සියලු ම හිමිකම් ඇවිරිණි /All Rights Reserved]

Ravindu Bandaranayake - ictfromabc

Information and Communication Technology
තොරතුරු හා සන්නිවේදන තාක්ෂණය

2026 QUIZ NO 390 MARKING

2026/01/04
ictfromabc
Ravindu Bandaranayake

1. What major advancement characterizes the Fourth Generation of computers, and what impact did it have on computing?
හතරවන පරම්පරාවේ පරිගණකවල ඇති ප්‍රධාන දියුණුව කුමක්ද සහ එය පරිගණනයට ඇති කළ බලපෑම කුමක්ද?

    1) Vacuum Tubes: significantly increased computing power.
    රික්තක නළ: පරිගණක බලය සැලකිය යුතු ලෙස වැඩි කිරීම.

    2) Transistors: reduced size and heat generation.
    ට්‍රාන්සිස්ටර: ප්‍රමාණය සහ තාප උත්පාදනය අඩු කිරීම.

    3) Integrated Circuits: enabled miniaturization of computers.
    අනුකලිත පරිපථ: පරිගණක ප්‍රමාණයෙන් කුඩා කිරීම.

    4) Microprocessors: revolutionized personal computing and made computers more affordable.
    ක්ෂුද්‍ර සකසන: පුද්ගලික පරිගණකකරණයේ විප්ලවීය වෙනසක් ඇති කළ අතර පරිගණක වඩාත් ලාභදායී බවට පත් කළේය.

    5) Quantum Computing: introduced massive parallel processing capabilities.
    ක්වොන්ටම් පරිගණකකරණය: දැවැන්ත සමාන්තර සැකසුම් හැකියාවන් හඳුන්වා දුන්නේය.

Answer : 4)

The major advancement characterizing the Fourth Generation of computers is the development of **microprocessors**. Microprocessors are tiny integrated circuits that contain the entire processing unit of a computer. This advancement revolutionized personal computing by significantly reducing the size, cost, and power consumption of computers, making them more affordable and accessible to the general public. As a result, personal computers became widely available, leading to the rise of home computing and the proliferation of modern computing devices.
පරිගණකවල හතරවන පරම්පරාවේ ප්‍රධාන දියුණුව නම් **ක්ෂුද්‍ර සකසන** සංවර්ධනයයි. ක්ෂුද්‍ර සකසනය යනු පරිගණකයක සම්පූර්ණ සැකසුම් ඒකකයම අඩංගු කුඩා සංයුක්ත පරිපථ වේ. මෙම ප්‍රගතිය පෞද්ගලික පරිගණනයේ විප්ලවීය වෙනසක් ඇති කළේය එහි ප්‍රතිඵලයක් ලෙස පුද්ගලික පරිගණක බහුලව ලබා ගත හැකි වූ අතර එය ගෘහ පරිගණනයේ දියුණුවටත් නවීන පරිගණක උපාංග ව්‍යාප්ත වීමටත් හේතු විය.

Therefore, the correct answer is **4).**
එබැවින් නිවැරදි පිළිතුර **4)** වේ.

2. What is the 8 bit two's complement of 8_{10} and -11_{10}?
8_{10} සහ -11_{10} හි බිටු 8 දෙකේ අනුපූරකය කුමක්ද?

1) 00000110_2 and 10101010_2
2) 11100101_2 and 11101010_2
3) 01010101_2 and 00101010_2
4) 00001000_2 and 11110101_2
5) 10100101_2 and 11101010_2

Answer : 4)

First Let's find the 8 bit 2's Complement of 8							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	0	0	0	1	0	0	0
8 bit 2's Complement of 8 = 00001000_2							
Now Let's find the 8 bit 2's Complement of -11							
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	0	0	0	1	0	1	1
1	**1**	**1**	**1**	**0**	**1**	**0**	**0**
						+	**1**
1	1	1	1	0	1	0	1
8 bit 2's Complement of -11 = 11110101_2							

Therefore the correct answer is **4) 00001000 and 11110101**
එබැවින් නිවැරදි පිළිතුර **4) 00001000 and 11110101** වේ

3. Which two logic gates are used to implement a half-adder circuit?
අර්ධ ආකලක පරිපථයක් ක්‍රියාත්මක කිරීම සඳහා භාවිතා කරන තාර්කික ද්වාර දෙක මොනවා ද?

1) AND / OR gates
2) XOR / AND gates
3) OR / NOT gates
4) NAND / NOR gates
5) XOR / OR gates

Answer : 2)

A half-adder circuit performs the addition of two single-bit binary numbers and produces two outputs: the sum and the carry. The XOR gate is used to produce the sum output because it outputs a 1 only when exactly one of the inputs is 1, which matches the behavior of binary addition for the sum bit. The AND gate is used to produce the carry output because it outputs a 1 only when both inputs are 1, which represents the carry bit in binary addition.
අර්ධ ආකලක පරිපථයක් තනි-බිටි ද්විමය සංඛ්‍යා දෙකක් එකතු කිරීම සිදු කරන අතර ප්‍රතිදානයන් දෙකක් නිපදවයි: එකතුව සහ රැගෙන යාම. XOR ද්වාරය චෛක්‍ය ප්‍රතිදානය නිපදවීමට භාවිතා කරන්නේ එය 1 ප්‍රතිදානය කරන්නේ හරියටම ආදාන වලින් එකක් 1 වන විට පමණි, එය එකතුව බිටු සඳහා ද්විමය එකතු කිරීමේ හැසිරීමට ගැළපේ. AND ද්වාරය රැගෙන යාමේ ප්‍රතිදානය නිපදවීමට භාවිතා කරනුයේ එය 1 ප්‍රතිදානය කරන්නේ ආදාන දෙකම 1 වන විට පමණක් වන බැවිනි, එය ද්විමය එකතු කිරීමේදී රැගෙන යාමේ බිටි නියෝජනය කරයි.

Therefore, the correct answer is **2).**
එබැවින් නිවැරදි පිළිතුර **2)** වේ.

4. Which of the following describes a "Many-to-Many" relationship in a relational database?
සම්බන්ධතා දත්ත ගබඩාවක "බහු-බහු" සම්බන්ධතාවයක් විස්තර කරන්නේ පහත සඳහන් මොනවාද?

1) One record in a table can relate to multiple records in another table.
වගුවක ඇති එක් වාර්තාවක් තවත් වගුවක ඇති වාර්තා කිහිපයකට සම්බන්ධ විය හැක.
2) Each record in the first table is related to one and only one record in the second table.
පළමු වගුවේ ඇති සෑම වාර්තාවක්ම දෙවන වගුවේ එක් වාර්තාවකට පමණක් සම්බන්ධ වේ.
3) Each record in the first table can be related to many records in the same table.
පළමු වගුවේ ඇති සෑම වාර්තාවක්ම එකම වගුවේ ඇති බොහෝ වාර්තා වලට සම්බන්ධ විය හැක.
4) <u>Multiple records in the first table can relate to multiple records in the second table through a join table.</u>
පළමු වගුවේ ඇති බහු වාර්තා, එකතු කිරීමේ වගුවක් හරහා දෙවන වගුවේ බහු වාර්තා වලට සම්බන්ධ විය හැක.
5) A table can only have a one-to-one relationship with another table.
වගුවකට තවත් වගුවක් සමඟ ඒක-ඒක සම්බන්ධයක් පමණක් තිබිය හැක.

Answer : 4)

In a Many-to-Many relationship in a relational database:
සම්බන්ධතා දත්ත සමුදායක බහු-බහු සම්බන්ධතාවයකදී:

• Multiple records in one table can relate to multiple records in another table.
එක් වගුවක බහු වාර්තා තවත් වගුවක බහු වාර්තා සමඟ සම්බන්ධ විය හැකිය.

• This type of relationship is implemented using a join table (also called a junction table or associative table).
මෙම ආකාරයේ සම්බන්ධතාවය ක්‍රියාත්මක කරනු ලබන්නේ සම්බන්ධක වගුවක් (junction වගුවක් හෝ සහකාර වගුවක් ලෙසද හැඳින්වේ) භාවිතා කරමිනි.

Explanation of each option / එක් එක් විකල්පය පැහැදිලි කිරීම:

Option / විකල්ප 1)
This describes a One-to-Many relationship.
මෙය ඒක-බහු සම්බන්ධතාවයක් විස්තර කරයි.

Option / විකල්ප 2)
This describes a One-to-One relationship.
මෙය ඒක-ඒක සම්බන්ධතාවයක් විස්තර කරයි.

Option / විකල්ප 3)
This describes a Self-Referencing or Recursive relationship.
මෙය ස්වයං-යොමු කිරීම හෝ පුනරාවර්තන සම්බන්ධතාවයක් විස්තර කරයි.

Option / විකල්ප 4)
This describes a Many-to-Many relationship.
මෙය බහු-බහු සම්බන්ධතාවයක් විස්තර කරයි.

Option / විකල්ප 5)
Incorrect, as databases can support multiple types of relationships (One-to-One, One-to-Many, Many-to-Many). This statement limits the relationship types unnecessarily.
දත්ත සමුදායන් බහු ආකාරයේ සම්බන්ධතා සඳහා සහාය විය හැකි බැවින් (ඒක-ඒක, ඒක-බහු, බහු-බහු) වැරදිය. මෙම ප්‍රකාශය සම්බන්ධතා වර්ග අනවශ්‍ය ලෙස සීමා කරයි.

Therefore, the correct answer is 4).
එබැවින්, නිවැරදි පිළිතුර 4) වේ.

5. Which SQL query retrieves only the most expensive videogame in the table?
වගුවේ ඇති වඩාත්ම මිල අධික විඩියෝ ක්‍රීඩාව පමණක් ලබා දෙන්නේ කුමන SQL විමසුමද?

1) `SELECT Game_Name FROM Games ORDER BY Price ASC LIMIT 1;`
2) `SELECT Game_Name FROM Games ORDER BY Price DESC;`
3) `SELECT * FROM Games WHERE Price = (SELECT MAX(Price) FROM Games);`
4) `SELECT * FROM Games ORDER BY Price DESC;`
5) `SELECT MAX(Price) FROM Games;`

Answer: 3)

Let's analyze each SQL code / එක් එක් SQL කේතය විශ්ලේෂණය කරමු.

This question is about retrieving the most expensive video game from the table.
මෙම ප්‍රශ්නය සඳහන් වන්නේ වගුවෙන් වඩාත්ම මිල අධික වීඩියෝ ක්‍රීඩාව නැවත ලබා ගැනීම පිළිබඳවය.

1. **`SELECT Game_Name FROM Games ORDER BY Price ASC LIMIT 1;`**
This query orders the video games by price in ascending order and limits the result to only one video game. As a result, it retrieves the video game with the lowest price rather than the most expensive one. This option is incorrect.
මෙම විමසුම වීඩියෝ ක්‍රීඩාව ආරෝහණ අනුපිළිවෙලට මිල අනුව සකසන අතර අතර ප්‍රතිඵලය එක් වීඩියෝ ක්‍රීඩාවකට පමණක් සීමා කරයි. එහි ප්‍රතිඵලයක් වශයෙන්, එය වඩාත්ම මිල අධික එකට වඩා අඩුම මිලට වීඩියෝ ක්‍රීඩාව ලබා ගනී. මෙම විකල්පය සාවද්‍ය වේ.

2. **`SELECT Game_Name FROM Games ORDER BY Price DESC;`**
This query orders the games by price in descending order, returning all game names but does not limit the result to just one game. Therefore, it does not specifically identify the most expensive game. This option is also incorrect.
මෙම විමසුම වීඩියෝ ක්‍රීඩාවල මිල ගණන් අනුව අවරෝහණ අනුපිළිවෙලට ඇණවුම් කරයි, සියලුම වීඩියෝ ක්‍රීඩා නම් ආපසු ලබා දෙයි, නමුත් ප්‍රතිඵලය එක් වීඩියෝ ක්‍රීඩාවකට පමණක් සීමා නොකරයි. එමනිසා, එය වඩාත්ම මිල අධික වීඩියෝ ක්‍රීඩාව විශේෂයෙන් හඳුනා නොගනී. මෙම විකල්පය සාවද්‍ය වේ.

3. **`SELECT * FROM Games WHERE Price = (SELECT MAX(Price) FROM Games);`**
This query accurately retrieves all details of the game(s) that have the maximum price. By using a subquery to identify the highest price, it effectively returns the most expensive game. This option is correct.
මෙම විමසුම උපරිම මිල ඇති වීඩියෝ ක්‍රීඩා(ව) පිළිබඳ සියලු විස්තර නිවැරදිව ලබා ගනී. ඉහළම මිල හඳුනා ගැනීමට උප විමසුමක් භාවිතා කිරීමෙන්, එය වඩාත්ම මිල අධික වීඩියෝ ක්‍රීඩාව ඵලදායී ලෙස ලබා දෙයි. මෙම විකල්පය නිරවද්‍ය වේ.

4. **`SELECT * FROM Games ORDER BY Price DESC;`**
Similar to option 2, this query retrieves all games and orders them by price in descending order but does not restrict the results to the most expensive game. This option is incorrect.
විකල්ප 2ට සමානව, මෙම විමසුම මඟින් සියලුම වීඩියෝ ක්‍රීඩා ලබාගෙන ඒවා මිල ගණන් අනුව අවරෝහණ අනුපිළිවෙලට ඇණවුම් කරන නමුත් ප්‍රතිඵල වඩාත්ම මිල අධික වීඩියෝ ක්‍රීඩාවට සීමා නොකරයි. මෙම විකල්පය සාවද්‍ය වේ.

5. **`SELECT MAX(Price) FROM Games;`**
This query retrieves only the maximum price value from the table and does not return the game associated with that price. This option is incorrect.
මෙම විමසුම වගුවෙන් උපරිම මිල අගය පමණක් ලබා ගන්නා අතර එම මිල හා සම්බන්ධ වීඩියෝ ක්‍රීඩාව ආපසු ලබා නොදේ. මෙම විකල්පය සාවද්‍ය වේ.

Option 3) query effectively retrieves the most expensive video game.
විකල්ප 3) විමසුම වඩාත් මිල අධික වීඩියෝ ක්‍රීඩාව ඵලදායී ලෙස ලබා ගනී.

Therefore, the correct answer is 3).
එබැවින්, නිවැරදි පිළිතුර 3) වේ.

6. What is the purpose of the `DISTINCT` keyword in SQL?
SQL හි `DISTINCT` මුල පදයේ අරමුණ කුමක්ද?

1) To sort the results in ascending order / ප්‍රතිඵල ආරෝහණ අනුපිළිවෙලට වර්ග කිරීමට.
2) To return unique values from the selected column(s)
තෝරාගත් තීරු(ව) වෙතින් අනන්‍ය අගයන් ලබා දීමට.
3) To filter records with null values / ශුන්‍ය අගයන් සහිත වාර්තා පෙරීමට.
4) To count the number of records / වාර්තා ගණන ගණනය කිරීමට.
5) None of the above / ඉහත කිසිවක් නොවේ

Answer: 2)

Explanation / පැහැදිලි කිරීම:

In SQL, the `DISTINCT` keyword is used to eliminate duplicate values from the result set and only return unique values in the specified column(s). It helps ensure that each value appears only once in the output.
SQL හි `DISTINCT` මුල පදය භාවිතා කරනුයේ ප්‍රතිඵල කට්ටලයෙන් අනුපිටපත් අගයන් ඉවත් කිරීමට සහ නිශ්චිත තීරු(ව) තුළ අනන්‍ය අගයන් පමණක් ලබා දීමට ය. ප්‍රතිදානයේ එක් එක් අගය එක් වරක් පමණක් දිස්වන බව සහතික කිරීමට එය උපකාරී වේ.

Therefore, the correct answer is 2) To return unique values from the selected column(s).
එබැවින්, නිවැරදි පිළිතුර 2) තෝරාගත් තීරු(ව) වෙතින් අනන්‍ය අගයන් ලබා දීමට වේ.

Other options / වෙනත් විකල්ප:

Option / විකල්ප 1)
Sorting is done using the `ORDER BY` keyword, not `DISTINCT`.
වර්ග කිරීම `DISTINCT` නොව මුල පදය අනුව `ORDER BY` සිදු කෙරේ.

Option / විකල්ප 3)
This is done using a condition like `WHERE column IS NOT NULL`.
මෙය සිදු කරන්නේ `WHERE column IS NOT NULL` වැනි කොන්දේසියක් භාවිතා කරමිනි.

Option / විකල්ප 4)
Use the `COUNT()` function for this purpose.
මෙම කාර්යය සඳහා `COUNT()` ශ්‍රිතය භාවිතා කරන්න.

Option / විකල්ප 5)
Not applicable since option 2 is correct.
විකල්ප 2 නිවැරදි බැවින් අදාළ නොවේ.

7. What is slicing used for in Python?
පයිතන් හි slicing භාවිතා කරන්නේ කුමක් සඳහාද?

1) To delete elements from a list / list වල මූලාංග මකා දැමීමට.
2) To add elements to a list. list / වලට මූලාංග එකතු කිරීමට.
3) To modify elements in a list / list වල ඇති මූලාංග වෙනස් කිරීමට.
4) To replace elements in a list / list වල ඇති මූලාංග ප්‍රතිස්ථාපනය කිරීමට.
5) To extract elements from a list or tuple / list හෝ tuple එකකින් මූලාංග උකහා ගැනීමට.

Answer : 5)

In Python, slicing is used to extract specific elements from a sequence, such as a list or tuple, without modifying the original sequence. Slicing allows you to select a portion (or "slice") of the list or tuple by specifying a start and end index. This is particularly useful when you need to access a subset of the elements. In the given options, answer **5) To extract elements from a list or tuple** is correct, as slicing is primarily for retrieving elements based on their position. The other options involve modifying or adding elements, which slicing does not do.
Python හි, මුල් අනුපිළිවෙල වෙනස් නොකර, ලැයිස්තුවක් හෝ ටපල් වැනි අනුපිළිවෙලකින් නිශ්චිත මූලද්‍රව්‍ය උකහා ගැනීමට slicing භාවිතා කරයි. Slicing මඟින් ඔබට ආරම්භක සහ අවසන් දර්ශකයක් සඳහන් කිරීමෙන් ලැයිස්තුවේ හෝ tuple හි කොටසක් (හෝ "slice") තෝරා ගැනීමට ඉඩ සලසයි. ඔබට මූලද්‍රව්‍යවල උප කුලකයකට ප්‍රවේශ වීමට අවශ්‍ය වූ විට මෙය විශේෂයෙන් ප්‍රයෝජනවත් වේ.

ලබා දී ඇති විකල්පවල, පිළිතුර **5) ලැයිස්තුවකින් හෝ tuple වෙතින් මූලද්‍රව්‍ය උකහා ගැනීම** නිවැරදි වේ, මන්ද slicing මූලික වශයෙන් ඒවායේ පිහිටීම මත පදනම්ව මූලද්‍රව්‍ය ලබා ගැනීම සඳහා වේ. අනෙකුත් විකල්පවලට කොටස් කිරීම සිදු නොකරන මූලද්‍රව්‍ය වෙනස් කිරීම හෝ එකතු කිරීම ඇතුළත් වේ.

8. Consider the following python code.
පහත පයිතන් කේතය සලකා බලන්න.

```
List1=[5,6,7,8]
List2=[9,10,11,12]
L=List1+List2
print(L)
```

Which of the following is the output of it?
පහත සඳහන් ඒවායින් කුමක් ප්‍රතිදානය වේද?

1) `5, 6, 7, 8, 9, 10, 11, 12`
2) `[5, 6, 7, 8]+[9, 10, 11, 12]`
3) `[5, 6, 7, 8, 9, 10, 11, 12]`
4) `(5, 6, 7, 8)+(9, 10, 11, 12)`
5) `List1[5, 6, 7, 8]+List2[9, 10, 11, 12]`

Answer : 3)

List Initialization: Two lists, `List1` and `List2`, are initialized with the values `[5, 6, 7, 8]` and `[9, 10, 11, 12]` respectively.
ලැයිස්තු ආරම්භ කිරීම: ලැයිස්තු දෙකක්, `List1` සහ `List2` පිළිවෙලින් `[5, 6, 7, 8]` සහ `[9, 10, 11, 12]` අගයන් සමඟ ආරම්භ කර ඇත.

List Concatenation: The `+` operator is used to concatenate these two lists. In Python, the `+` operator combines the elements of two lists into a new list. So, `List1 + List2` creates a new list that includes all elements from List1 followed by all elements from List2. The result is `[5, 6, 7, 8, 9, 10, 11, 12]`.
`+` කාරකය භාවිතා කරන්නේ මෙම ලැයිස්තු දෙක ඒකාබද්ධ කිරීමටයි. Python හි, `+` ක්‍රියාකරු විසින් ලැයිස්තු දෙකක මූලද්‍රව්‍ය නව ලැයිස්තුවකට ඒකාබද්ධ කරයි. එබැවින්, `List1 + List2` විසින් `List1` වෙතින් සියලුම මූලද්‍රව්‍ය පසුව `List2` වෙතින් සියලුම අංග ඇතුළත් නව ලැයිස්තුවක් නිර්මාණය කරයි. ප්‍රතිපලය වන්නේ `[5, 6, 7, 8, 9, 10, 11, 12]`.

Printing the Result: The `print(L)` statement outputs the concatenated list.
ප්‍රතිඵලය මුද්‍රණය කිරීම: `print(L)` ප්‍රකාශය සංකලිත ලැයිස්තුව ප්‍රතිදානය කරයි.

The correct output is `[5, 6, 7, 8, 9, 10, 11, 12]` which matches option 3. This result demonstrates how Python’s list operations work, specifically how concatenation with the `+` operator combines elements from multiple lists into a single list.
නිවැරදි ප්‍රතිදානය `[5, 6, 7, 8, 9, 10, 11, 12]` වන අතර එය විකල්ප 3) ට ගැලපේ. මෙම ප්‍රතිඵලය Python හි ලැයිස්තු මෙහෙයුම් ක්‍රියා කරන ආකාරය, විශේෂයෙන් `+` ක්‍රියාකරු සමඟ සම්බන්ධ කිරීම බහු ලැයිස්තු වල මූලද්‍රව්‍ය තනි ලැයිස්තුවකට ඒකාබද්ධ කරන ආකාරය නිරූපණය කරයි.

9. What will be the output of the following python operations?
පහත සඳහන් පයිතන් මෙහෙයුම් වල ප්‍රතිදානය කුමක්ද?

```
print(12 & 5)
print(12 | 5)
```

1)	2)	3)	4)	5)
4 13	8 17	12 5	0 1	9 16

Answer : 1)

In the above Python program,
ඉහත Python වැඩසටහනේ,

- `print(12 & 5)`
  Line 1, the integers 12 and 5 are converted to binary, to perform bitwise AND operation and the result is converted to decimal is printed.
  1 පේළිය, නිඛිල 12 සහ 5 ද්විමය බවට පරිවර්තනය කර, බිටු අනුසාරිත AND ක්‍රියාකාරිත්වය සිදු කර ප්‍රතිඵලය දශමයට පරිවර්තනය කර මුද්‍රණය කෙරේ.

Conversion of 12_{10} into binary 12_{10} ද්විමය බවට පරිවර්තනය කිරීම				
16	8	4	2	1
0	1	1	0	0
01100_2				

Conversion of 5_{10} into binary 5_{10} ද්විමය බවට පරිවර්තනය කිරීම				
16	8	4	2	1
0	0	1	0	1
00101_2				

Perform bitwise AND operation බිටු අනුසාරිත AND ක්‍රියාකාරිත්වය			
0	1	0	1
& (AND)			
0	1	0	0
0100			

Conversion of binary 0100 into decimal ද්විමය 0100 දශමය බවට පරිවර්තනය කිරීම			
8	4	2	1
0	1	0	0
4			

- **Outputs / ප්‍රතිදානය 4**

  `print(12 | 5)`

  Line 2, the integers 12 and 5 are converted to binary, to perform bitwise OR operation and the result is converted to decimal is printed.
  පේළිය 2 නිඛිල 12 සහ 5 ද්විමය බවට පරිවර්තනය කර, බිටු අනුසාරිත OR ක්‍රියාකාරිත්වය සිදු කර ප්‍රතිඵලය දශමයට පරිවර්තනය කර මුද්‍රණය කෙරේ.

Perform bitwise OR operation බිටු අනුසාරිත OR ක්‍රියාකාරිත්වය			
1	1	0	0
0	1	0	1
| (OR)			
1	1	0	1
1101			

Conversion of binary 1101 into decimal ද්විමය 1101 දශමය බවට පරිවර්තනය කිරීම			
8	4	2	1
1	1	0	1
8+4+1			
13			

There fore the correct answer is, 1).
එබැවින් නිවැරදි පිළිතුර වන්නේ, 1) වේ.

10. Which of the following statement about the HTML <p> tag is incorrect?
HTML, <p> උසුලනය පිළිබඳ පහත සඳහන් ප්‍රකාශවලින් කුමන ප්‍රකාශය වැරදිද?

1) The <p> tag is used to define a paragraph in HTML.
HTML හි ජේදයක් අර්ථ දැක්වීමට <p> උසුලනය භාවිතා කරයි.

2) Multiple <p> tags can be used to separate paragraphs on a webpage.
වෙබ් පිටුවක ජේද වෙන් කිරීමට <p> උසුලනය කිහිපයක් භාවිතා කළ හැකිය.

3) The <p> tag adds line breaks before and after the content by default.
<p> උසුලනය පෙරනිමියෙන් අන්තර්ගතයට පෙර සහ පසු පේළි බිඳීම් එක් කරයි.

4) <u>The <p> tag is used to create horizontal lines in a webpage.</u>
<p> උසුලනය වෙබ් පිටුවක තිරස් රේඛා නිර්මාණය කිරීමට භාවිතා කරයි.

5) By default, the browser adds a margin above and below a paragraph.
පෙරනිමියෙන්, අතිරික්සුව ජේදයකට ඉහළින් සහ පහළින් margin කරයි.

Answer: 4)

The correct answer is 4) The <p> tag is used to create horizontal lines in a webpage. This statement is incorrect because the <p> tag defines paragraphs, while horizontal lines are created using the <hr> tag. The other statements are true: the <p> tag is specifically designed to define a paragraph in HTML **(1)**, and multiple <p> tags can be used to separate paragraphs on a webpage (2). By default, browsers add line breaks before and after content within a <p> tag to ensure proper spacing (3), and they also apply a default margin above and below paragraphs unless customized using CSS (5). This makes the <p> tag essential for structuring text content in web design.
නිවැරදි පිළිතුර 4) <p> උසුලනය වෙබ් පිටුවක තිරස් රේඛා නිර්මාණය කිරීමට භාවිතා කරයි. <p> උසුලනය ජේද නිර්වචනය කරන අතර තිරස් රේඛා <hr> උසුලනය භාවිතයෙන් නිර්මාණය කර ඇත. අනෙක් ප්‍රකාශ සත්‍ය වේ: <p> උසුලනය විශේෂයෙන් HTML **(1)** හි ජේදයක් අර්ථ දැක්වීමට නිර්මාණය කර ඇති අතර, බහු <p> උසුලනය වෙබ් පිටුවක ජේද වෙන් කිරීමට භාවිතා කළ හැකිය **(2)**. පෙරනිමියෙන්, අතිරික්සුව නිසි පරතර (spacing) සහතික කිරීම සඳහා <p> උසුලනය තුළ අන්තර්ගතයට පෙර සහ පසු රේඛා බිඳීම් එකතු කරන අතර, CSS (5) භාවිතයෙන් අභිරුචිකරණය නොකළහොත් ඒවා ජේදවලට ඉහළින් සහ පහළින් පෙරනිමි ආන්තිකයක් ද යොදයි. මෙය වෙබ් නිර්මාණයේ පාඨ අන්තර්ගතය ව්‍යුහගත කිරීම සඳහා <p> උසුලනය අත්‍යවශ්‍ය කරයි.