https://www.Padasalai.Net https://www.CBSEtips.in

12 ஆம் வகுப்பு

உடனடித்தேர்வு - ஜுன் 2023

PART - III

கணினி அறிவியல் (விடைகளுடன்)

தேர்வு எண்

கால அளவு : 3.00 மணி நேரம்] [மொத்த மதிப்பெண்கள் : 70

அறிவுரைகள்:

(1) அனைத்து வினாக்களும் சரியாக பதிவாகி உள்ளதா என்பதனை சரிபார்த்துக் கொள்ளவும். அச்சுப்பதிவில் குறையிருப்பின், அறைக் கண்காணிப்பாளரிடம் உடனடியாகத் தெரிவிக்கவும்.

(2) **நீலம்** அல்லது **கருப்பு** மையினை மட்டுமே எழுதுவதற்கும், அடிகோடிடுவதற்கும் பயன்படுத்த வேண்டும். படங்கள் வரைவதற்கு பென்சில் பயன்படுத்தவும்.

பகுதி - I

குறிப்பு: (i) **அனைத்து** வினாக்களுக்கும் விடையளிக்கவும். **(15 × 1 = 15)**

(ii) கொடுக்கப்பட்டுள்ள மாற்று விடைகளில் மிகவும் ஏற்புடைய விடையைத் தேர்ந்தெடுத்துக் குறியீட்டுடன் விடையினையும் சேர்த்து எழுதவும்.

1. செயற்கூறு வரையறையில் உள்ள மாறிகள் ________ என அழைக்கப்படுகிறது.
(அ) துணை நிரல்கள் (ஆ) செயற்கூறு
(இ) அளபுருக்கள் (ஈ) செயலுருபு

2. மாற்றம் செய்ய முடியாத பொருளின் தொடர்வரிசை:
(அ) Built in (ஆ) List
(இ) Tuple (ஈ) Derived data

3. மாறியின் பெயரை பொருளுடன் மேப்பிங் செய்வதற்கான இடம் ________ ஆகும்.
(அ) வரையெல்லை (ஆ) மேப்பிங்
(இ) பிணைத்தல் (ஈ) Namespaces

4. நெறிமுறையின் செயல்திறனை தீர்மானிக்கும் இரண்டு முக்கிய காரணிகள்.
(அ) செயலி மற்றும் நினைவகம்
(ஆ) சிக்கல் மற்றும் கொள்ளளவு
(இ) நேரம் மற்றும் இடம் (ஈ) தரவு மற்றும் இடம்

5. ஓர் புதிய பைத்தான் நிரலை உருவாக்கப் பயன்படும் சாவி சேர்மானம்.
(அ) Ctrl+C (ஆ) Ctrl+F
(இ) Ctrl+B (ஈ) Ctrl+N

6. "elif" என்பது எதனுடைய சுருக்கம்?
(அ) nested if (ஆ) if.else
(இ) else.if (ஈ) if..elif

7. "பெயரில்லா செயற்கூறு" என்று அழைக்கப்படும் செயற்கூறு :
(அ) லாம்டா (ஆ) தற்சுழற்சி
(இ) செயற்கூறு (ஈ) வரையறை

8. format( ) செயற்கூறுடன் பயன்படும் பதிலீடு குறியீடு.
(அ) {} (ஆ) < > (இ) ++ (ஈ) ^^

9. தரவினத் தொகுதியின் தொடர்பில்லாத ஒன்றைத் தேர்வு செய்க.
(அ) List (ஆ) Tuple
(இ) Dictionary (ஈ) Loop

10. பொருளை உருவாக்கும் செயல்முறை :
(அ) ஆக்கி (ஆ) அழிப்பு
(இ) மதிப்பிருத்தல் (ஈ) சான்றுருவாக்கம்

11. ஒரு அட்டவணை என்பது ________.
(அ) வரிசை (tuple)
(ஆ) பண்புக்கூறுகள் (attribute)
(இ) உறவுகள் (relation) (ஈ) அமைப்பு (entity)

12. ஒரு தரவுத்தளத்தில் உள்ள தரவை வரிசையாக்கம் செய்ய பயன்படும் Clause:
(அ) SORT BY (ஆ) ORDER BY
(இ) GROUP BY (ஈ) SELECT

13. CSV கோப்பினில் ஒரு வரிசையை தவிர்க்க பயன்படும் கட்டளை:
(அ) next( ) (ஆ) skip( )
(இ) omit( ) (ஈ) bounce( )

14. விண்டோஸ் இயக்க முறைமையுடன் தொடர்பு கொள்ள அனுமதிக்கும் கூறுநிலை:
(அ) OS கூறுநிலை (ஆ) Sys கூறுநிலை
(இ) CSV கூறுநிலை (ஈ) getopt கூறுநிலை

15. SQL-ல் மிகவும் பொதுவாக பயன்படுத்தப்படும் கூற்று:
(அ) cursor (ஆ) select
(இ) execute (ஈ) commit

பகுதி - II

குறிப்பு: **எவையேனும் ஆறு** வினாக்களுக்கு விடையளிக்கவும். வினா எண் 24-க்கு **கட்டாயமாக** விடையளிக்கவும். **(6 × 2 = 12)**

16. இடைமுகத்தையும், செயல்படுத்துதலையும் வேறுபடுத்துக.
17. Pair என்றால் என்ன? எடுத்துக்காட்டு தருக.
18. வில்லைகள் பற்றி சிறுகுறிப்பு வரைக.
19. பைத்தானில் உள்ள கட்டுபாட்டு கட்டமைப்புகளைப் பட்டியலிடுக.
20. செயற்கூறின் முக்கிய நன்மைகள் யாவை?
21. தரவு நிலைத் தன்மை என்றால் என்ன?
22. அட்டவணை கட்டுப்பாட்டிற்கும், நெடுவரிசை கட்டுப்பாட்டிற்கும் உள்ள வேறுபாடுகளை எழுதுக.
23. குறிப்பு வரைக.
(i) MAX( ) செயற்கூறு (ii) MIN( ) செயற்கூறு
24. பைத்தானில் Set என்றால் என்ன?

பகுதி - III

குறிப்பு: **எவையேனும் ஆறு** வினாக்களுக்கு விடையளிக்கவும். வினா எண் 33-க்கு **கட்டாயமாக** விடையளிக்கவும். **(6 × 3 = 18)**

25. தொகுதிகளின் மூன்று பண்பியல்புகளை எழுதுக.
26. Asymptotic குறியீடு - குறிப்பு வரைக.
27. சரநிலையுறு என்றால் என்ன?

Kindly send me your study materials to padasalai.net@gmail.com

https://www.Padasalai.Net https://www.CBSEtips.in

28. if else அமைப்பைப் பற்றி குறிப்பு வரைக.
29. செயற்கூறில் தொகுப்பு பற்றி எடுத்துக்காட்டுடன் எழுதுக.
30. List மற்றும் Dictionary இடையேயான வேறுபாடுகள் யாவை?
31. DBA-ன் பணி யாது?
32. பைத்தான் மற்றும் C++-ஐ வேறுபடுத்துக.
33. கொடுக்கப்பட்ட வருடம் லீப் வருடமா இல்லையா என்பதனைச் சோதிக்கும் பைத்தான் நிரலை எழுதுக.

பகுதி - IV

குறிப்பு: அனைத்து வினாக்களுக்கும் விடையளிக்கவும்.

(5 × 5 = 25)

34. (அ) இருமத் தேடல் என்றால் என்ன? எடுத்துக்காட்டுடன் விளக்குக.

அல்லது

(ஆ) தரவு அருவமாக்கத்தினை எவ்வாறு செயல்படுத்துவாய்? எடுத்துக்காட்டுடன் விளக்குக.

35. (அ) Input( ) மற்றும் print( ) செயற்கூறுகள் பற்றி விளக்குக.

அல்லது

(ஆ) மாறியின் வரையெல்லைகளை எடுத்துக்காட்டுடன் விளக்குக.

36. (அ) பைத்தானில் சரசெயற்குறிகளை தகுந்த எடுத்துக்காட்டுடன் விளக்குக.

அல்லது

(ஆ) List-ல் ஒரு உறுப்பை சேர்ப்பதற்கான பல்வேறு வழிமுறைகள் யாவை? பொருத்தமான எடுத்துக்காட்டுடன் விளக்குக.

37. (அ) DBMS-ன் பண்பியல்புகளை விளக்குக.

அல்லது

(ஆ) SQLite பற்றி விரிவாக எழுதவும் அதனை பயன்படுத்தும் படிநிலைகளை எழுதுக.

38. (அ) பைத்தானில் ஒரு கோப்பினை படிப்பதற்கான பல்வேறு வழிமுறைகள் யாவை?

அல்லது

(ஆ) பைத்தானில் ஏதேனும் 5 பண்புக் கூறுகளை எழுதுக.

விடைகள்

பகுதி - I

1. (ஈ) செயலுருபு
2. (இ) Tuple
3. (ஈ) Namespaces
4. (இ) நேரம் மற்றும் இடம்
5. (ஈ) Ctrl+N
6. (இ) else if
7. (அ) லாம்டா
8. (அ) {}
9. (ஈ) Loop
10. (ஈ) சான்றுருவாக்கல்
11. (இ) உறவுகள் (relation)
12. (ஆ) ORDER BY
13. (அ) next( )
14. (அ) OS கூறுநிலை
15. (ஆ) select

பகுதி - II

16.

இடைமுகம்	செயல்படுத்துதல்
ஒரு பொருள் செய்யக்கூடிய நடவடிக்கையை வரையறுக்கிறது, ஆனால் அவற்றை உண்மையில் செய்யக் கூடியது இல்லை.	இடைமுகத்தில் வரையறுக்கப்பட்டுள்ள கட்டளைகளை நிறைவேற்றுகிறது.

17. இரு மதிப்புகளை ஒன்றாக இணைக்கும் முறையை இணைகள் (Pairs) என்று அழைக்கிறோம். List அதிகமாக பயன்படுத்தப்படும் முறையாகும். எனவே, List-கள் Pair என்று அழைக்கப்படுகிறது.

எ.கா. lst : = [10, 20]
lst [0] = 10, lst [1] = 20.

இவ்வாறு இடமதிப்பும், உறுப்பின் மதிப்பும் பிணைக்கப்படுதல் Pair எனப்படும்.

18. பைத்தான், நிரலில் இடம்பெறும் நிரல் வரிகளை அடிப்படை மொழிக்கூற்றுகளாக பிரிக்கிறது. இந்த கூறுகள் வில்லைகள் எனப்படும் பொதுவாக வில்லைகளின் வகைகளாவன.
(i) குறிப்பெயர்கள் (ii) சிறப்புச் சொற்கள்
(iii) செயற்குறிகள் (iv) வரம்புக்குறிகள்
(v) நிலைஉருக்கள்

வில்லைகளை பிரிப்பதற்கு வெற்று இடைவெளி கண்டிப்பாக இருத்தல் வேண்டும்.

19. (i) வரிசை முறை - Sequential
(ii) மாற்று அல்லது கிளைப் பிரிப்பு - Alternative or Branching.
(iii) சுழற்சி அல்லது மடக்கு – Iterative or Looping

20. **செயற்கூறுவின் முதன்மை நன்மைகள் :**
(i) குறிமுறையை மீண்டும் எழுதுவதை தவிர்த்து குறிமுறையின் மறு பயனாக்கத்திற்கு உதவுகிறது.
(ii) நமது பயன்பாட்டிற்குச் சிறந்த கூறுநிலையை வழங்குகிறது.

21. தரவு நிலைத்தன்மை என்பது தரவுத்தளத்தில் அனைத்து இடங்களிலும் ஒத்த மதிப்புடைய தரவுகளாகும்.

22.

நெடுவரிசை கட்டுப்பாடு	அட்டவணை கட்டுப்பாடு
தனிப்பட்ட நெடுவரிசைக்கு மட்டுமே நெடுவரிசை கட்டுப்பாட்டை பயன்படுத்த முடியும்.	அட்டவணை கட்டுப்பாடு ஒன்று அல்லது அதற்கு மேற்பட்ட நெடுவரிசைகளுக்கு பயன்படுத்த முடியும்.

23. **MAX( ) மற்றும் MIN( ) செயற்கூறுகள்:**
(i) The MAX( ) சார்பு தேர்ந்தெடுக்கப்பட்ட நெடுவரிசையில் உள்ள மிகப்பெரிய மதிப்பை விடையாகக் கொடுக்கும்.
(ii) The MIN( ) சார்பு தேர்ந்தெடுக்கப்பட்ட நெடுவரிசையில் உள்ள மிகச்சிறிய மதிப்பை விடையாகக் கொடுக்கும்.

24. (i) பைத்தானில், Set என்பது தரவின் தொகுப்பின் மற்றொரு வகையாகும். Set என்பது மாறக்கூடிய மற்றும் நகல்கள் இல்லாத வரிசைப்படுத்தப்படாத உறுப்புகளின் தொகுப்பாகும்.
(ii) அதாவது, Set-ல் உள்ள உறுப்புகள் மீண்டும் இடம்பெற முடியாது. இந்த சிறப்பியல்பு உறுப்பு சோதனையை சேர்க்கவும் மற்றும் நகல் உறுப்புகளை நீக்கவும் பயன்படுகிறது.

enquiry@surabooks.com Kindly send me your study materials to padasalai.net@gmail.com Ph: 8124201000 / 8124301000

https://www.Padasalai.Net https://www.surabooks.com
https://www.CBSEtips.in https://t.me/tkp/pUm0

பகுதி - III

25. **தொகுதி பண்பியல்கள் பின்வருமாறு :**
   (i) தொகுதிகள் தரவு, தகவல் மற்றும் தருக்க செயலாக்கத்தைக் கொண்டுள்ளன.
   (ii) தொகுதிகள் தனியாக தொகுக்கப்பட்டு நூலகத்தில் சேமிக்கப்படும்.
   (iii) தொகுதிகள் நிரலில் சேர்க்க முடியும்.

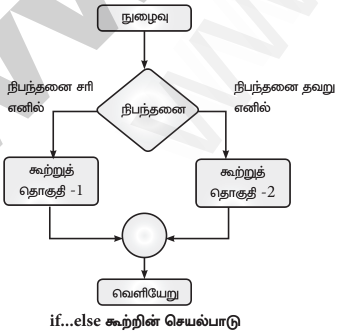
26. Asymptotic குறியீடுகள் நேரம் மற்றும் இடச்சிக்கல்களைப் பற்றிய அர்த்தமுள்ள கூற்றுகளைப் பயன்படுத்தும் ஒரு மொழியாகும். பின்வரும் மூன்று Asymptotic குறியீடுகள் நெறிமுறையில் நேரச் சிக்கலைக் குறிக்க மிகவும் பயன்படுகிறது.
   (i) Big O : நெறிமுறையின் மிக மோசமான நிலையை விவரிக்க Big O பெரும்பாலும் பயன்படுத்தப்படுகிறது.
   (ii) Big Ω: Big Omega, Big O-வின் தலைகீழ் ஆகும். Big O-asymptotic (மோசமான நிலையில்) செயற்கூறின் உச்ச வரம்பையும், Big Omega அதன் கீழ்வரையை குறிக்கும் (சிறந்த நிலையில்).
   (iii) Big Θ: நெறிமுறையானது கீழ் எல்லை = மேல் எல்லை என்னும் சிக்கலைக் கொண்டிருந்தால், **உதாரணத்திற்கு** O (n log n) மற்றும் Ω(n log n), ஆகிய சிக்கல்களைக் கொண்டுள்ளது என வைத்துக் கொள்வோம். உண்மையில் அதனுடைய சிக்கல் Θ (n log n), என்பது ஆகும். இதனுடைய அர்த்தம் என்னவென்றால் நெறிமுறையின் இயங்கு நேரம் மிகச் சிறந்த நிலை மற்றும் மிக மோசமான நிலை ஆகிய இரண்டு நிலையிலுமே எப்பொழுதும் n log n ஆக இருக்கும்.

27. பைத்தானில், சர நிலையுருக்கள், என்பது மேற்கோள் குறிக்குள் அடைக்கப்பட்ட குறியுருக்களின் தொடராகும். பைத்தானில், சரங்களை ஒற்றை, இரட்டை மற்றும் மூன்று மேற்கோள் குறிகளில் அடைக்கலாம். குறியுரு மதிப்பிருக்கள் ஒற்றை அல்லது இரட்டை மேற்கோள்குறியை கொண்டிருக்கும். பல வரி சர நிலையுருவை மூன்று மேற்கோள் ''' ''' குறிக்குள் கொடுக்கப்பட வேண்டும். பைத்தானில் ஒரு குறியுருவானது சர நிலையுருவாகவும் கருதப்படும்.

28. if..else கூற்றானது சரி தொகுதி மற்றும் தவறு தொகுதி இரண்டையுமே சரிபார்ப்பதற்கான கட்டுப்பாட்டை வழங்குகிறது. 'if..else' கூற்றின் தொடரியல்.

   தொடரியல் :

```
if <condition>:
        statements-block 1
        else:
        statements-block 2
```



if...else கூற்றின் செயல்பாடு

29. (i) செயற்கூறு திருப்பி அனுப்பும் மதிப்பு மற்றொரு செயற்கூறிற்கு செயலுருபாக, பின்னலான அமைப்பில் பயன்படுத்தினால் அதற்கு தொகுப்பு (composition) என்று பெயர்.
   (ii) எடுத்துக்காட்டாக, பயனரிடமிருந்து எண் மதிப்பை அல்லது கோவையை உள்ளீடாகப் பெற விரும்பினால், input() செயற்கூறு மூலம் பயனரிடமிருந்து சரத்தை உள்ளீடாகப் பெற்று eval() செயற்கூறு மூலம் அதன் மதிப்பை மதிப்பீடு செய்ய வேண்டும்.

30. (i) List என்பது வரிசைப்படுத்திய உறுப்புகளின் தொகுப்பாகும். ஆனால், Dictionary ஒரு உறுப்பை (திறவுகோல்) மற்றொரு உறுப்புடன் (மதிப்பு) பொருத்தப் பயன்படும் தரவு அமைப்பாகும்.
   (ii) List-ன் சுட்டெண்கள் குறிப்பிட்ட உறுப்பை அணுகுவதற்குப் பயன்படுகின்றன. ஆனால், Dictionary-ல் திறவுகோல் சுட்டெண்ணைக் குறிக்கிறது. ஒரு சரத்தின் எண்ணாகவும், திறவுகோல் இருக்கலாம் என்பதை நினைவில் கொள்க.
   (iii) சுட்டெண் மதிப்புகள் பார்க்க List-ன் மதிப்பை பயன்படுகிறது. Dictionary ஒரு மதிப்பை எடுத்துக் கொண்டு மற்றொரு மதிப்பை பார்க்க பயன்படுகிறது.

31. தரவுத்தள நிர்வாகி அல்லது DBA என்பவர் முழு தரவுத்தள மேலாண்மை அமைப்பையும் நிர்வகிப்பவர் ஆவார். இவர் DBMS பாதுகாப்பு, உரிமங்களை நிர்வகித்தல், பயனர் கணக்குகள் மற்றும் அணுகல்களை நிர்வகித்தல் போன்றவற்றை கவனித்துக் கொள்கிறார்.

32.

	பைத்தான்	C++
1.	பைத்தான் ஒரு 'வரி மொழி மாற்றி' (Interpreter) மொழியாகும்.	C++ மொழி, ஒரு நிரல்பெயர்ப்பி (Complier) மொழியாகும்.
2.	பைத்தான் மாறும் தன்மைக் கொண்ட மொழியாகும்.	C++ நிலையான தன்மைக் கொண்ட தொகுக்கப்பட்ட மொழி.
3.	மாறியை அறிவிக்கும் போது, அது சார்ந்த தரவின வகையை குறிப்பிடத் தேவையில்லை	மாறியை அறிவிக்கும் போது, தரவினத்தை குறிப்பிட வேண்டும்.
4.	இது Scripting மற்றும் பொதுப் பயன் மொழியென இருவகையிலும் செயல்பட முடியும்.	இது ஒரு பொதுப்பயன் நிரலாக்க மொழியாகும்.

33.
```
y = int (input ("Enter year"))
    if y % 4 = = 0 :
        print ("The given year is a leap year")
    else :
        print ("The given year is not a leap year")
```

வெளியீடு :
Enter the year 2012
Leap year

பகுதி - IV

34. அ) **இருமத் தேடல் :** இருமத் தேடலை பாதி இடைவெளித் தேடல் நெறிமுறை என்றும் அழைக்கலாம். வரிசைப்படுத்தப்பட்ட அணிக்குள் இலக்கு மதிப்பின் இருப்பிடத்தைக் கண்டுபிடிக்கிறது. பிரித்து-கைப்பற்றுதல் நெறிமுறையைப் போல் இருமத் தேடலைச் செய்து மடக்கை நேரத்தில் நிறைவேற்றப்படும்.

www.surabooks.com
enquiry@surabooks.com Ph: 8124201000 / 8124301000
Kindly send me your study materials to padasalai.net@gmail.com

www.Padasalai.Net www.CBSEtips.in

இருமத் தேடலுக்கான வழிமுறை :

1. மைய உறுப்பிலிருந்து தொடங்கவும் :

(i) இலக்கு மதிப்பும் அணியின் மைய உறுப்பும் நிகர் எனில் (அதாவது, மைய இலக்கு = உறுப்புகளின் எண்ணிக்கை /2) மைய உறுப்பின் சுட்டெண்ணைத் திருப்பி அனுப்பும்.

(ii) நிகரில்லை என்றால், மைய உறுப்பை மதிப்போடு ஒப்பிடவும்,

(iii) மைய சுட்டெண்ணிலுள்ள எண் இலக்கு மதிப்பை விட பெரியது எனில், மைய சுட்டெண்ணிற்கு வலப்புறம் உள்ள உறுப்புகளைத் தேர்ந்தெடுத்து படிநிலை -1லிருந்து தொடங்கவும்.

(iv) மைய சுட்டெண்ணிலுள்ள எண் இலக்கு மதிப்பை விடசிறியதுஎனில்மையசுட்டெண்ணிற்குஇடப்புறம் உள்ள உறுப்புகளைத் தேர்ந்தெடுத்து படிநிலை -1 லிருந்து தொடங்கவும்.

2. பொருத்தமான தேடல் கண்டுபிடிக்கப்பட்டால், பொருந்திய உறுப்பின் சுட்டெண்ணைத் திருப்பி அனுப்பும்.

3. பொருத்தம் இல்லையெனில், -1 என்ற மதிப்பைத் திருப்பி அனுப்பும் அல்லது தேடல் நிறைவேற்றப்படவில்லை என்ற தகவலை அறிவிக்கவும்.

இருமத் தேடல் இயங்கும் கோட்பாடுகள் :

(i) இருமத்தேடலில் பயன்படும் அணி வரிசையாக்கம் செய்யப்பட்ட அணியாகயிருக்க வேண்டும். இருமத் தேடலைப் பயன்படுத்தி மதிப்பு 60-ன் இருப்பிடத்தைத் தேடுவதாக எடுத்துக்கொள்வோம்.

10	20	30	40	50	60	70	80	90	99
0	1	2	3	4	5	6	7	8	9

(ii) முதலில் நாம் அணியின் மைய உறுப்பை mid = low + (high - low) / 2 என்ற வாய்ப்பாட்டைப் பயன்படுத்தித் தீர்மானிக்க வேண்டும். இங்கு, 0 + (9 - 0) / 2 = 4 (4.5 யின் முழு மதிப்பு எடுத்துக்கொள்ளவும்). அதனால் அணியின் மையம் 4 ஆகும்.

10	20	30	40	50	60	70	80	90	99
0	1	2	3	4	5	6	7	8	9

(iii) இப்பொழுது நாம் 4-ம் சுட்டெண் இருப்பிடத்தில் சேமிக்கப்பட்ட மதிப்போடு தேடப்படும் மதிப்பை (அதாவது, 60) ஒப்பிடு செய்வோம். 4-ம் சுட்டெண் இருப்பிடத்தில் உள்ள மதிப்பான 50 என்பது, இது தேடப்படும் மதிப்பு கிடையாது. தேடப்படும் மதிப்பானது 50-விட அதிகமாக இருப்பதால்

10	20	30	40	50	60	70	80	90	99
0	1	2	3	4	5	6	7	8	9

(iv) low மதிப்பை m + 1 என மாற்றி புதிய mid மதிப்பை மறுபடியும் கண்டுபிடிக்க வேண்டும்.

low = mid + 1

mid = low + (high - low) / 2

(v) இப்பொழுது நமது mid மதிப்பு 7 ஆகும். நாம் இருப்பிடம் 7-ல் சேமிக்கப்பட்ட மதிப்பை இலக்கு மதிப்போடு (அதாவது 60) ஒப்பிடுவோம்.

10	20	30	40	50	60	70	80	90	99
0	1	2	3	4	5	6	7	8	9

(vi) இருப்பிடம் 7ல் சேமிக்கப்பட்ட மதிப்பு தேடப்படும் மதிப்பு கிடையாது. மாறாக, நாம் தேடுவதை விட அதிகமான மதிப்பாக இருக்கிறது. எனவே, தேடப்படும் மதிப்பு இந்த இருப்பிடத்தை விட குறைவான பகுதியில் இருக்க வேண்டும்.

10	20	30	40	50	60	70	80	90	99
0	1	2	3	4	5	6	7	8	9

(vii) எனவே, நாம் mid மதிப்பைத் திரும்பவும் கணக்கீடு செய்ய வேண்டும்.

high = mid -1

mid = low + (high - low)/2

தற்பொழுது mid மதிப்பு 5 ஆகும்.

10	20	30	40	50	60	70	80	90	99
0	1	2	3	4	5	6	7	8	9

(viii) நாம் இருப்பிடம் 5-ல் உள்ள சேமிக்கப்பட்ட மதிப்பை இலக்கு மதிப்போடு ஒப்பீடு செய்வோம். இது ஒரு சரியான பொருத்தமாகும்.

10	20	30	40	50	60	70	80	90	99
0	1	2	3	4	5	6	7	8	9

(ix) இலக்கு மதிப்பு 60, இருப்பிடம் 5-ல் சேமிக்கப்பட்டுள்ளது என்று நாம் முடிவு செய்கிறோம். **எடுத்துக்காட்டாக,** இலக்கு மதிப்பு 95 எனில், இந்த செயல்முறை -1 என்ற மதிப்பைத் திருப்பி அனுப்பும்.

அல்லது

ஆ) தரவு அருவமாக்கினை செயல்படுத்த, ஆக்கிகள் (Constructor) மற்றும் செலக்டர் (Selectors) என்ற இரண்டு செயற்கூறுகள் உருவாக்கப்பட வேண்டும்.

ஆக்கிகள் மற்றும் செலக்டர்கள் (constructors and selectors) :

(i) ஆக்கி செயற்கூறுகள் அருவமாக்கம் தரவு வகையை கட்டமைக்க பயன்படுகிறது. செலக்டர் செயற்கூறுகள் தகவல்களை தரவு வகையிலிருந்து பெறுவதற்கு பயன்படுகிறது.

(ii) எடுத்துக்காட்டாக, city என்று ஒரு அருவமாக்க தரவு வகை உள்ளது என வைத்துக்கொள். City என்ற பொருள் நகரத்தின் பெயர், அட்சரேகை மற்றும் தீர்க்கரேகை பற்றிய தகவல்களை சேமித்திருக்கும் city என்ற பொருளை உருவாக்க பின்வரும் செயற்கூற்றினை பயன்படுத்தலாம்.

city := makecity (name, lat, lon)

(iii) city பொருளின் தகவல்களை பெறுவதற்கு பின்வரும் செயற்கூறுகளை பயன்படுத்தலாம்.

* getname(city)
* getlat(city)
* getlon(city)

(iv) பின்வரும் போலி குறிமுறை இரு நகரங்களுக்கு இடையேயான தொலைதூரத்தை கணக்கிடும்.

distance(city1, city2):

lt1, lg1 := getlat(city1), getlon(city1)

lt2, lg2:= getlat (city2), getlon (city2)

return ((lt1 – lt2)**2 + (lg1 – lg2)**2))$^{1/2}$

enquiry@surabooks.com Kindly send me your study materials to padasalai.net@gmail.com Ph: 8124201000 / 8124301000

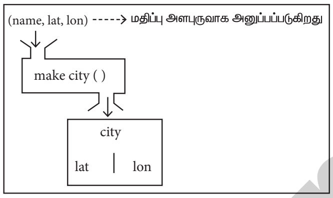
www.Padasalai.Net https://www.surabooks.com www.CBSEtips.in https://bit.ly/...Um0

(v) மேலே காணும் குறிமுறையில், distance(), getlat() மற்றும் getlon() ஆகியவை செயற்கூறுகள் ஆகும். lt என்பது அட்ச ரேசை மற்றும் lg என்பது தீர்க்கரேகையும் குறிக்கிறது. longitude. := என்பதை "assigned as" அல்லது "becomes" என்று வாசிக்க வேண்டும்.
It 1, lg1 := getlat(city1), getlon(city1)

(vi) இதனை lt1 என்பது getlat(city1)ன் மதிப்பாகிறது மற்றும் lg1 என்பது getlont(city1) ன் மதிப்பாகிறது. என்று வாசிக்க வேண்டும்.

(vii) இந்த செயற்கூறுகள் எவ்வாறு செயல்படுத்தப்படுகிறது என்பதை தெரிந்து கொள்ள வேண்டியதில்லை. இதை வேறு ஒருவர் நமக்காக வரையறுத்துள்ளார் என்று கருதிக் கொள்ள வேண்டும். பயனர் செயற்கூறுகள் எப்படி செயல்படுத்தப்படுகின்றது என்று தெரிந்து வைத்திருக்க வேண்டியதில்லை எனினும் வேறு ஒருவரால் இச்செயற்கூறுகள் வரையறுக்கப்பட்டிருக்க வேண்டும்.

(viii) மேலே காணும் குறிமுறையில், ஆக்கி மற்றும் செலக்டர்களை அடையாளம் காண்போம். ஆக்கி செயற்கூறுகள் அருவமாக்க தரவு வகையை கட்டமைக்கப் பயன்படுகிறது என்பதை நாம் அறிவோம். மேலே காணும் போலிக் குறிமுறையில், cityயின் பொருளை உருவாக்கும் செயற்கூறு, ஆக்கி ஆகும்.
city := makecity(name, lat, lon) இங்கு makecity(name, lat, lon) என்ற ஆக்கி city எனும் பொருளை உருவாக்குகிறது.



constructor

(ix) செலக்டர் செயற்கூறுகள் தகவல்களை தரவு வகையிலிருந்து பெறுவதற்கு பயன்படுகிறது. மேலே காணும் குறிமுனையில்,
* getname(city) * getlat(city)
* getlon(city) என்பவை city எனும் பொருளிலிருந்து தகவல்களை பெற்றுத் தரும் செலக்டர் செயற்கூறுகளாகும்.

35 அ) **உள்ளீடு மற்றும் வெளியீடு செயற்கூறுகள் :** ஒரு நிரல் பயனர் விரும்பிய பணியை நிறைவேற்றுவதற்கு பயனருடன் தொடர்பு கொள்ள வேண்டும். இதற்கு உள்ளீட்டு - வெளியீட்டு செயற்கூறுகள் பயன்படுகிறது. input() செயற்கூறு, ஓர் நிரலை இயக்கும் பொழுது தரவுகளை உள்ளீடு செய்யவும், print() என்ற வெளியீட்டு செயற்கூறு நிரலின் தீர்வுகளை திரையில் காண்பிக்க உதவுகிறது.

Print() செயற்கூறு : பைத்தானில், print() செயற்கூறு நிரலை இயக்கும் பொழுது தரவுகளை வெளியிட பயன்படுகிறது. print() செயற்கூறின் தொடரியல் கீழே கொடுக்கப்பட்டுள்ளது

```
print ("string to be displayed as output")
print (variable)
print ("String to be displayed as output", variable)
print ("String1", variable, "String 2", variable, "String 3" .......)
```

எடுத்துக்காட்டு:

```
>>> print ("Welcome to Python Programming")
    Welcome to Python Programming
>>> x = 5
>>> y = 6
>>> z = x + y
>>> print (z)
    11
>>> print ("The sum = ", z)
    The sum = 11
>>> print ("The sum of ", x, "and", y, "is", z)
    The sum of 5 and 6 is 11
```

வெளியீடு:

```
Welcome to Python programming
11
The sum = 11
The sum of 5 and 6 is 11
```

print() செயற்கூறு தீர்வுகளை திரையில் காண்பிக்கும் முன் கோவையை மதிப்பீடு செய்யும். print() செயற்கூறினுள் கொடுக்கப்படும் முழு கூற்றினையும் திரையில் காட்டும். காற்புள்ளி(,) கொண்டு print() செயற்கூறினுள் ஒன்றுக்கு மேற்பட்ட உருப்புகளை பிரிக்கலாம்.

input () செயற்கூறு : பைத்தானில், input() செயற்கூறு நிரலை இயக்கும் பொழுது தரவுகளை உள்ளீடாக பெற்றுக் கொள்ளப் பயன்படுகிறது. input() செயற்கூறின் தொடரியல்,

Variable = input ("prompt string")

இந்த தொடரியலில் உள்ள "prompt string" பயனர் கொடுக்கப்பட வேண்டிய உள்ளீடு எதுவென்பதை உணர்த்தும் கூற்று அல்லது செய்தியை கொண்டிருக்கும்.

input() கட்டளை அமைப்பில் உள்ள "prompt string" திரையில் தோன்றும், பயனர் இதை படித்து உள்ளீட்டு சாதனத்திலிருந்து எதிர்பார்க்கப்படும் தரவுகளை கொடுக்கலாம். input() செயற்கூறு, விசைப்பலகையின் மூலம் தட்டச்சு செய்தவற்றை மாறியில் சேமித்து வைக்கும். "prompt string" கொடுக்கப்படவில்லையெனில் திரையில் எந்த தகவலும் தோன்றாது. இதனால் பயனர் எவற்றை உள்ளீடு செய்ய வேண்டும் என்பதை தெரிந்து கொள்ள முடியாது.

எடுத்துக்காட்டு 1: "prompt string" கொண்டுள்ள input() செயற்கூறு

```
>>> city=input ("Enter Your City:")
        Enter Your City: Madurai
>>> print ("I am from", city)
        I am from Madurai
```

எடுத்துக்காட்டு 2:
"prompt string" இடம்பெறாத input( ) செயற்கூறு

```
>>> city=input()
        Rajarajan
>>> print (I am from", city)
        I am from Rajarajan
```

மேலே உள்ள எடுத்துக்காட்டு 2ல், input() செயற்கூற்றில் "prompt string" இடம்பெறவில்லை. எனவே, எவற்றை உள்ளீடாக தர வேண்டும் என்பது பயனர் புரிந்து கொள்ள இயலாது. நிரலுக்கு தொடர்பில்லாத தரவுகளை உள்ளிட்டால், தவறான வெளியீட்டை வெளியிடும்.

எனவே பயனர் நிரலை உரிய முறையில் இயக்க, "prompt string" input() செயற்கூறில் பயன்படுத்த வேண்டும்.

input() செயற்கூறு அனைத்து தரவுகளையும் சரங்களாக ஏற்றுக் கொள்ளும். பயனர் எண்களை மதிப்புகளாக உள்ளீடு செய்யும் போது, அவை வெளிப்படையாக எண்வகை தரவுகளாக மாற்றப்பட வேண்டும். int() செயற்கூறு சரவகை தரவை எண்வகை தரவாக வெளிப்படையாக மாற்ற உதவுகிறது.

அல்லது

enquiry@surabooks.com Kindly send me your study materials to padasalai.net@gmail.com Ph: 8124201000 / 8124301000

https://www.Padasalai.Net https://www.CBSEtips.in

ஆ) மாறியின் வரையெல்லை என்பது நிரலின் அணுகக்கூடிய பகுதியைக் குறிப்பதாகும். அதாவது, எந்த பகுதியில் மாறியைப் பயன்படுத்துகிறோமோ அதைக் குறிக்கிறது. வரையெல்லை நடப்பு மாறித் தொகுதிகள் மற்றும் அதன் மதிப்புகளைக் கொண்டிருக்கும். மாறியின் வரையெல்லைகளை உள்ளமை வரையெல்லை மற்றும் முழுதளாவிய வரையெல்லை என இரண்டு வகையாகப் பிரிக்கலாம்.

உள்ளமை வரையெல்லை (Local Scope) : ஒரு செயற்கூறுவின் உடற்பகுதியின் உள்ளே அறிவிக்கப்படும் மாறி உள்ளமை மாறி எனப்படும்.

உள்ளமை மாறியின் விதிமுறைகள்

(i) உள்ளமை மாறியின் வரையெல்லை அது வரையறுக்கப்பட்டுள்ள தொகுதிக்குள் மட்டுமே பயன்படுத்த முடியும்.
(ii) செயற்கூறினுள் மாறி உருவாக்கப்படும் போது அது உள்ளமைவாக அமையும்.
(iii) செயற்கூறு இயக்கப்படும் போது மட்டுமே உள்ளமை மாறிகள் உயிர் பெறும்.
(iv) முறையான அளபுருக்கள் அதன் செயற்கூறுக்கு உள்ளமையானது ஆகும்.

எடுத்துக்காட்டு : உள்ளமை மாறியை உருவாக்குதல்

```
def loc():
y=0 # local scope
print(y)
loc()
```

வெளியீடு :

0

முழுதளாவிய வரையெல்லை (Global Scope) : அனைத்து செயற்கூறுகளுக்கும் மேலாக அறிவிக்கப்படும் மாறி முழுதளாவிய மாறி எனப்படும். முழுதளாவிய வரையெல்லை உடைய மாறியை நிரலின் எந்த பகுதியிலும் அணுகமுடியும் முழுதளாவிய வரையெல்லை கொண்ட மாறியை ஒரு செயற்கூறினுள் அணுகுவதற்க்கு 'global' சிறப்புச் சொல்லானது பயன்படுகிறது.

பைத்தானின் 'Global' சிறப்புச் சொல்லின் அடிப்படை விதிகள் :

(i) செயற்கூறுக்கு வெளியே மாறியை அறிவிக்கும் போது அது தானமைவாக முழுதளாவிய வரையெல்லை ஆகும். `global' என்ற சிறப்புச் சொல்லை பயன்படுத்த வேண்டியதில்லை.
(ii) செயற்கூறினுள் முழுதளாவிய மாறியின் மதிப்பை மாற்றியமைக்க `global' என்ற சிறப்புச்சொல் பயன்படுத்த வேண்டும்.
(iii) செயற்கூறுவிற்கு வெளியே `global' என்ற சிறப்புச் சொல் எந்த விளைவையும் ஏற்படுத்தாது.

'Global' சிறப்புச் சொல்லின் பயன்பாடு :

எடுத்துக்காட்டு : செயற்கூறுவின் உள்ளிருந்து முழுதளாவிய மாறியை அணுகுதல்

```
c = 1          # global variable
def add():
        print(c)
        add()
```

வெளியீடு :

1

36. அ) பைத்தான் பின்வரும் செயற்குறிகளை வழங்குகிறது. இச்செயற்குறிகள் சரங்களை கையாள உதவுகிறது.

(i) **இணைப்பு (Concatenation +) :** இரண்டு அல்லது அதற்கு மேற்பட்ட சரங்களை இணைக்கும் செயல்பாடு சேர்த்தல் அல்லது இணைத்தல் எனப்படும். கூட்டல் செயற்குறி சரங்களை இணைக்கப் பயன்படுகிறது.

எடுத்துக்காட்டு :

```
>>> "welcome" + "Python"
'welcomePython'
```

(ii) **சேர்த்தல் (Append +=) :** ஏற்கனவே உள்ள சரத்தின் இறுதியில் மேலும் புதிய சரம் அல்லது சரங்களை சேர்க்கும் செயல் சேர்த்தல் எனப்படும். += செயற்குறி ஏற்கனவே உள்ள சரத்தின் இறுதியில் புதிய சரத்தினை சேர்க்க பயன்படுகிறது.

எடுத்துக்காட்டு :

```
>>> str1="Welcome to "
>>> str1+="Learn Python"
>>> print (str1)
Welcome to Learn Python
```

(iii) **பலமுறை (Repeating (*)) :** பெருக்கல் செயற்குறி கொடுக்கப்பட்ட சரத்தினை பல தடவைகள் வெளிப்படுத்த பயன்படுகிறது.

எடுத்துக்காட்டு :

```
>>> str1="Welcome "
>>> print (str1*4)
Welcome Welcome Welcome Welcome
```

அல்லது

ஆ) **List -ல் உறுப்புகளை சேர்த்தல் :**

(i) பைத்தானில், append () செயற்கூறு ஒரு உறுப்பையும் extend() செயற்கூறு ஒன்றுக்கும் மேற்பட்ட உறுப்புகளையும் ஏற்கனவே உள்ள List-ல் சேர்க்க பயன்படுகின்றது.

தொடரியல்:

```
List.append (element to be added)
List..extend ([elements to be added])
```

(ii) extend( ) செயற்கூறில், பல உறுப்புகளை சதுர அடைப்புக் குறிக்குள் செயற்கூறின் செயலுருபுகளைப் போலவே குறிப்பிட வேண்டும்.

எடுத்துக்காட்டு :

```
>>> MyList=[34, 45, 48]
>>> MyList.append(90)
>>> print(MyList)
[34, 45, 48, 90]
```

(iii) மேலே உள்ள எடுத்துக்காட்டில், My List என்பது மூன்று உறுப்புகளுடன் உருவாக்கப்பட்டுள்ளது. >>> MyList.append(90) என்ற கூற்றின் மூலம் 90 என்ற கூடுதல் மதிப்பு ஏற்கனவே உள்ள List-ல் கடைசி உறுப்பாக சேர்க்கப்படுகிறது. Print கூற்று MyList என்ற லிஸ்ட்-ல் உள்ள அனைத்து உறுப்புகளையும் காண்பிக்கிறது.

எடுத்துக்காட்டு :

```
>>> MyList.extend ([71, 32, 29])
>>> print(MyList)
[34, 45, 48, 90, 71, 32, 29]
```

(iv) மேலே உள்ள குறிமுறையில், extend( ) செயற்கூறு பல உறுப்புகளை சேர்ப்பதற்கு பயன்படுகிறது. Print கூற்று, கூடுதல் உறுப்புகள் சேர்க்கப்பட்ட பிறகு List-ல் உள்ள அனைத்து உறுப்புகளையும் காண்பிக்கிறது.

enquiry@surabooks.com Kindly send me your study materials to padasalai.net@gmail.com Ph:8124201000 / 8124301000

www.Padasalai.Net www.CBSEtips.in

List உறுப்புகளை செருகுதல் :

(i) பைத்தானில் உள்ள append( ) ஒரு உறுப்பை சேர்க்கப் பயன்படுகிறது. ஆனால், இது List-ன் இறுதியில் உறுப்பை சேர்க்கிறது.

(ii) நீங்கள் உங்களுக்கு விருப்பமான இடத்தில் ஒரு உறுப்பை சேர்க்க விரும்பினால், insert ( ) செயற்கூறை பயன்படுத்த வேண்டும். insert( ) செயற்கூறு, List-ன் எந்தவொரு இடத்திலும் ஒரு உறுப்பை சேர்க்கப் பயன்படுகிறது.

தொடரியல் : List.insert (position index, element)

எடுத்துக்காட்டு :

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

37. அ)

மிகைமை (தேவைக்கு அதிகமானவற்றை) குறைத்தல்	நவீன உலகில் வன்வட்டுகளின் விலை மிகவும் குறைவு. ஆனால் முந்தைய காலங்களில் வன்வட்டுகள் மிகவும் விலை உயர்ந்தவை, ஆகையால் தரவுதளத்தில் உள்ள தேவையற்ற தரவுகள் பெரிய சிக்கலாக இருந்தது. ஆனால் DBMS தரவுகளை பிரித்தாளும் இயல்பு நிலையைப் பின்பற்றி தேவையற்ற தரவுகள் மீண்டும் இடம்பெறுவதைக் குறைக்கிறது.
தரவுகளின் நிலைத்தன்மை	தொடர்ச்சியாக புதுப்பிக்கப்படும் மற்றும் பராமரிக்கப்படும் நேரடி தரவின் (Live data) நிலைத்தன்மையை பராமரிப்பது ஒரு சவாலாக மாறும். ஆனால் DBMS தானாகவே அதை கையாளுகிறது.
பல பயனர்கள் மற்றும் ஒத்த அணுகுதலை ஆதரித்தல்	DBMS பல பயனர்களை ஒரே நேரத்தில் ஒரு தரவின் மீது வேலை (தரவுகளைப் புதுப்பித்தல், செருகுதல், நீக்குதல்) செய்ய அனுமதிக்கிறது. இருப்பினும் தரவின் நிலைத்தன்மைப் பராமரிப்பை கையாளுகிறது.
வினவல் மொழி	DBMS ஒரு எளிய வினவல் மொழியை பயனருக்கு வழங்குகிறது. இதைப்பயன்படுத்தி தரவுத்தளத்திலிருந்து தரவுகளை எளிமையாக பெறவும், சேர்க்கவும், நீக்கவும், புதுப்பிக்கவும் முடியும்.
பாதுகாப்பு	DBMS, தரவின் பாதுகாப்பை கவனித்துக்கொள்கிறது. அங்கீகாரமற்ற அணுகுதலில் இருந்து தரவுகளுக்கு பாதுகாப்பை வழங்குகிறது. வழக்கமான DBMSல், பல்வேறு அணுகலுக்கான அனுமதியுடன் பயனர் கணக்குகளை நாம் உருவாக்கமுடியும். இதனைப் பயன்படுத்தி பயனர் அணுகுதலை கட்டுப்படுத்தி நம்முடைய தரவுகளை எளிமையாக பாதுகாத்துக் கொள்ள முடியும்.
DBMS பரிவர்த்தனைகளை ஆதரித்தல்	இது நிகழ் உலக பயன்பாடுகளில் தரவு ஒருமைப்பாட்டை கையாள மற்றும் நிர்வகிக்க நம்மை அனுமதிக்கிறது. இங்கு பல்புரியாக்கம் (mutithreading) விரிவாக பயன்படுகிறது.

அல்லது

ஆ) (i) SQLite என்பது எளிய உறவுநிலை தரவுத்தள அமைப்பாகும். இது தரவுகளை வழக்கமான தரவுக்கோப்புகளாக கணினியின் உட்புற நினைவகத்தில் சேமித்து வைக்கும்.

(ii) பயன்பாடுகளில் தனியாக MySQL மற்றும் Oracle போன்ற சேவையக தரவுத்தளங்களை பயன்படுத்துவதற்கு பதிலாக இது பயன்பாடுகளுடன் ஒன்றிணைந்து செயல்படும் வகையில் வடிவமைக்கப்பட்டுள்ளது.

(iii) வேகமாகவும், மிகுந்த சோதிக்கப்பட்டதாகவும் மற்றும் நெகிழ்வானதாகவும் உள்ளதால் SQLite-ல் வேலை செய்வது எளிதாகும். SQLite ற்காக பைத்தான் உள்ளிணைந்த நூலகத்தைக் கொண்டுள்ளது. SQLite யைப் பயன்படுத்த

(iv) **படிநிலை 1:** sqlite 3 இணைக்கவும்.

(v) **படிநிலை 2:** connect ( ) வழிமுறையைப் பயன்படுத்தி இணைப்பை உருவாக்கி தரவுத்தளத்தின் பெயரை இணைக்கவும்.

(vi) **படிநிலை 3:** cursor=connection.cursor() என்றக் கூற்றைப் பயன்படுத்தி cursor என்னும் பொருளை அனுப்பவும்.

(vii) படிநிலை 2 ல் தரவுத்தளத்தை இணைத்தல் என்பது, அணுக வேண்டிய தரவுத்தளத்தின் பெயரைஅனுப்புதல் என்பதாகும். அவ்வாறு அனுப்பும்போது, அத்தரவுத்தளம் ஏற்கனவே இருக்குமாயின் அது இணைக்கப்படும். இல்லாவிடில், பைத்தான், கொடுக்கப்பட்ட பெயரில் ஒரு புதிய தரவுத்தளத்தை உருவாக்கும்.

(viii) படிநிலை3ல் cursor என்பது ஒரு கட்டுப்பாட்டு அமைப்பாகும், இது தரவுத்தளப் பதிவுகளை இணைக்கப் பயன்படுகிறது.

(ix) பைத்தானில், cursor மிக முக்கிய பங்கு வகிக்கின்றது. அனைத்து கட்டளைகளும் cursor பொருள் மூலம் மட்டுமே இயக்கப்படும்.

(x) ஒரு தரவுத்தளத்தில் அட்டவணையை உருவாக்க, ஒரு பொருளை உருவாக்கி அதற்கான SQL கட்டளைகளை எழுத வேண்டும். எடுத்துக்காட்டு:- sql_comm = "SQL statement".

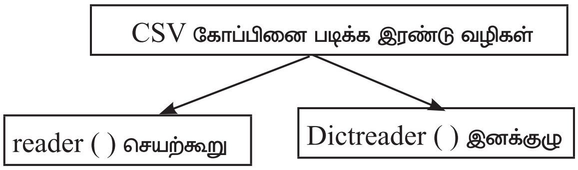
(xi) கட்டளைகளை நிறைவேற்ற cursor வழிமுறையைப் பயன்படுத்தி sql கட்டளையின் பெயரை ஒன்றன் பின் ஒன்றாக அளபுருக்களாக அனுப்ப வேண்டும். sql_commல் பல கட்டளைகள் சேமிக்கப்பட்டு அவை நிறைவேற்றப்படும். அனைத்து செயல்களையும் முடித்த பின்னர் மாற்றங்களைக் கோப்பில் சேமித்து பின்னர் இணைப்பை மூட வேண்டும்.

Kindly send me your study materials to padasalai.net@gmail.com

www.Padasalai.Net www.CBSEtips.in

38 அ) **பைத்தான் பயன்படுத்தி CSV கோப்பினை படித்தல் :** CSV கோப்பினை படிக்க இரண்டு வழி முறைகள் உள்ளன.

(i) CSV தொகுதி படித்தல் செயற்கூறை பயன்படுத்துல்.
(ii) DictReader இனக்குழுவை பயன்படுத்துதல்.



CSV -ன் Reader செயற்கூறு :

(i) CSV கோப்பின் உள்ளடக்கத்தை படிக்க csv.reader() என்ற செயற்கூறானது பயன்படுத்தப்படுகிறது.
(ii) reader() செயற்கூறானது கோப்பின் ஒவ்வொரு வரியையும் படித்து அவற்றை நெடுவரிசைகளின் பட்டியலாக (List) அமைக்கும்.
(iii) மாறியின் தரவிற்கு தேவையான நெடுவரிசையை தேர்வு செய்யலாம். இச்செயற்கூறை பயன்படுத்தி பயனர் கோப்பின் தரவுகளில் உள்ள இரட்டை மேற்கோள் குறி (" "), (|) மற்றும் (,) போன்ற பல்வேறு வடிவமைப்புகளை பயன்படுத்தி படிக்கலாம்.

csv.reader() செயற்கூறின் தொடரியல்
csv.reader(fileobject, delimiter, fmtparams)
இங்கு,

(iv) **file object :** கோப்பின் பாதையையும் முறைமையும் திருப்பி அனுப்பும்.
(v) **delimiter :** இது விருப்பத்தேர்வு அளபுருவாகும். இது செந்தர வரம்புக் குறிகளை கொண்டிருக்கும். (',|) போன்றவை மற்றவைகள் தவிர்க்கப்படும்.
(vi) **fmtparams :** கொடாநிலை மதிப்புகளை மேற்பதிப்பு செய்ய/நீக்க பயன்படும் விருப்பத் தேர்வு அளபுருவாகும்.
  (i) CSV கோப்பு – கொடாநிலை பிரிப்பானான காற்புள்ளியை கொண்ட தரவு.
  (ii) CSV கோப்பு – தொடக்கத்தில் இடம் கொண்ட தரவுகள்
  (iii) CSV கோப்பு – மேற்கோளுடன் கூடிய தரவுகள்
  (iv) CSV கோப்பு – தனிப்பயன் பிரிப்பான கொண்ட தரவு.

கொடாநிலை பிரிப்பான் காற்புள்ளியுடன் கூடிய CSV (,) கோப்புகள் : பின்வரும் நிரலானது கொடாநிலை பிரிப்பானான காற்புள்ளியுடன் கூடிய "sample1.csv" என்ற கோப்பினை படித்து ஒவ்வொரு வரிசையாக அச்சிடும் நிரல்.

```
#importing csv
import csv
#opening the csv file which is in different location with
read mode
with open('c:\pyprg\sample1.csv', 'r') as F:
#other way to open the file is f= ('c:\pyprg\sample1.csv', 'r')
        reader = csv.reader(F)
        for row in reader :
# printing each line of the Data row by row
        print(row)
F.close()
```

வெளியீடு

```
['SNO', 'NAME', 'CITY']
['12101', 'RAM', 'CHENNAI']
['12102', 'LAVANYA', 'TIRUCHY']
['12103', 'LAKSHMAN', 'MADURAI']
```

CSV கோப்பினை Dictionary படிக்க:

(i) CSV செயற்கூறிலுள்ள DictReader ஐ பயன்படுத்தி CSV கோப்பை படித்தல். இதன் செயல்பாடு reader() விதிமுறை செயல்பாட்டை ஒத்திருக்கும். ஆனால் இது ஒரு பொருளை உருவாக்கி அதை Dictionaryயில் இணைக்கும்.
(ii) DictReader CSV கோப்பில் உள்ள முதல் வரியை காற்புள்ளியை இந்த வரியின் Dictionary திறவுகோல் (Dictionary Key) பயன்படுத்தி படிக்கும். அடுத்தடுத்துள்ள வரிசையில் உள்ள நெடுவரிசையானது Dictionaryயின் மதிப்புகளாக செயல்பட்டு அவற்றை உரிய திறவுகோள் மூலம் அணுக முடியும் (புலத்தின் பெயர்).
(iii) CSV கோப்பின் முதல் வரிசையானது புலப்பெயரை கொண்டிருக்கவில்லையெனில் புலப்பெயர் அளபுருவை DictReader's ஆக்கிக்கு அனுப்பி Dictionary திறவு கோல்களை நாமாக ஒதுக்க முடியும்.
(iv) csv.reader() மற்றும் DictReader() க்கு இடையேயான முக்கிய வேறுபாட்டை எளிமையாக கூறுவதெனில் csv.reader மற்றும் csv.writer ஆனது பட்டியல் (list/tuple) பதிவுடன் வேலை செய்யும். csv.DictReader மற்றும் csv.DictWriter ஆனது அகராதியில் வேலை செய்யும்.

```
import csv
filename = `c:\pyprg\sample8.csv'
input_file =csv.DictReader(open(filename,'r'))
for row in input_file:
        print(dict(row))        #dict() to print data
```

வெளியீடு :

```
{`ItemName `: `Keyboard `, `Quantity': `48'}
{`ItemName `: `Monitor', `Quantity': `52'}
{`ItemName `: `Mouse `, `Quantity': `20'}
```

(v) csv.DictReader மற்றும் csv.DictWriter இரண்டும் கூடுதல் அளபுருவாக புலப்பெயரினை பெற்று Dictionary திறவுகோளாக பயன்படுத்தும்.

(எடுத்துக்காட்டாக) "sample8.csv" என்ற கோப்பானது Dictionary –யை படித்தல்.

(vi) **செயல்முறை:** மேலே கொடுக்கப்பட்டுள்ள நிரலில் dict() செயல்கூறில் நீக்கவும் மற்றும் print(row) பயன்படுத்தவும். கீழே கொடுக்கப்பட்டுள்ள வெளியீடை தரும்.

```
OrderedDict [(`ItemName`,`Keyboard`), (`Quantity', `48')]
OrderedDict [(`ItemName`,`Monitor'), (`Quantity', `52')]
OrderedDict [(`ItemName`,`Mouse`), (`Quantity', `20')]
```

அல்லது

ஆ) (i) பைத்தான் பயனிலா (garbage) மதிப்புகளைச் சேகரிக்கும் தானியங்கியைப் பயன்படுத்துகிறது. இந்த பண்புக்கூறு C++ல் கிடையாது.
(ii) C++ நிலையான வகையைச் சார்ந்த மொழி, ஆனால் பைத்தான் ஒரு மாறக்கூடிய வகையைச் சார்ந்த மொழியாகும்.
(iii) பைத்தான் வரி மொழி மாற்றி மூலம் இயங்குகிறது. ஆனால் C++ நிரல் பெயர்ப்பியின் மூலம் தொகுக்கப்பட்டு இயக்கப்படுகிறது.
(iv) C++ நிரல் குறிமுறையைக் காட்டிலும் பைத்தான் குறிமுறை 5லிருந்து 10 மடங்கு குறைவானது.
(v) பைத்தானில், வெளிப்படையாக தரவினங்களை அறிவிக்க தேவையில்லை.

☆☆☆

Kindly send me your study materials to padasalai.net@gmail.com