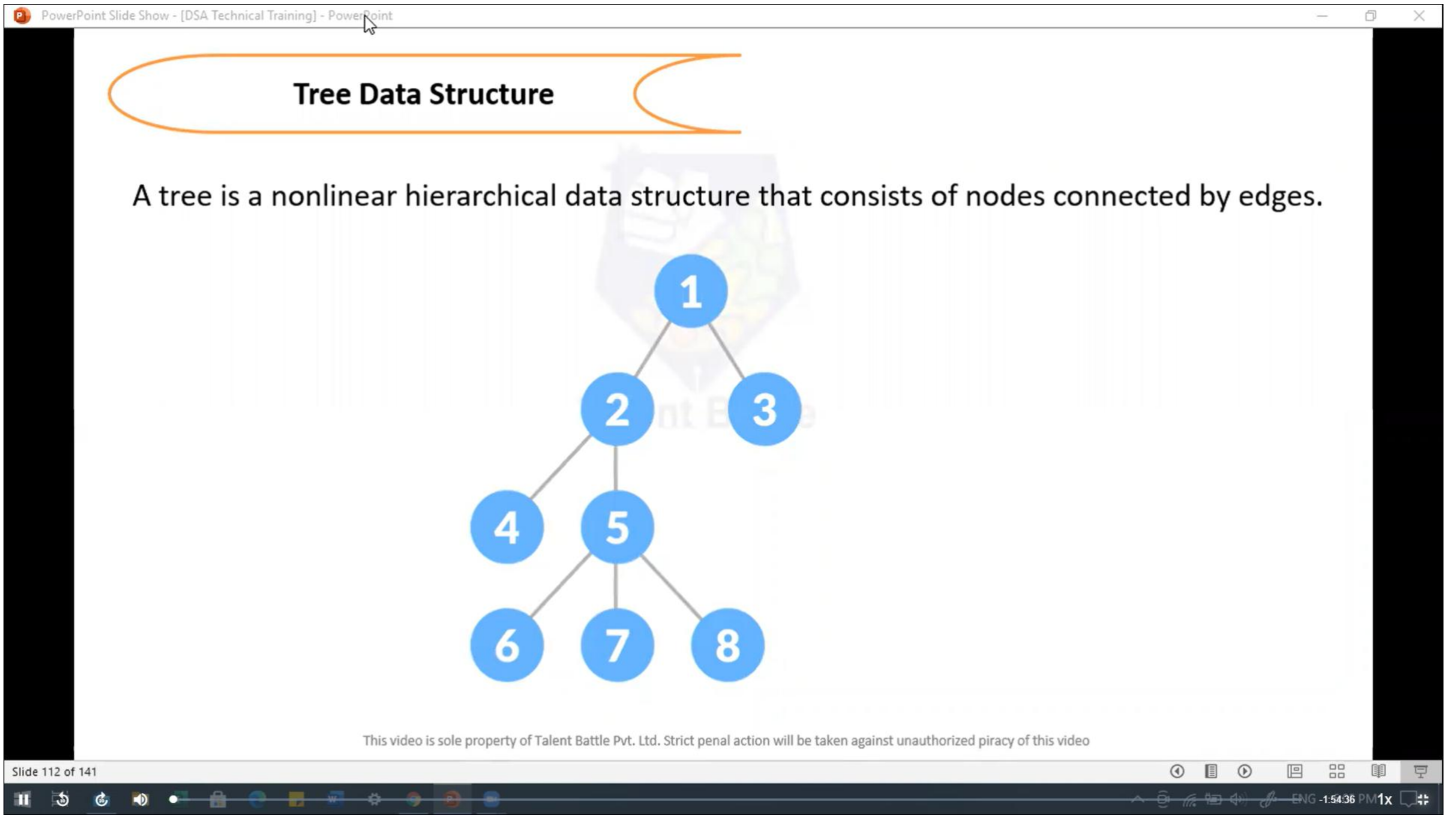
Session-7

Tree Data Structure Part 1

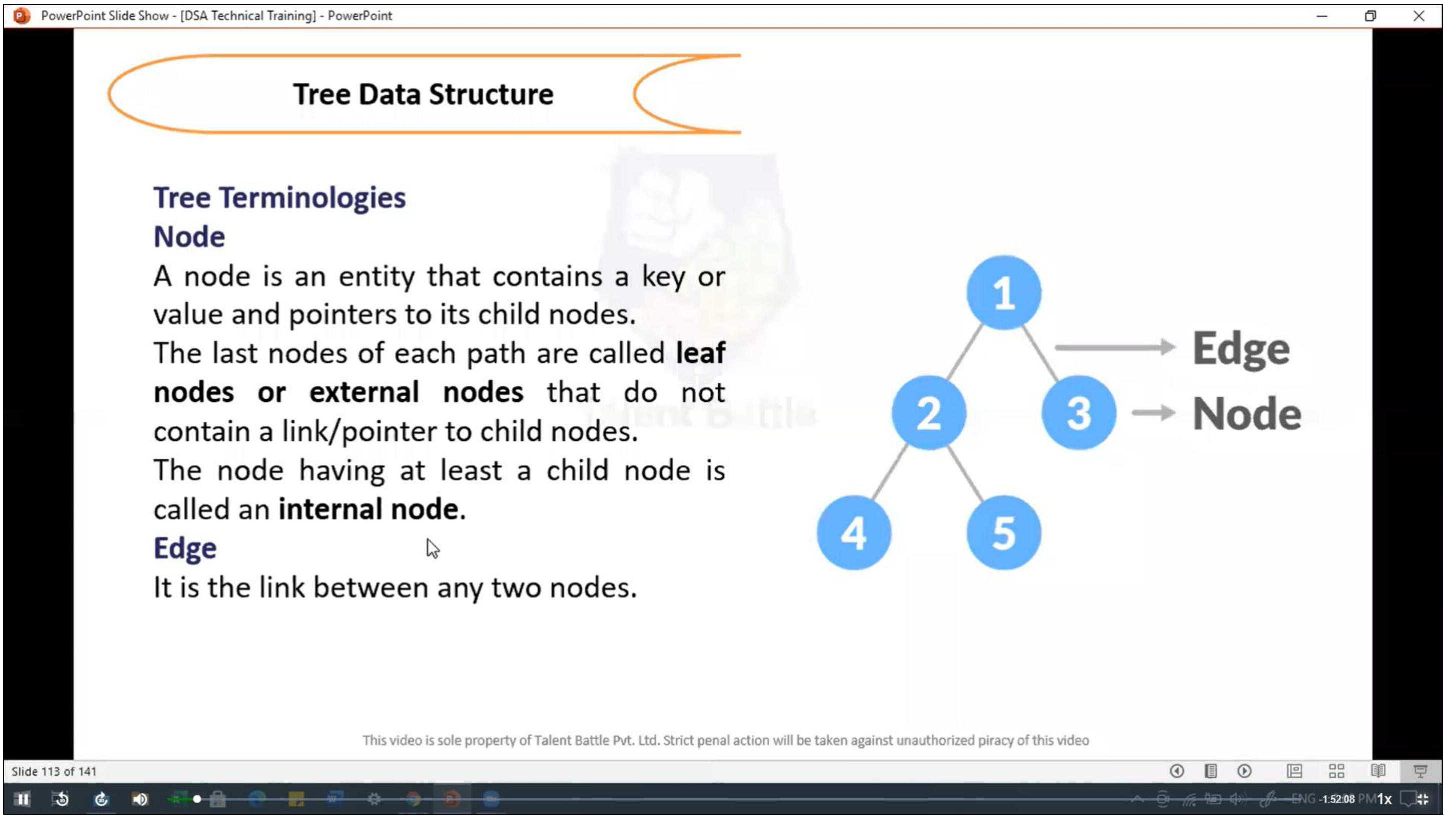


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

A tree is a nonlinear hierarchical data structure that consists of nodes connected by edges.

Slide 112 of 141



PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Tree Terminologies

Node

A node is an entity that contains a key or value and pointers to its child nodes.

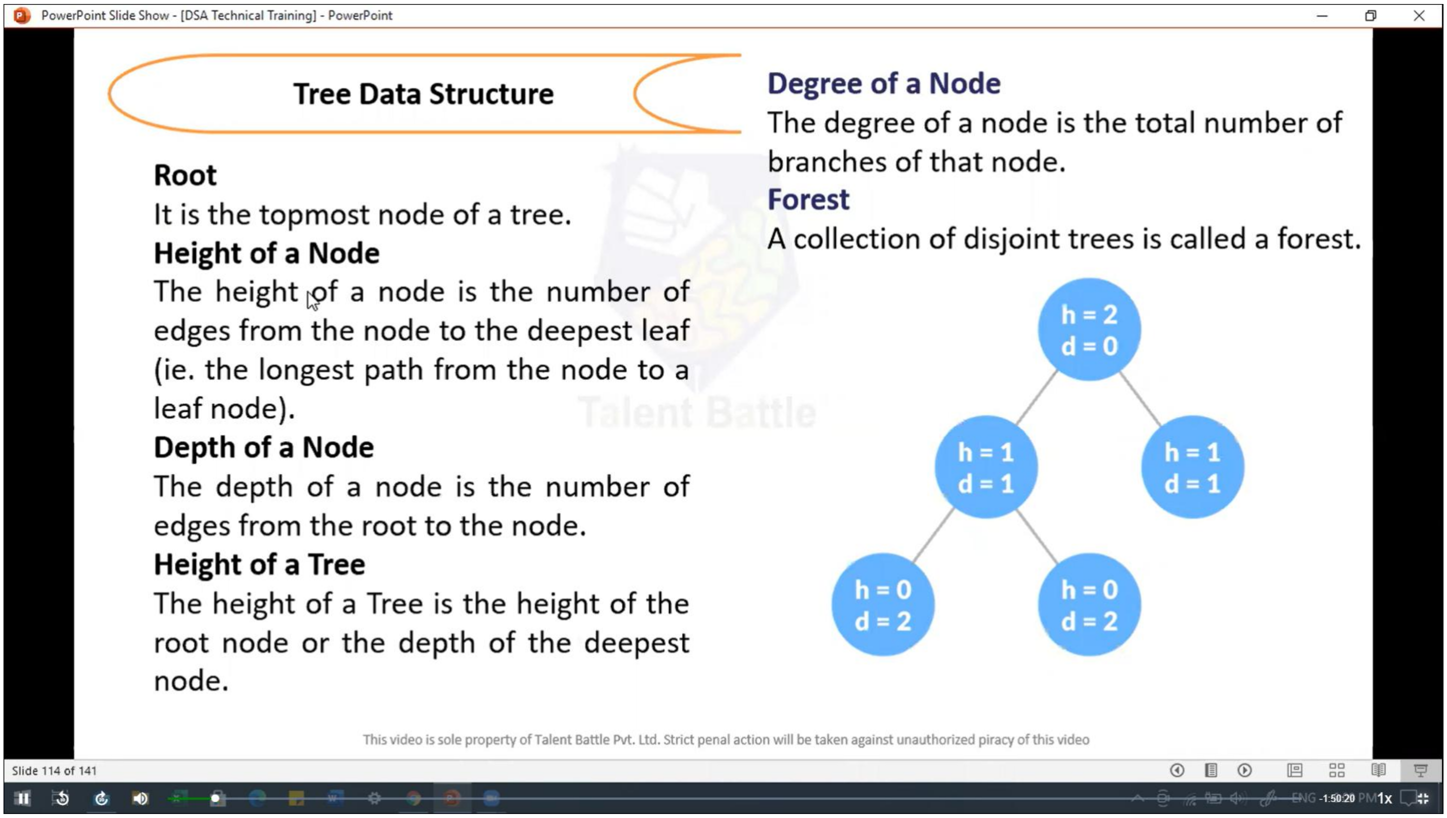
The last nodes of each path are called **leaf nodes or external nodes** that do not contain a link/pointer to child nodes.

The node having at least a child node is called an **internal node**.

Edge

It is the link between any two nodes.

Slide 113 of 141



PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Root

It is the topmost node of a tree.

Height of a Node

The height of a node is the number of edges from the node to the deepest leaf (ie. the longest path from the node to a leaf node).

Depth of a Node

The depth of a node is the number of edges from the root to the node.

Height of a Tree

The height of a Tree is the height of the root node or the depth of the deepest node.

Degree of a Node

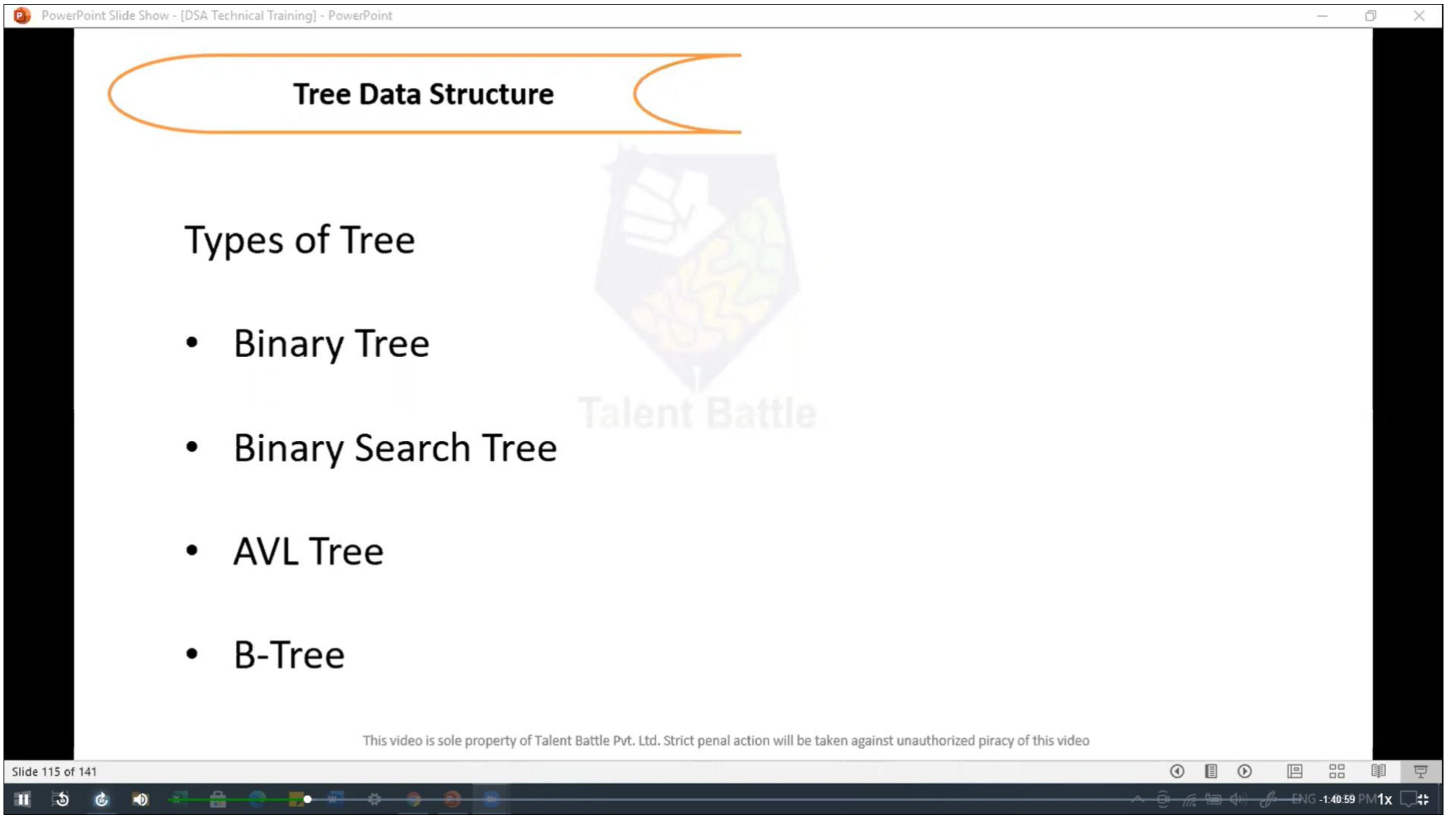
The degree of a node is the total number of branches of that node.

Forest

A collection of disjoint trees is called a forest.

This video is sole property of Talent Battle Pvt. Ltd. Strict penal action will be taken against unauthorized piracy of this video

Slide 114 of 141



PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

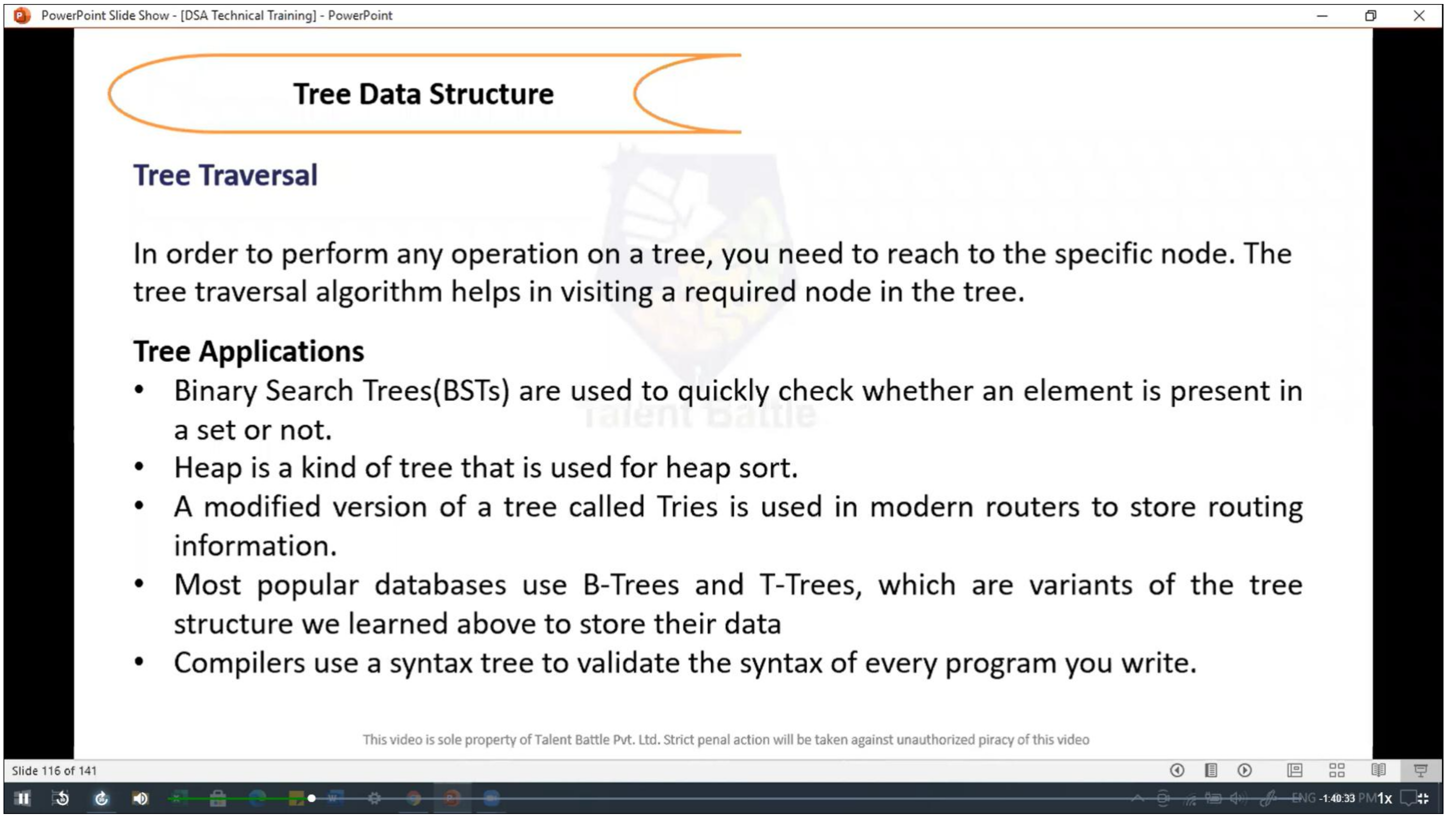
Tree Data Structure

Types of Tree

- Binary Tree
- Binary Search Tree
- AVL Tree
- B-Tree

This video is sole property of Talent Battle Pvt. Ltd. Strict penal action will be taken against unauthorized piracy of this video

Slide 115 of 141


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint
Slide 116 of 141

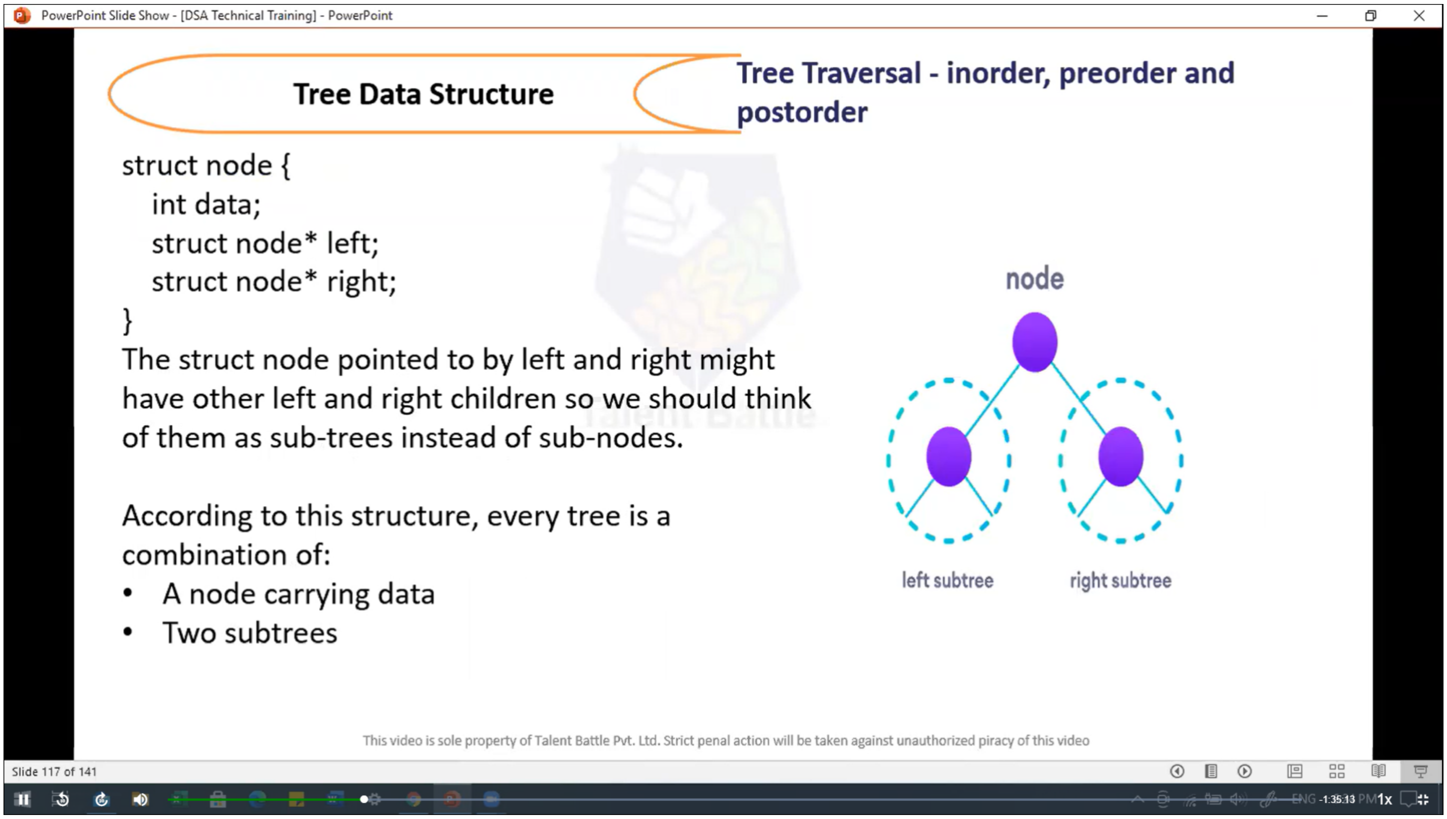
Tree Data Structure

Tree Traversal

In order to perform any operation on a tree, you need to reach to the specific node. The tree traversal algorithm helps in visiting a required node in the tree.

Tree Applications

- Binary Search Trees(BSTs) are used to quickly check whether an element is present in a set or not.
- Heap is a kind of tree that is used for heap sort.
- A modified version of a tree called Tries is used in modern routers to store routing information.
- Most popular databases use B-Trees and T-Trees, which are variants of the tree structure we learned above to store their data
- Compilers use a syntax tree to validate the syntax of every program you write.


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint
Slide 117 of 141
node
left subtree
right subtree

Tree Data Structure

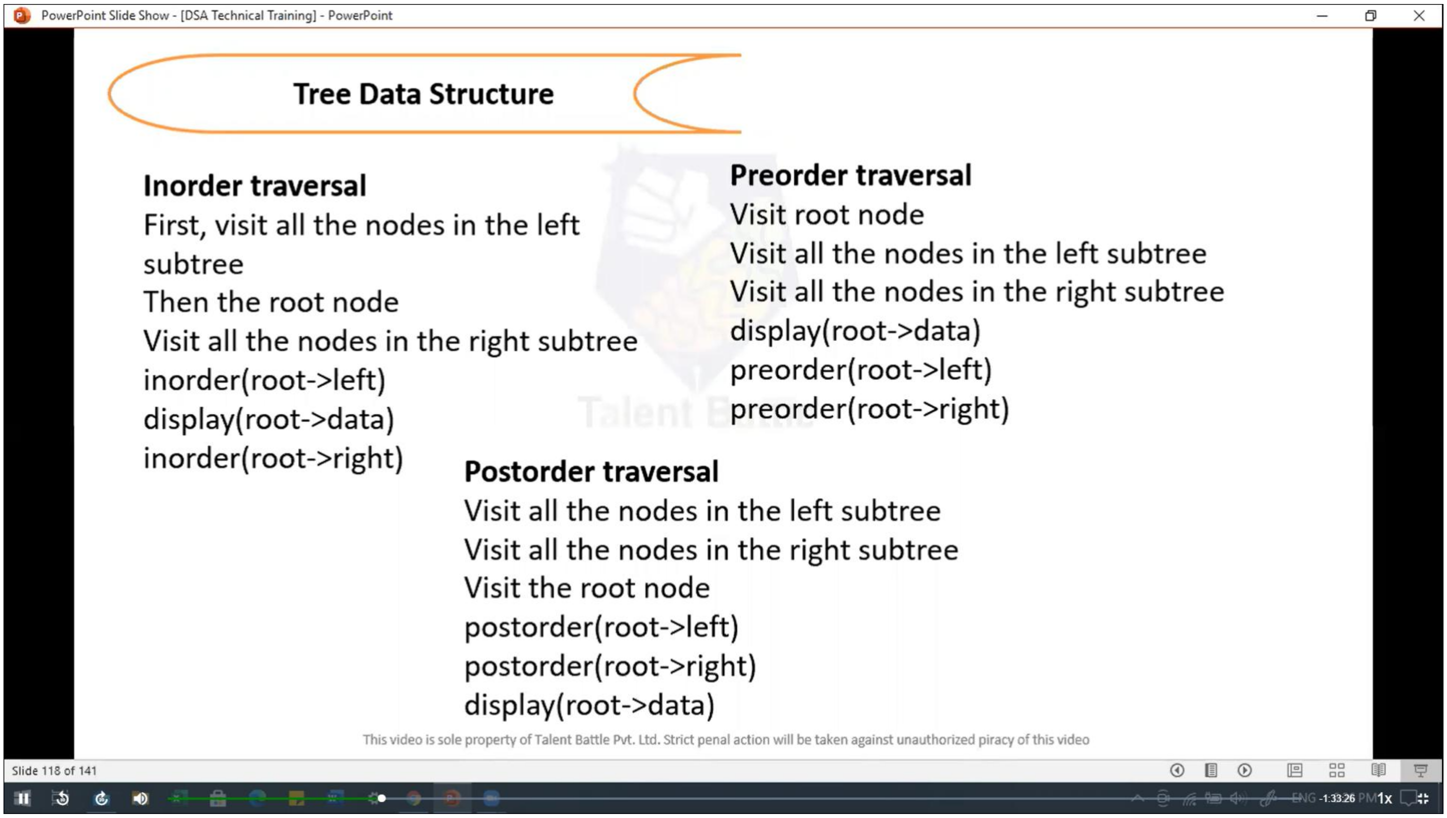
Tree Traversal - inorder, preorder and postorder

```
struct node {
   int data;
   struct node* left;
   struct node* right;
}
```

The struct node pointed to by left and right might have other left and right children so we should think of them as sub-trees instead of sub-nodes.

According to this structure, every tree is a combination of:

- A node carrying data
- Two subtrees


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint
Slide 118 of 141

Tree Data Structure

Inorder traversal

First, visit all the nodes in the left subtree

Then the root node

Visit all the nodes in the right subtree

```
inorder(root->left)
display(root->data)
inorder(root->right)
```

Preorder traversal

Visit root node

Visit all the nodes in the left subtree

Visit all the nodes in the right subtree

```
display(root->data)
preorder(root->left)
preorder(root->right)
```

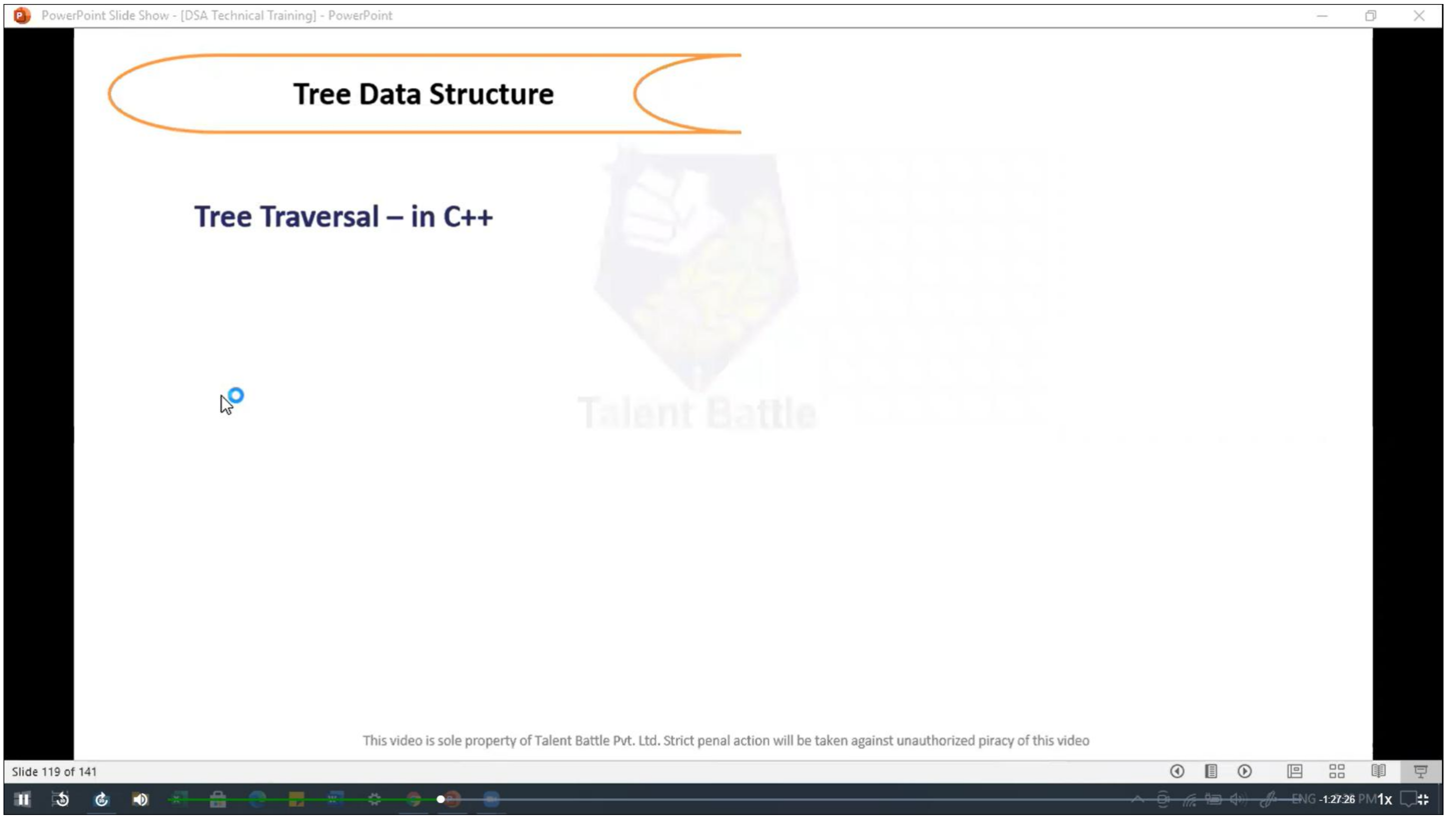
Postorder traversal

Visit all the nodes in the left subtree

Visit all the nodes in the right subtree

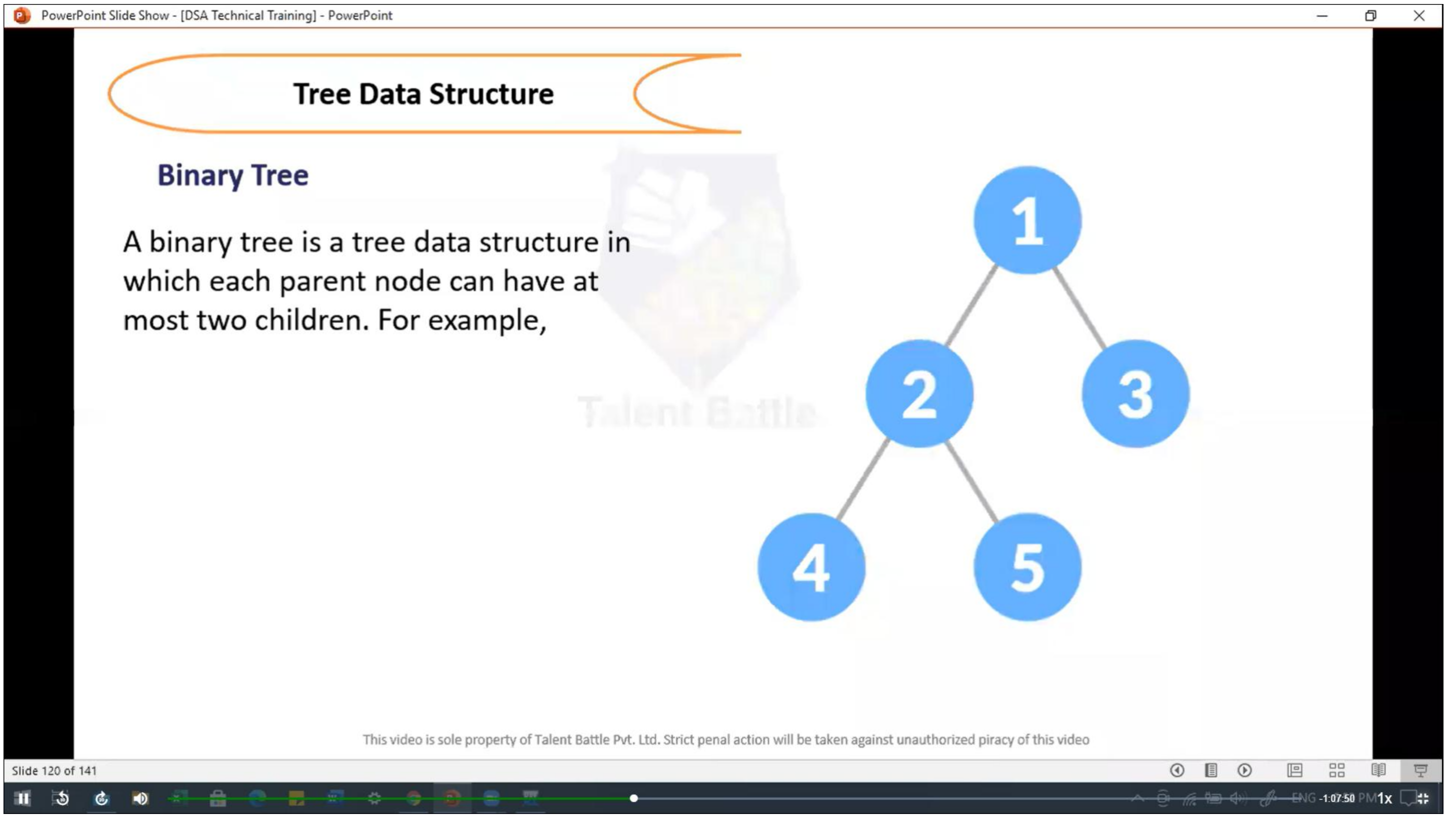
Visit the root node

```
postorder(root->left)
postorder(root->right)
display(root->data)
```


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint
Slide 119 of 141

Tree Data Structure

Tree Traversal – in C++



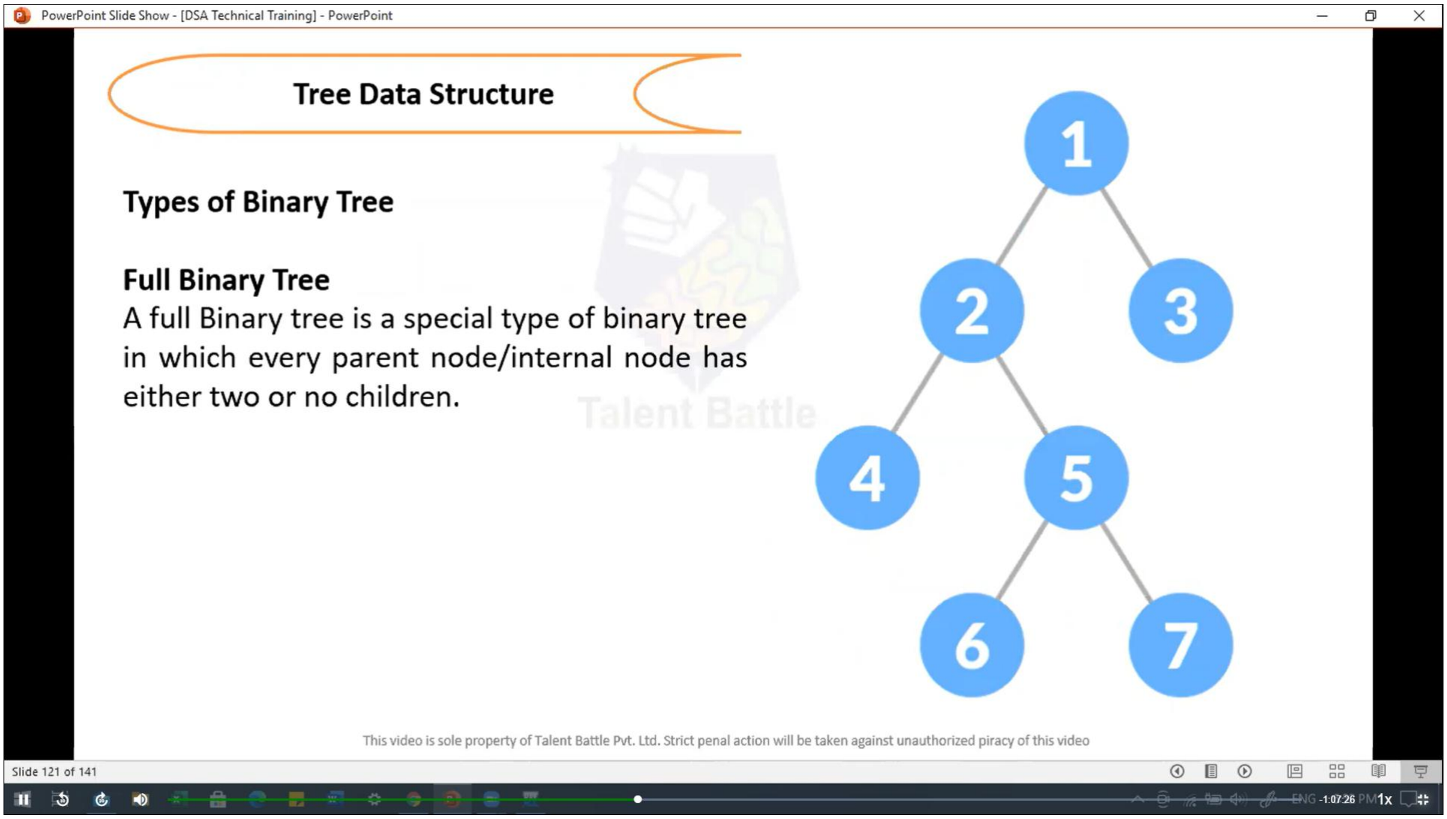
PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Binary Tree

A binary tree is a tree data structure in which each parent node can have at most two children. For example,

Slide 120 of 141



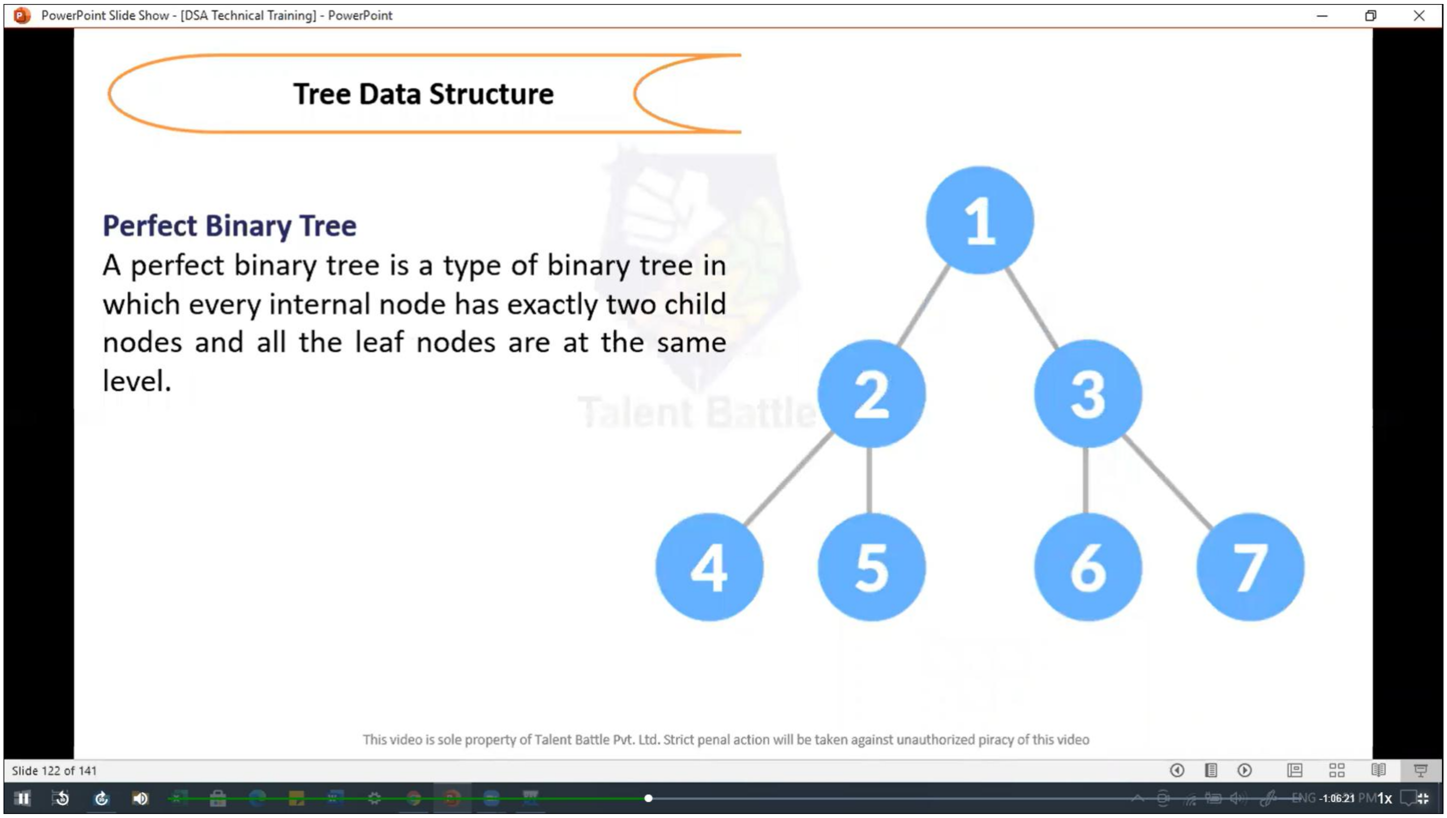
PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Types of Binary Tree

Full Binary Tree
A full Binary tree is a special type of binary tree in which every parent node/internal node has either two or no children.

Slide 121 of 141



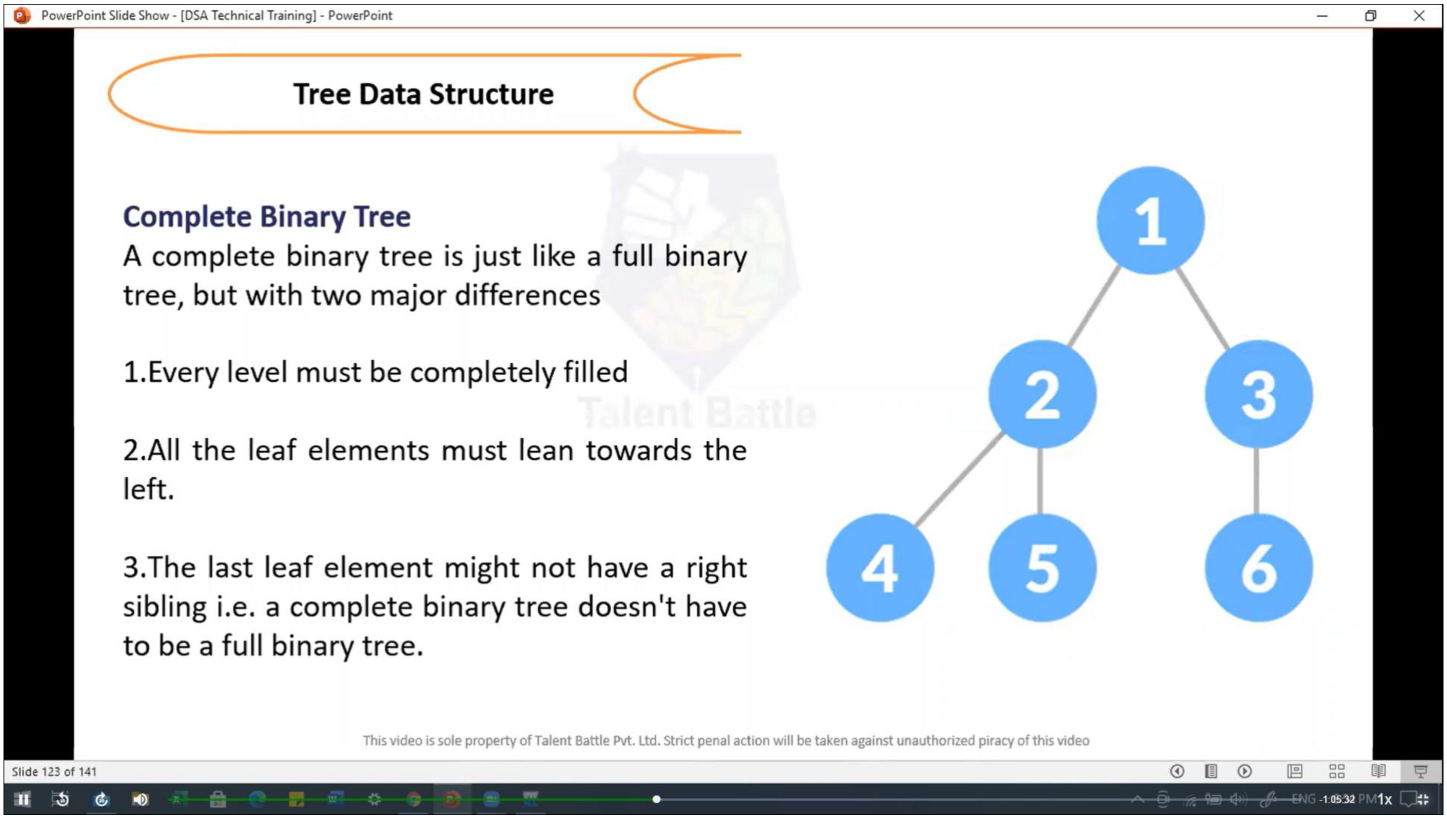
PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Perfect Binary Tree

A perfect binary tree is a type of binary tree in which every internal node has exactly two child nodes and all the leaf nodes are at the same level.

Slide 122 of 141



PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Complete Binary Tree

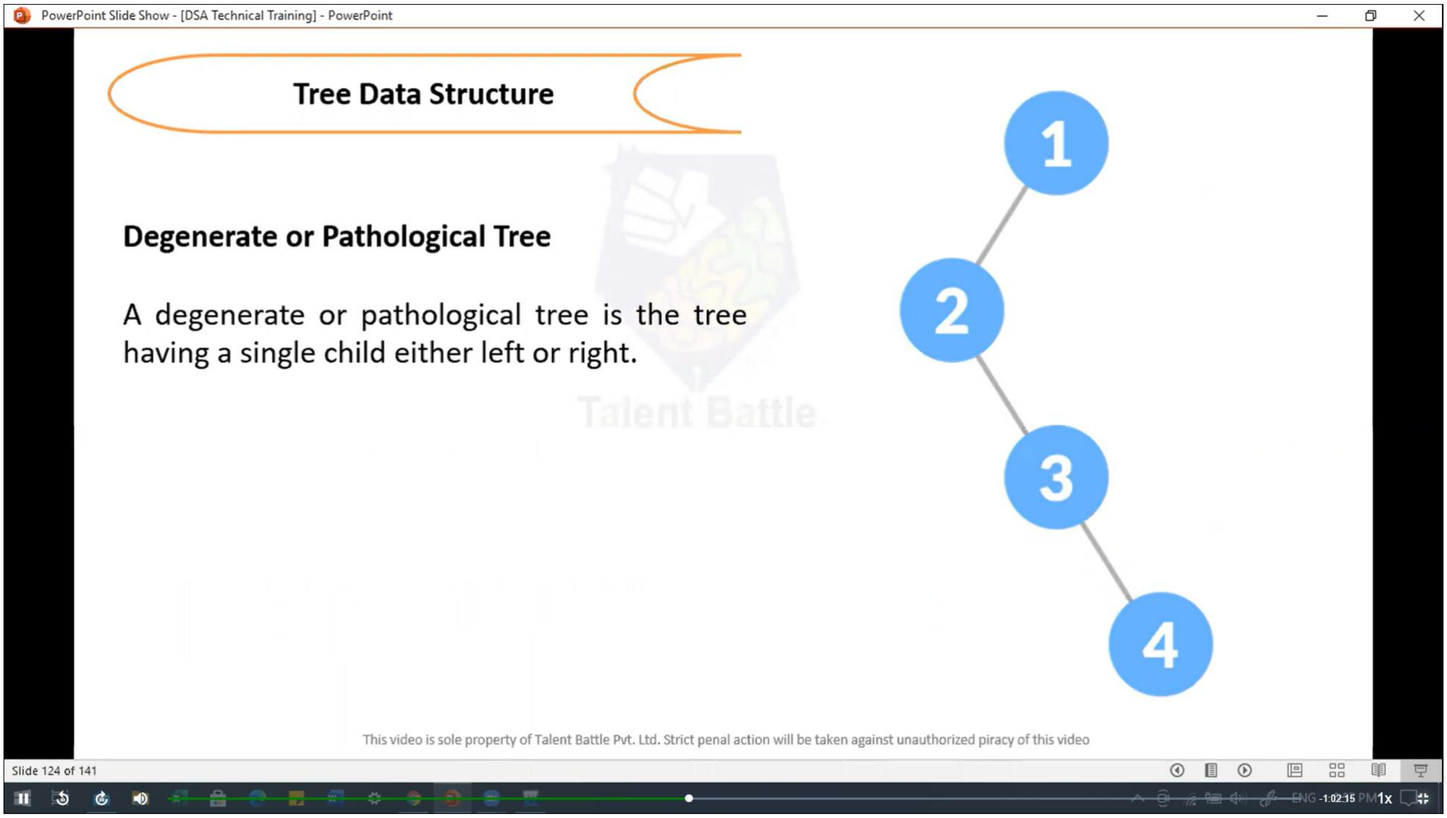
A complete binary tree is just like a full binary tree, but with two major differences

1.Every level must be completely filled

2.All the leaf elements must lean towards the left.

3.The last leaf element might not have a right sibling i.e. a complete binary tree doesn't have to be a full binary tree.

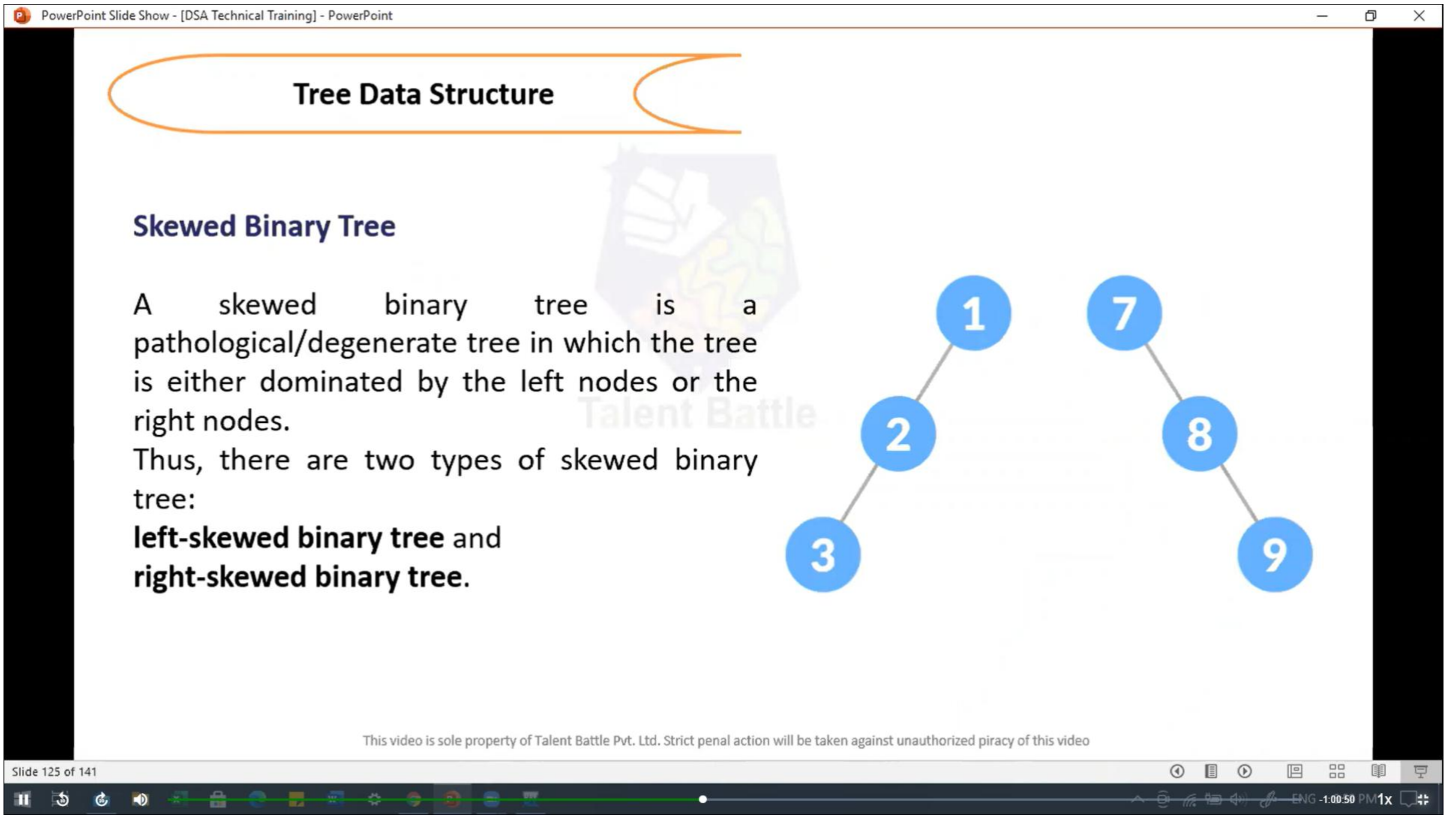
Slide 123 of 141


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint
1
2
3
4
Slide 124 of 141
ENG
1:02:15 PM
1x

Tree Data Structure

Degenerate or Pathological Tree

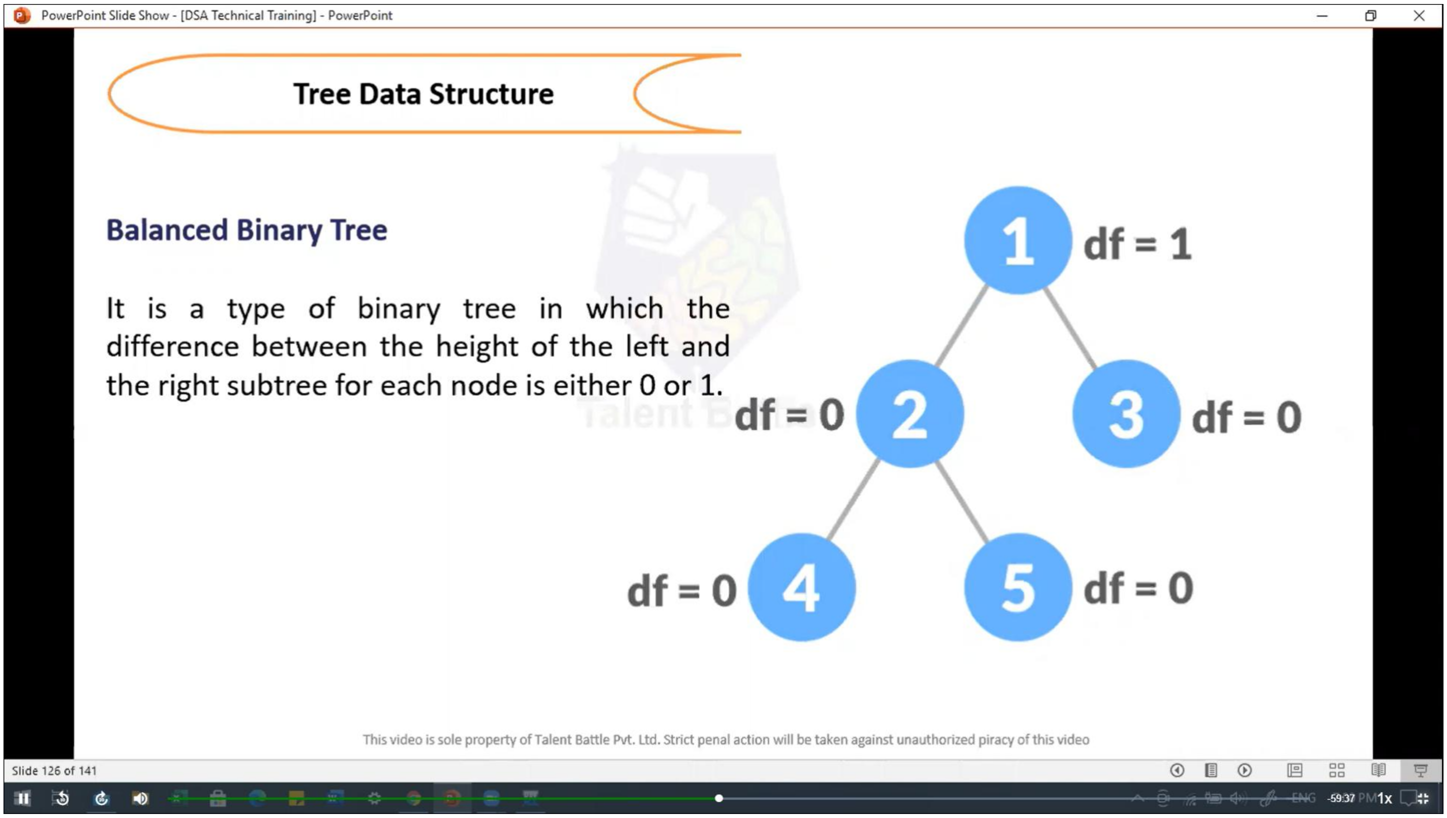
A degenerate or pathological tree is the tree having a single child either left or right.


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint
1
2
3
7
8
9
Slide 125 of 141
ENG
1:00:50 PM
1x

Tree Data Structure

Skewed Binary Tree

A skewed binary tree is a pathological/degenerate tree in which the tree is either dominated by the left nodes or the right nodes.
Thus, there are two types of skewed binary tree:
left-skewed binary tree and
right-skewed binary tree.



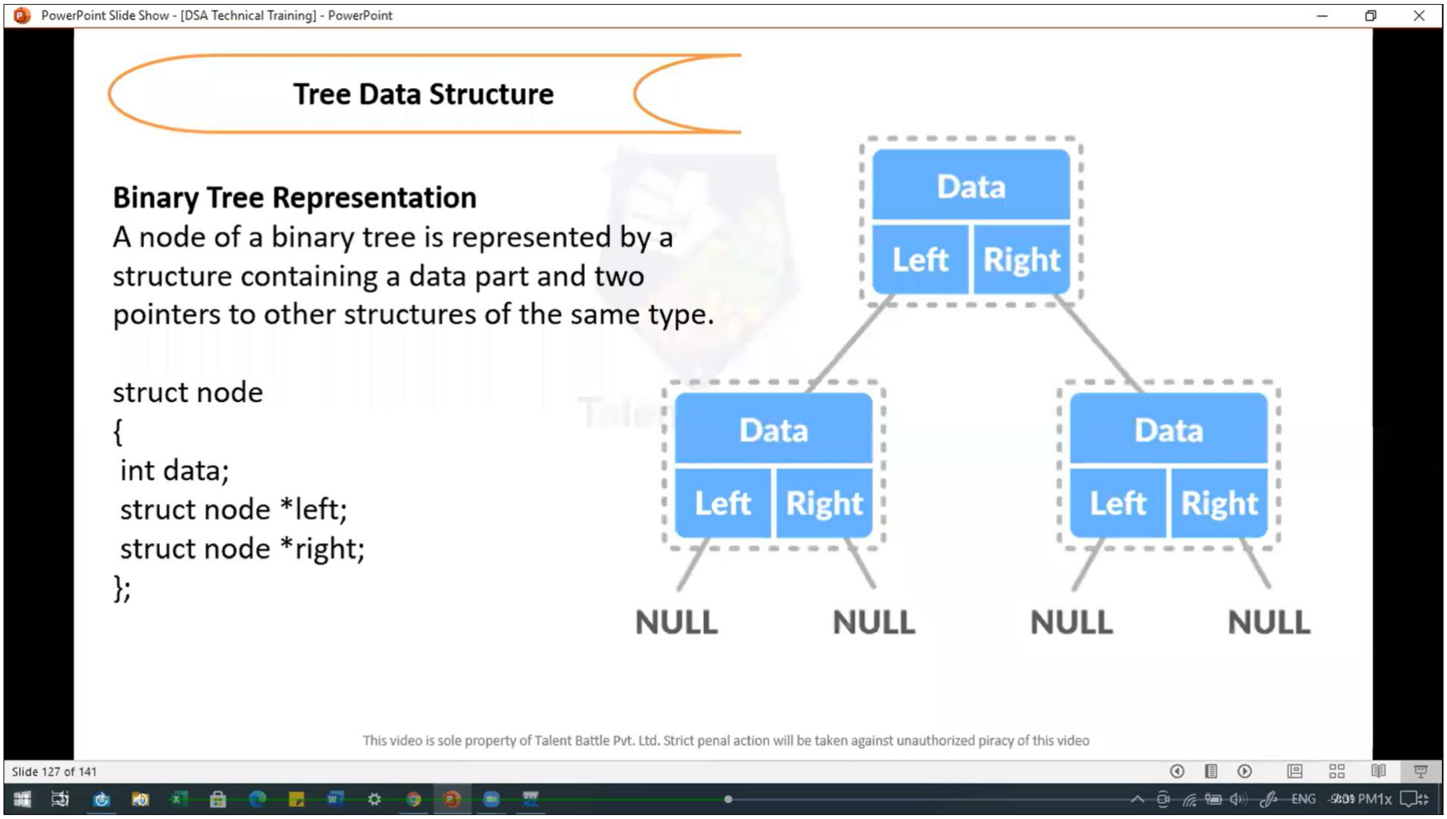
PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Balanced Binary Tree

It is a type of binary tree in which the difference between the height of the left and the right subtree for each node is either 0 or 1.

1
df = 1
2
df = 0
3
df = 0
4
df = 0
5
df = 0
Slide 126 of 141



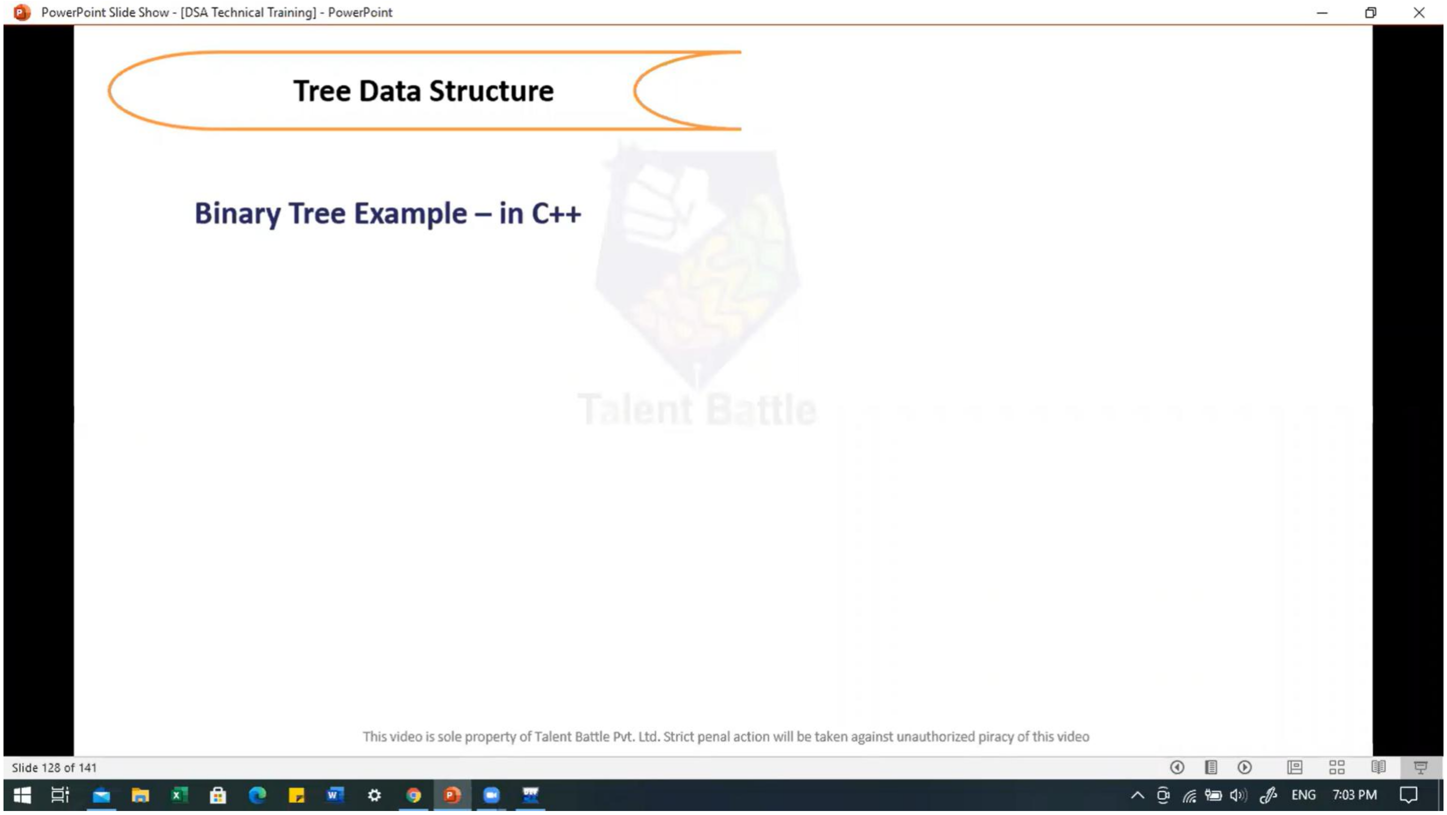
PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Binary Tree Representation

A node of a binary tree is represented by a structure containing a data part and two pointers to other structures of the same type.

```
struct node
{
 int data;
 struct node *left;
 struct node *right;
};
```

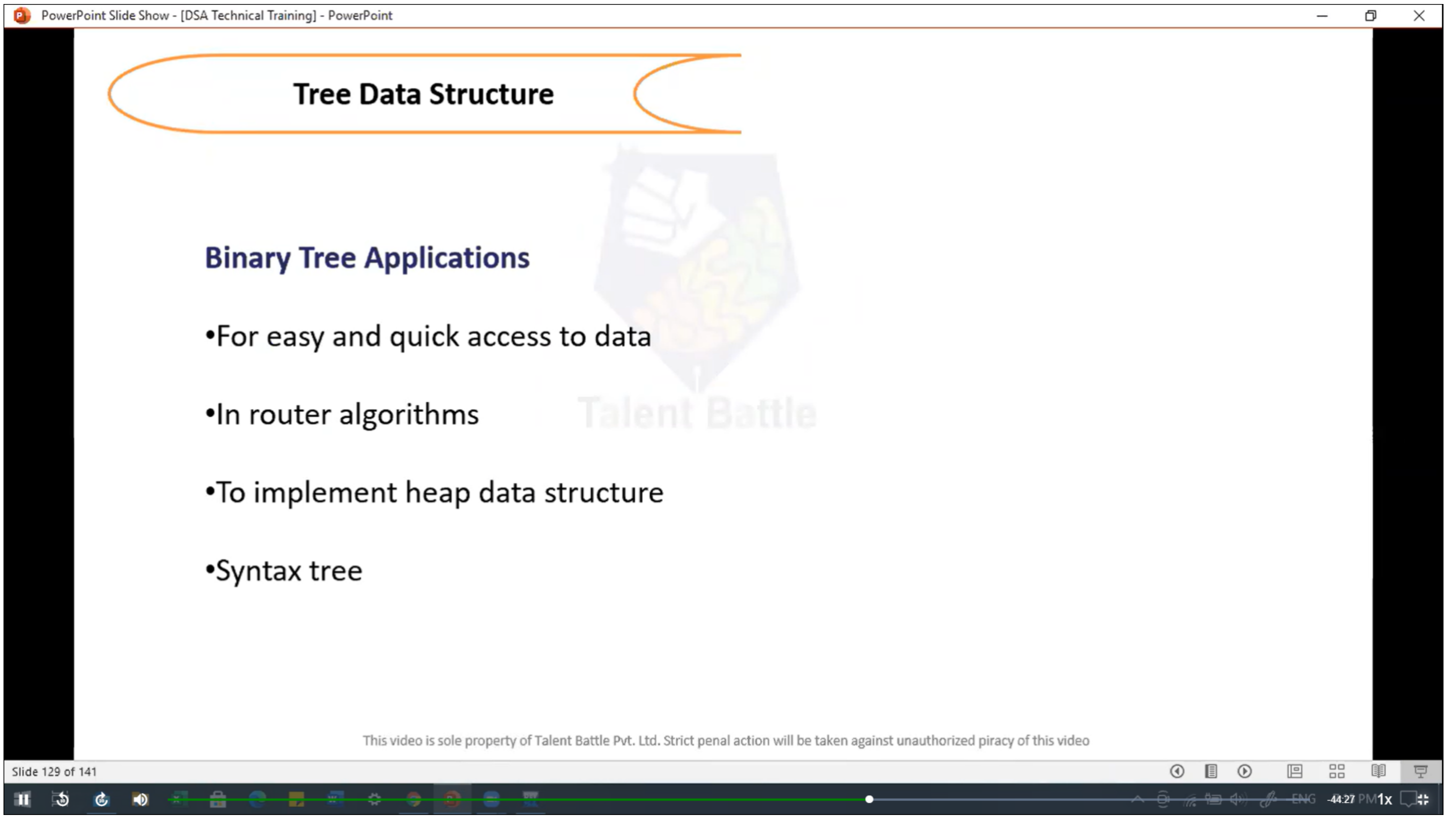


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Binary Tree Example – in C++

Slide 128 of 141
ENG
7:03 PM



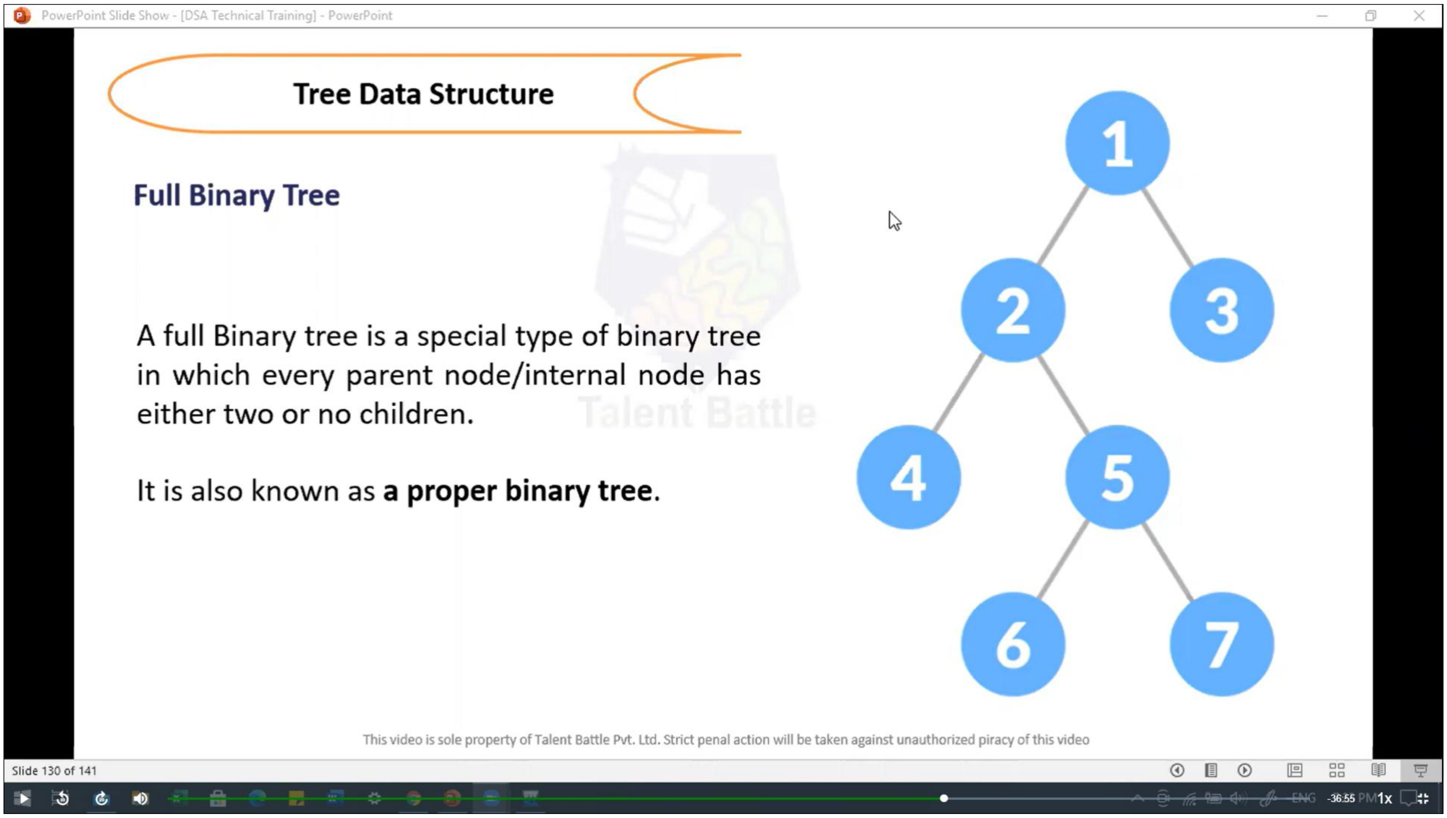
PowerPoint Slide Show - [DSA Technical Training] - PowerPoint

Tree Data Structure

Binary Tree Applications

- For easy and quick access to data
- In router algorithms
- To implement heap data structure
- Syntax tree

Slide 129 of 141
-44:27
1x

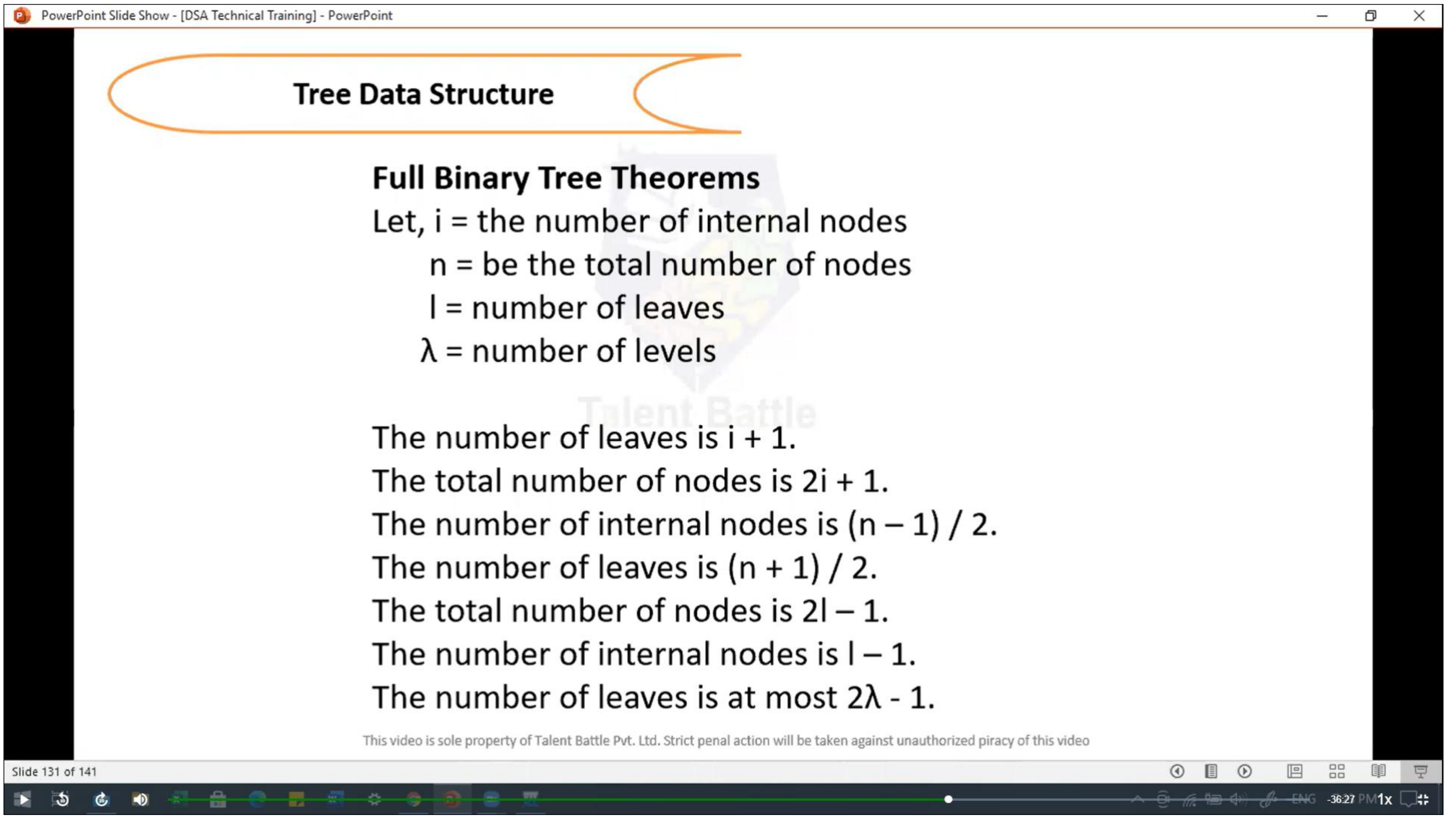

PowerPoint Slide Show - [DSA Technical Training] - PowerPoint
1
2
3
4
5
6
7
Slide 130 of 141

Tree Data Structure

Full Binary Tree

A full Binary tree is a special type of binary tree in which every parent node/internal node has either two or no children.

It is also known as **a proper binary tree**.


PowerPoint Slide Show - [DSA Technical Training] - PowerPoint
Slide 131 of 141

Tree Data Structure

Full Binary Tree Theorems

Let, i = the number of internal nodes
n = be the total number of nodes
l = number of leaves
λ = number of levels

The number of leaves is i + 1.
The total number of nodes is 2i + 1.
The number of internal nodes is (n – 1) / 2.
The number of leaves is (n + 1) / 2.
The total number of nodes is 2l – 1.
The number of internal nodes is l – 1.
The number of leaves is at most 2λ - 1.