









Implementing a Binary Search Tree in Python

In this lesson, we'll implement a very basic Binary Search Tree in Python



- Introduction
 - The Node Class
 - The BinarySearchTree class
 - Putting the two together

Introduction

The Node Class

To implement a BST, the first thing you'd need is a node. A node should have a value, a left child, a right child, and a parent. This node can be implemented as a Python class and here is the code.

```
Node.py
```

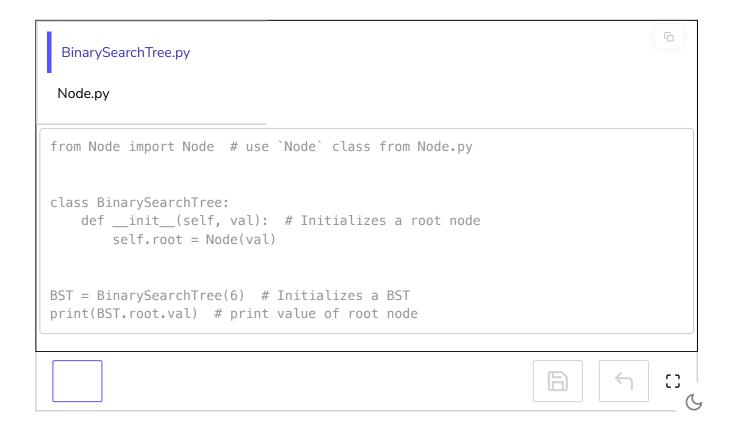


The **BinarySearchTree** class

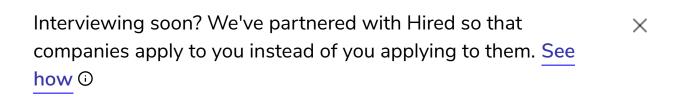
You can then choose to create a wrapper class for the tree itself; this can sometimes make your code cleaner and easier to read, but not always. However, this is a programming convention so let's create a tree class:

Putting the two together

When both classes are put together, you get a BST. Let's try running this.



Now that we have some bare bones code for binary search translated a high-level algorithm and code to insert values into a BST!





! Report an Issue

