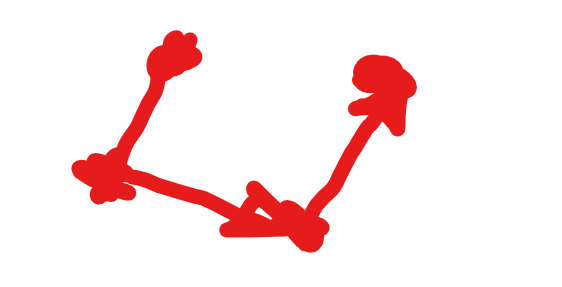
Un árbol es un grafo acíclico dirigido donde cada nodo tiene un solo nodo como padre, no puede tener más de uno, a excepción del nodo raíz, que no tiene padres  
Un nodo puede tener cualquier numero de descendientes, piénsalo como un tipo de árbol genealógico  
Al primer nodo se le llama Raíz (Root)  
A través de este mismo, se puede acceder a los demás nodos de manera indirecta  
Si un nodo no tiene descendientes, se le llama hoja (leaf, leaves)  
Dos nodos con el mismo padre se le llama hermanos (siblings)  
Un árbol es una estructura recursiva, cada nodo del árbol es a su vez, un árbol, o que “un árbol es un subconjunto de árboles”



**Aplicaciones**

Se pueden usar en compiladores (árbol sintáctico), en minería de datos (arboles de decisión), en juegos (alfa-beta), para búsqueda y ordenación de datos, el sistema de archivos tiene estructura de árbol, DOM HTML/XML: la estructura de etiquetas sigue una estructura de árbol

**Tipos**

En este curso se verán dos tipos: Generales y Binarios de búsqueda

Generales: Cada nodo puede tener cualquier numero de hijos, y no cumplen ninguna propiedad en especial

Binarios: Cada nodo puede tener a lo mucho dos hijos, están pensados para indexación rápida de datos, por ejemplo se utilizan en bases de datos para recuperar de forma eficiente registros  
Tiene restricciones, cada nodo tiene un índice numero, el hijo de la izquierda vale menos que el padre, y el del lado de la derecha vale mas que el padre

Python no cuenta en su biblioteca estándar con esta estructura de datos, por lo que se debe hacer a mano. Se vera una implementación basada en objetos y clases. La implementación se divide en 2 clase, una que represente cualquier nodo del árbol, una que representa el árbol al si. Se realizará implementación gradual en clases

**Operaciones**

**Crear árbol, agregar hijo, es hoja, recuperar padre, imprimir árbol, borrar hijo(podar)**

**Recorrer arboles**

Por profundidad: Recorrer siempre primero a la izquierda hasta llegar a una hoja, al llegar a una hoja regresar un paso, ir a la derecha y continuar por la izquierda, el control se puede hacer mediante una pila (con solo recursividad se puede)

Por primero en amplitud: se recorre en niveles