ற உறுப்புகளை உள்ளடக்கியது.
- **சமச்சீரான வேறுபாடு:** இது இரண்டு set களில் உள்ள பொதுவான உறுப்புகளை மட்டும் தவிர்த்து மற்ற அனைத்து உறுப்புகளையும் உள்ளடக்கியது.

6. List மற்றும் Dictionary இடையேயான வேறுபாடுகள் யாவை?

List	Dictionary
List என்பது வரிசைபடுத்தப்பட்ட உறுப்புகளின் தொகுப்பாகும்.	Dictionary ஒரு உறுப்பை மற்றொரு உறுப்புடன் பொருத்த பயன்படும் தரவு அமைப்பாகும்.
List - ன் சுட்டெண் குறிப்பிட்ட உறுப்பை அணுகுவதற்கு பயன்படுகிறது.	Dictionary - ல் திறவுகோல் சுட்டெண்ணை குறிக்கிறது.
List - ன் மதிப்பை பார்த்துக் கொள்ள பயன்படுகிறது.	Dictionary - ல் ஒரு மதிப்பை எடுத்துக்கொண்டு மற்றொரு மதிப்பை பார்த்துக்கொள்ள பயன்படுகிறது.

5 மதிப்பெண் வினாக்கள்:

1. List - ல் ஒரு உறுப்பை சேர்ப்பதற்கான பல்வேறு வழிகள் யாவை? பொருத்தமான எடுத்துக்காட்டுடன் விளக்குக.

- **append() செயற்கூறு:** பைத்தானில், ஏற்கனவே உள்ள list - ன் இறுதியில் ஒரு உறுப்பை சேர்ப்பதற்கு append() செயற்கூறு பயன்படுகிறது.

தொடரியல்: List.append(element to be added)

எ.கா:
>>> MyList=[34, 45, 48]
>>> MyList.append(90)
>>> print(MyList)

வெளியீடு: [34, 45, 48, 90]

- **extend() செயற்கூறு:** பைத்தானில், ஏற்கனவே உள்ள list - ன் இறுதியில் ஒன்றுக்கும் மேற்பட்ட உறுப்புகளை சேர்ப்பதற்கு extend() செயற்கூறு பயன்படுகிறது. **தொடரியல்:** List.extend([elements to be added])

எ.கா:
>>> MyList=[34, 45, 48]
>>> MyList.extend ([71, 32, 29])
>>> print(MyList)

வெளியீடு: [34, 45, 48, 90, 71, 32, 29]

- **Insert() செயற்கூறு:** பைத்தானில், ஏற்கனவே உள்ள list - ன் எந்தவொரு இடத்திலும் ஒரு உறுப்பை சேர்ப்பதற்கு Insert() செயற்கூறு பயன்படுகிறது.

தொடரியல்: List.insert (position index, element)

எ.கா:
>>> MyList=[34,98,47,'Kannan', 'Gowrisankar', 'Lenin', 'Sreenivasan']
>>> MyList.insert(3, 'Ramakrishnan')
>>> print(MyList)

வெளியீடு: [34, 98, 47, 'Ramakrishnan', 'Kannan', 'Gowrisankar', 'Lenin', 'Sreenivasan']

2. Range() ன் நோக்கம் என்ன? எடுத்துக்காட்டுடன் விளக்குக.

- Range() செயற்கூறு பைத்தானில் தொடர் மதிப்புகளை உருவாக்க பயன்படுகிறது. Range() செயற்கூறு மூன்று செயலுருபுகளைக் கொண்டுள்ளது.

தொடரியல்: range (start value, end value, step value)

- start value - தொடரின் தொடக்க மதிப்பு.
- end value - தொடரின் உச்ச வரம்பு.
- step value - இது ஒரு விருப்ப செயலுருபு , இது வெவ்வேறு இடைவெளிகளில் மதிப்புகளை உருவாக்கப் பயன்படுகிறது.

எ.கா: 2 முதல் 10 வரை உள்ள இரட்டைப்படை எண்களை உருவாக்குதல்.

```
For x in range(2,11,2):  
    Print(x, end = ' ')
```

வெளியீடு: 2 4 6 8 10

3. பின்னலான Tuples என்றால் என்ன? எடுத்துக்காட்டுடன் விளக்குக.

- பைத்தானில், ஒரு Tuples - ஐ மற்றொரு Tuples - க்குள் வரையறை செய்வதை பின்னலான Tuples என்கிறோம்.
- பின்னலான Tuples - ல் ஒவ்வொரு Tuples - ம் ஒரு உறுப்பாக கருதப்படுகிறது.
- For மடக்கு பின்னலான Tuples - ன் அனைத்து உறுப்புகளை அணுகுவதற்கு பயன்படுகிறது.

எ.கா: Toppers = ("Vinodini", "XII-F", 98.7), ("Soundarya", "XII-H", 97.5), ("Tharani", "XII-F", 95.3)
for i in Toppers:
 print(i)

வெளியீடு: ('Vinodini', 'XII-F', 98.7)
('Soundarya', 'XII-H', 97.5)
('Tharani', 'XII-F', 95.3)

4. பைத்தானிலுள்ள பல்வேறு set செயல்பாடுகளை பொருத்தமான எடுத்துக்காட்டுகளுடன் விளக்குக.

சேர்ப்பு (Union): இரண்டு அல்லது அதற்கு மேற்பட்ட set களின் அனைத்து உறுப்புகளையும் உள்ளடக்கும். பைத்தானில், |, Union செயற்கூறுகள் set சேர்ப்புக்காக பயன்படுத்தப்படுகிறது.

எ.கா:
set_A={2,4,6,8}
set_B={'A', 'B', 'C', 'D'}
U_set=set_A|set_B
print(U_set)

வெளியீடு: {2, 4, 6, 8, 'A', 'D', 'C', 'B'}

வெட்டு (Interseccion): இது இரண்டு set களின் பொதுவான உறுப்புகளை உள்ளடக்கியது. பைத்தானில், &, intersection செயற்கூறுகள் set களை வெட்டுவதற்கு பயன்படுகிறது.

எ.கா:
set_A={'A', 2, 4, 'D'}
set_B={'A', 'B', 'C', 'D'}
print(set_A & set_B)

வெளியீடு: {'A', 'D'}

வேறுபாடு (Difference): இது முதல் set-ல் உள்ள உறுப்புகள் இரண்டாவது set-ல் இருந்தால் அதை தவிர்த்து மற்ற உறுப்புகளை உள்ளடக்கியது.. பைத்தானில், - (minus), Difference செயற்கூறுகள் வேறுபாடு செயற்கூறாக பயன்படுகிறது.

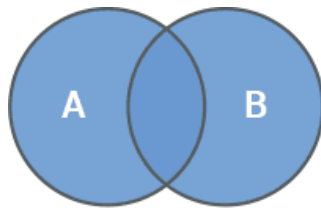
எ.கா:
set_A={'A', 2, 4, 'D'}
set_B={'A', 'B', 'C', 'D'}
print(set_A - set_B)

வெளியீடு: {2, 4}

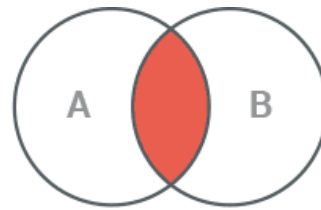
சமச்சீரான வேறுபாடு (Symmetric Difference): இது இரண்டு set களில் உள்ள பொதுவான உறுப்புகளை மட்டும் தவிர்த்து மற்ற அனைத்து உறுப்புகளையும் உள்ளடக்கியது. பைத்தானில், ^ (carret), Symmetric difference செயற்கூறுகள் சமச்சீரான வேறுபாடு கண்டறிய பயன்படுகிறது

எ.கா:
set_A={'A', 2, 4, 'D'}
set_B={'A', 'B', 'C', 'D'}
print(set_A ^ set_B)

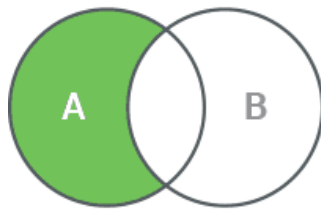
வெளியீடு: {2, 4, 'B', 'C'}



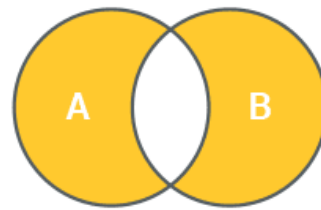
Union



Intersection



Difference



Symmetric Difference

அலகு - III கூறுநிலை மற்றும் பொருள்நோக்கு நிரலாக்கம்

பாடம் 10. பைத்தான் இனக்குழுக்கள் மற்றும் பொருள்கள்

1 மதிப்பெண் வினாக்கள்:

சரியான விடையை தேர்ந்தெடுத்து எழுதுக.

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 வழிமுறையில் எது அழிப்பியாகப் பயன்படுகிறது?
அ) __init__() ஆ) __dest__() இ) __rem__() ஈ) __del__()
7. பின்வரும் 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
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) சான்றொதுவாக்கல்

2 மதிப்பெண் வினாக்கள்:

1. இனக்குழு என்றால் என்ன?

- இனக்குழு என்பது பொருளின் வார்ப்புரு ஆகும்.
- இது பைத்தானில் மிக முக்கிய கட்டமைப்பு கூறுகளாகத் திகழ்கிறது.

2. சான்றுருவாக்கல் என்றால் என்ன?

- இனக்குழு உருவாக்கப்பட்ட பின் அந்த இனக்குழுவின் பொருள் அல்லது சான்றுரு உருவாக்கப்பட வேண்டும்.
- பொருளை உருவாக்கும் இந்த முறைக்கு “சான்றுருவாக்கல்” என்று பெயர். .

3. பின்வரும் நிரலின் வெளியீடு என்ன?

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

S=Sample()
S.disp()
print(S.__num)
```

வெளியீடு:

- நிரலில் S.disp() என்ற கூற்று 10 என்ற மதிப்பை வெளியீடாகத் தரும்.
- ஆனால் print(S.__num) என்ற கூற்று Attribute Error என்ற பிழை செய்தியைக் காட்டும். ஏனெனில் __num என்ற மாறி இனக்குழுவின் private உறுப்பாகும். அதை இனக்குழுவிற்கு வெளியிலிருந்து அணுகமுடியாது.

4. பைத்தானில் ஆக்கியை எவ்வாறு உருவாக்குவாய்?

- பைத்தானில், Init என்னும் சிறப்பு செயற்கூறு ஆக்கியாக செயல்படுகிறது. இது இரட்டை அடிக்கீறில் தொடங்கி இரட்டை அடிக்கீறலுடன் முடியும்.
- ஆக்கி செயற்கூறை அளப்புக்களுடனோ அல்லது இல்லாமலோ வரையறுக்கலாம். ஆக்கி வழிமுறையின் பொதுவடிவம்:

```
def __init__(self,[args.....]):
    <statements>
```

5. அழிப்பியின் நோக்கம் என்ன?

- இனக்குழுவில் உருவாக்கப்பட்ட பொருளின் பயன்பாடு, முடிவுக்கு வரும் போது அழிப்பி என்னும் சிறப்பு செயற்கூறு தானாகவே இயக்கப்படும்.
- __del__() செயற்கூறு அழிப்பியாக பயன்படுத்தப்படுகிறது.

3 மதிப்பெண் வினாக்கள்:

1. இனக்குழு உறுப்புகள் என்றால் என்ன? அதனை எவ்வாறு வரையறுப்பாய்?

- இனக்குழுவிற்குள்ளே வரையறுக்கப்படும் மாறிகள் இனக்குழு மாறிகள் என்றும், செயற்கூறுகள் வழிமுறைகள் என்றும் அழைக்கப்படும். இனக்குழு மாறிகள் மற்றும் வழிமுறைகள் இரண்டும் சேர்த்து இனக்குழுவின் உறுப்புகள் எனப்படும்.
- இனக்குழுவிற்குள் மாறிகளை அறிவிக்கும் பொழுது இனக்குழுவின் பெயர் மற்றும் புள்ளி செயற்குறியைக் கொண்டு இருக்க வேண்டும்.
- வழிமுறையின் முதல் அளபுருவாக self என இருக்க வேண்டும்.

எ.கா: class Sample:

```
x = 10          → இனக்குழு மாறி
def disp(self): → வழிமுறை
    print(Sample.x)
s=Sample( )
s.disp( )
```


2. இரண்டு `private` இனக்குழு மாறிகளுடன், வழிமுறையை பயன்படுத்தி கூட்டுத்தொகை `sum` அச்சிடும் இனக்குழுவை வரையறுக்கவும்.
நிரல்:

```
class Sum:
    def __init__(self,n1,n2):
        self.__num1=n1
        self.__num2=n2
    def display(self):
        print(self.__num1+self.__num2)
S=Sum(12,14)
S.display()
```

வெளியீடு: 26

3. கொடுக்கப்பட்ட வெளியீட்டை பெற பின்வரும் நிரலில் உள்ள பிழைகளைக் 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

- `del F.display` என்ற வரிக்கு Attribute Error என்ற பிழை செய்தியைக் காட்டும். அதை நீக்கினால் கொடுக்கப்பட்ட வெளியீட்டை பெறலாம்.

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()
```

வெளியீடு: Good Morning Bindu Madhavan

5. பைத்தானில் ஆக்கி மற்றும் அழிப்பிகளை எவ்வாறு வரையறுப்பாய்?

ஆக்கி: Init என்னும் சிறப்பு செயற்கூறு ஆக்கியாக செயல்படுகிறது. இது இரட்டை அடிக்கீறில் தொடங்கி இரட்டை அடிக்கீறலுடன் முடியும்.

- ஆக்கி செயற்கூறை அளபுருக்களுடனோ அல்லது இல்லாமலோ வரையறுக்கலாம்.

ஆக்கி வழிமுறையின் பொதுவடிவம்:

```
def __init__ (self,[args.....]):
    <statements>
```

அழிப்பி: இது ஆக்கிக்கு முரணானது. பைத்தானில் `__del__()` செயற்கூறு அழிப்பியாக பயன்படுத்தப்படுகிறது.

5 மதிப்பெண் வினாக்கள்:

1. ஆக்கி மற்றும் அழிப்பி பற்றி பொருத்தமான எடுத்துக்காட்டுடன் விளக்குக.

ஆக்கி: ஒரு இனக்குழுவின் சான்றுரு பயன்பாட்டிற்கு வரும்பொழுது ஆக்கி என்னும் சிறப்புச் செயற்கூறு தானாகவே இயக்கப்படுகிறது.

- Init என்னும் சிறப்பு செயற்கூறு ஆக்கியாக செயல்படுகிறது. இது இரட்டை அடிக்கீறில் தொடங்கி இரட்டை அடிக்கீறலுடன் முடியும்.
- ஆக்கி செயற்கூறை அளபுருக்களுடனோ அல்லது இல்லாமலோ வரையறுக்கலாம்.

ஆக்கி வழிமுறையின் பொதுவடிவம்:

```
def __init__(self,[args....]):  
    <statements>
```

அழிப்பி: இனக்குழுவில் உருவாக்கப்பட்ட பொருளின் பயன்பாடு முடிவுக்கு வரும் போது அழிப்பி என்னும் சிறப்பு செயற்கூறு இயக்கப்படும்.

- இது ஆக்கிக்கு முரணானது. பைத்தானில் `__del__()` செயற்கூறு அழிப்பியாக பயன்படுத்தப்படுகிறது.

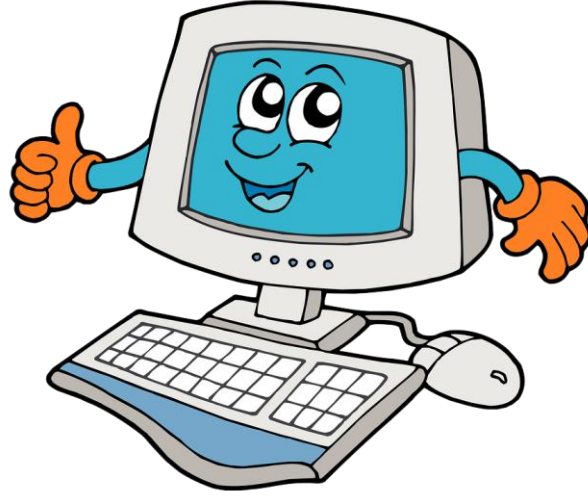
எ.கா:

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

வெளியீடு:

```
Constructor of class Sample...  
The value is : 10  
Destructor of class Sample...
```

இளமைக் காலத்தில்
கல்வியை புறக்கணித்தவன்
எதிர்கால வாழ்வை
இழந்தவன் ஆகிறான்.
கல்வியின் மேன்மை உணர்ந்து
அக்கல்வியைப் பெற்று சிறப்போடு வாழ
வாழ்த்துக்கள்.



ஜெ. கவிதா B.Sc, B.Ed, M.C.A, M.Phil.,
கணினி பயிற்றுநர் நிலை - I
அரசு மேல்நிலைப்பள்ளி,
சர்க்கார்சாமக்குளம்,
கோயம்புத்தூர் - 641107.
☎: 8940762362