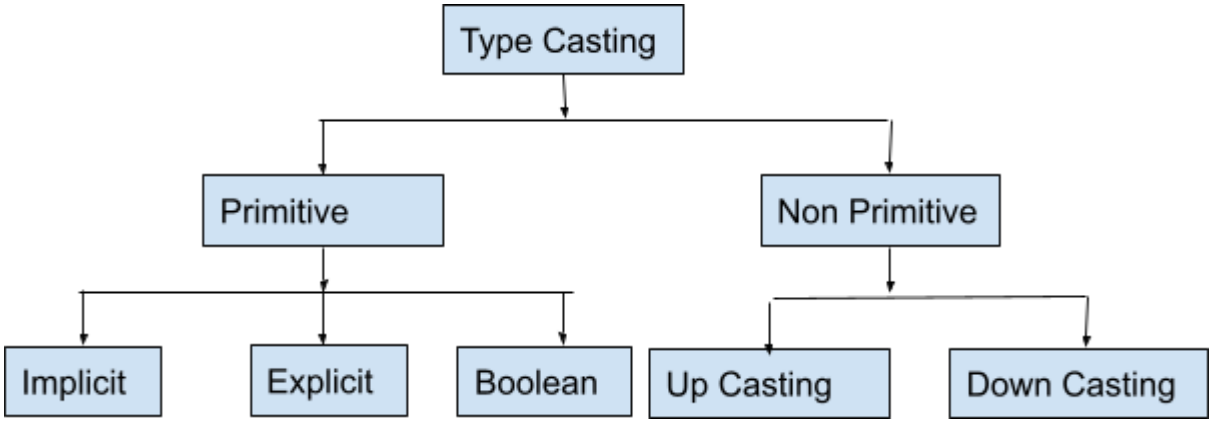
Type Casting
==========

In casting programmer can convert one type of information into another type of information, there are two basic types of casting



1. **Primitive Casting**
    - In primitive casting one datatype of information is converted into another data type of information
    - It is related to variables which are further divided into three types
    - Implicit, Explicit and Boolean type.

—----------

a. **Implicit Casting / widening casting**

- In Implicit casting, lower data type of information is converted into higher data type of information.
  byte -> **short** -> **char** -> **int** -> **long** -> **float** -> **double**
- Syntax

```
int a = 10;          // integer type of data type
double x = a;        // converted into float type of data type
```

—----------

b. **Explicit Casting / narrowing casting**

- In Explicit casting, higher data type of information is converted into lower data type of information.
  double -> **float** -> **long** -> **int** -> **char** -> **short** -> **byte**
- Syntax

```
double d = 10.3;     // double type of data type
int y = (int)d;      // converted into integer type of data type
```

—----------

c. **Boolean Casting** >> it is not supported in java

—----------

E.x

```
public class Variables
{
        static int a = 15;
        static double d = a;            // implicit casting

        static double b = 25.21;
        static int i = (int)b;          // explicit casting

        public static void main(String[] args)
        {
                System.out.println("Before casting" +"  "+ a);
                System.out.println("After casting" +"  "+ d);

                System.out.println("Before casting" +"  "+ b);
                System.out.println("After casting" +"  "+ i);
        }
}
```

Output > Before casting 15
After casting 15.0
Before casting 25.21
After casting 25

—----------

Non Primitive Casting / Object type casting
==========

- In this casting one type of class information is converted into another type of class information.
- There are two types 1. Up Casting, 2. Down Casting

Up Casting

- In this casting subclass properties are assigned to super class.
- To perform upcasting inheritance must be performed.
- In up casting parent reference is used to hold child class objects.
- In upcasting only those properties are upcasted which are inherited from superclass.
- New properties declared in subclass can not be upcasted to super class.

E.x **In normal object**

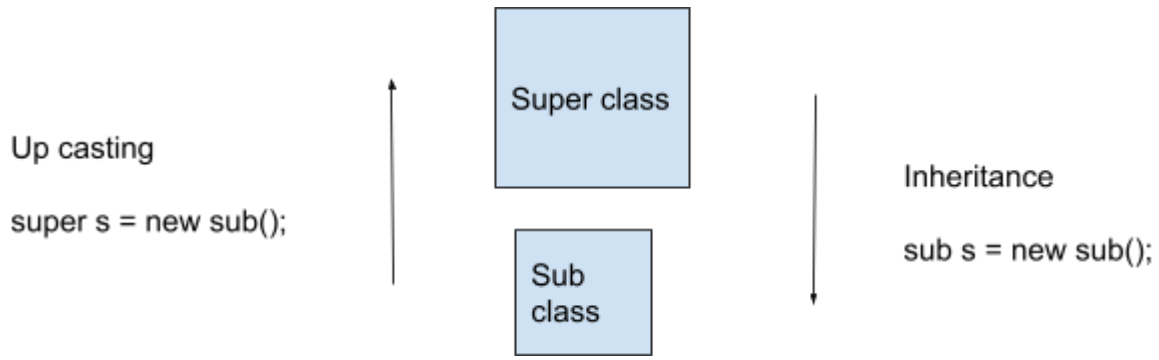
sub sb = new sub();

In a normal class object > a programmer is able to call both subclass as well as super class methods.

In Up casting

super s = new sub();

In Up casting, a programmer is able to call only parent class methods by using parent class reference variable.



—----------

Down Casting

- It is not supported in java.
- To perform down casting, up casting operation must be performed first.

—----------

E.x Up casting

Super class >>

```
public class aclass
{

        public void test1()                    // Overridden method
        {
                System.out.println("test 1 from super class");
        }
}
```

—----------

Sub class >>

```
class sclass extends aclass
{
        public void test1()
        {
                System.out.println("test1 from  sub class");
        }

        public void test2()
        {
                System.out.println("test2 from sub class ");
        }


        public static void main(String[] args)
        {
                // subclass reference           //sub class object

                        sclass sc       =       new sclass();  // Object creation

                        sc.test1();             // calling the overridden methods of super class
                        sc.test2();             // calling the subclass specific method

                // parent class reference        // sub class object

                        aclass ac       =       new sclass();  // Up casting

                        ac.test1();             // calling the overridden methods of super class
        }
}
```

—------

Difference between Sub s = new Sub(); and Super s = new Sub();

Sub s = new Sub();	**Super s = new Sub();**
1. If we know the exact runtime type of object then we should use this approach.	If we don't know the exact runtime type of object then we should use this approach.
1. By using child reference we can call both parent and child class methods.	By using parent reference we can call only methods available in parent class and child specific methods we can't call.
2. We can use child reference to hold only for that particular child class object only.	We can use parent reference to hold any child class object.

Note :

1. **Implicit Casting is also called as widening casting**
2. **Explicit Casting is also called as narrowing casting**