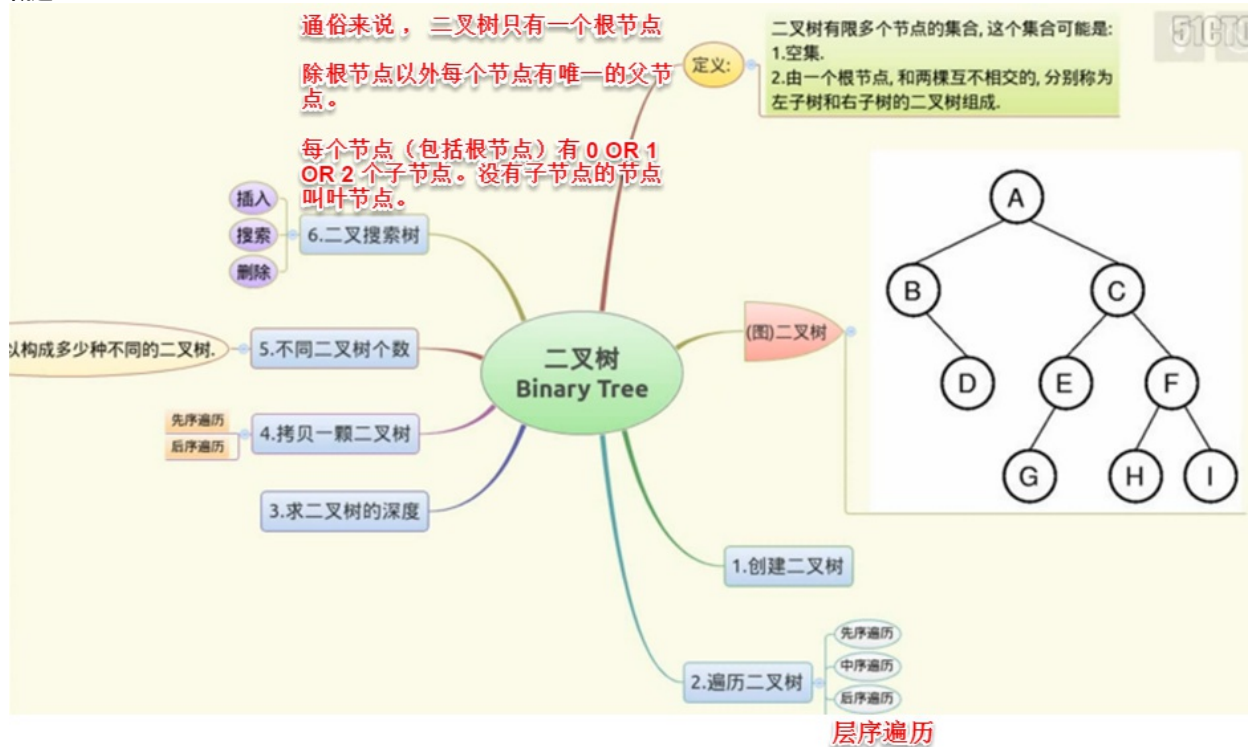


二叉树-1

? TreeNode.py

• 概述



• 案例：创建二叉树

- 代码-1：创建二叉树。还有一种初始化二叉树节点的方式 `A = TreeNode('A', TreeNode('B'), TreeNode('C'))`。

```
# coding:utf-8
```

```
# 节点通过类定义（这里是新式类）
```

```
class TreeNode(object):
```

```
    # data 节点的数据
```

```
    # left 左节点 right 右节点
```

```
    # left = None right = None 默认生成的是叶节点
```

```
    def __init__(self, data, left=None, right=None):
```

```
        self.data = data
```

```
        self.left = left
```

```
        self.right = right
```

```
    # 重写 str 方法
```

```
    # print TreeNode 对象的时候，打印出相应的 data
```

```
    def __str__(self):
```

```
        return str(self.data)
```

```
if __name__ == "__main__":
```

```
    # 通过列表拆包的形式创建多个节点（此时节点都没有左节点与右节点，都是孤立的）
```

```
    A, B, C, D, E, F, G, H, I = [TreeNode(x) for x in 'ABCDEFGHI']
```

```
    A.left = B
```

```
    A.right = C
```

```
    B.right = D
```

```
    C.left = E
```

```
    C.right = F
```

```
    E.left = G
```

```
    F.left = H
```

```
F.right = I
```

```
# 打印出 A.left.data  
print A.left
```

- **And So On**