AVL Tree Rotations with ASCII Examples

1️⃣ Left-Left (LL) Rotation

Occurs when a node is inserted into the **left subtree of the left child** of an unbalanced node.

Before Rotation (Unbalanced at 30)

```
    30
   /
  20
 /
10
```

🔴 **Imbalance at 30 (Left-heavy)**
✅ **Perform Right Rotation (RR) on 30**

After Rotation (Balanced)

```
   20
  /  \
 10   30
```

2️⃣ Right-Right (RR) Rotation

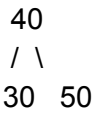
Occurs when a node is inserted into the **right subtree of the right child** of an unbalanced node.

Before Rotation (Unbalanced at 30)

```
30
  \
   40
    \
     50
```

🔴 **Imbalance at 30 (Right-heavy)**
✅ **Perform Left Rotation (LL) on 30**

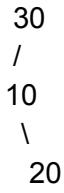
After Rotation (Balanced)



3️⃣ Left-Right (LR) Rotation

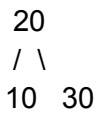
Occurs when a node is inserted into the **right subtree of the left child** of an unbalanced node.

Before Rotation (Unbalanced at 30)



🔴 **Imbalance at 30 (Left-Right case)**
✅ **Perform Left Rotation (LL) on 10, then Right Rotation (RR) on 30**

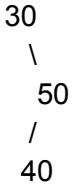
After Rotation (Balanced)



4️⃣ Right-Left (RL) Rotation

Occurs when a node is inserted into the **left subtree of the right child** of an unbalanced node.

Before Rotation (Unbalanced at 30)



🔴 **Imbalance at 30 (Right-Left case)**
✅ **Perform Right Rotation (RR) on 50, then Left Rotation (LL) on 30**

After Rotation (Balanced)