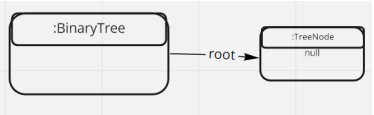
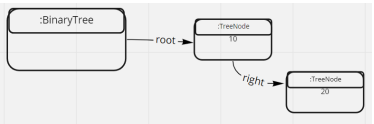
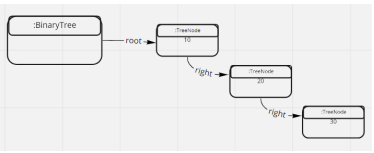
Test Cases Design

BinaryTreeTest

Name	Class	Scene
sc1	BinaryTreeTest	
sc2	BinaryTreeTest	
sc3	BinaryTreeTest	

Create Node Method Test

Objective: Validate the correct creation of a node when the tree is empty wich means it has no elements

Class	Method	Scene	Entries	Result
BinaryTree	createNode	sc1	10	After creating the node the tree is not going to be empty and the root of it is not null, because node(10) is the root now.

Objective: Validate the correct creation of a node when the tree has two elements, the root and the right child.

Class	Method	Scene	Entries	Result
BinaryTree	createNode	sc2	30	Before creating the new

				node(30) the initialLeaf was 20 and its right child was null but after creating it the wight child of 20 is not null, and the created node is now a leaf

IsEmpty Method test

Objective: Validate if when the tree is empty the method says it is empty

Class	**Method**	**Scene**	**Entries**	**Result**
BinaryTree	isEmpty	sc1	none	The tree is empty because this test have been done with scene 1.

Objective: Validate if when the tree has 2 elements the method says the tree is not empty

Class	**Method**	**Scene**	**Entries**	**Result**
BinaryTree	isEmpty	sc2	none	The tree is not empty because this test have been done with scene 2, and there are already two elements on the tree.

IsLeaf Method test

Objective: Validate if when there are 2 elements, the root and the right child it determines that the root is not a leaf but the right child is a leaf.

Class	**Method**	**Scene**	**Entries**	**Result**
BinaryTree	isLeaf	sc2	none	The root is not a

				leaf so its gonna be false, and the right child is a leaf so it is going to be true.

GetTreeNode Method test

Objective: Validate if when there are 2 elements, the root and the right child it determines that an element exists or not, if not the method returns null

Class	**Method**	**Scene**	**Entries**	**Result**
BinaryTree	getTreeNode	sc2	10, 11	When the method looks for the 10 it is going to return the rootNode because there is where the 10 is ubicated, but when the method looks for 11 it is going to return Null because it doesn't exist

DeleteNode Method test

Objective: Validate if the process to delete a leaf is doing correctly

Class	**Method**	**Scene**	**Entries**	**Result**
BinaryTree	deleteNode	sc3	initialIsLeaf= 20 False because 20 is not a leaf	The node(20) is now a leaf because node(30) has been deleted.

Objective: Validate if the process to delete the root is doing correctly

Class	Method	Scene	Entries	Result
BinaryTree	deleteNode	sc3	initialRoot=10 InitialRootRight Child=20	After deleting 10 the node that was the initialRootRight (20) is now the newRoot