**OOPs concepts – What is Aggregation in java?**

Aggregation is a special form of [**association**](http://beginnersbook.com/2013/05/association/). It is also a relationship between two classes like association, however its a **directional** association, which means it is strictly a **one way association.**It represents a **Has-A** relationship.

For example consider two classes Student class and Address class. each student must have an address so the relationship between student and address is a Has-A relationship. But if you consider its vice versa then it would not make sense as an Address doesn’t need to have a Studentnecessarily. Below example shows this theoretical explanation in a sample java program.  
Student Has-A Address

class Address

{

int streetNum;

String city;

String state;

String country;

Address(int street, String c, String st, String coun)

{

this.streetNum=street;

this.city =c;

this.state = st;

this.country = coun;

}

}

class StudentClass

{

int rollNum;

String studentName;

Address studentAddr;

StudentClass(int roll, String name, Address addr){

this.rollNum=roll;

this.studentName=name;

this.studentAddr = addr;

}

public static void main(String args[]){

Address ad = new Address(55, "Agra", "UP", "India");

StudentClass obj = new StudentClass(123, "Chaitanya", ad);

System.out.println(obj.rollNum);

System.out.println(obj.studentName);

System.out.println(obj.studentAddr.streetNum);

System.out.println(obj.studentAddr.city);

System.out.println(obj.studentAddr.state);

System.out.println(obj.studentAddr.country);

}

}

Output:

123

Chaitanya

55

Agra

UP

India

The above example shows the **Aggregation** between Student and Address classes. You can see that in Student class I have used Address class to obtain student address. Its a typical example of Aggregation in Java.

**Why we need Aggregation?**

**To maintain code re-usability**. To understand this lets consider the above example again. Suppose there are two other classes College and Staffalong with above two classes Student and Address. In order to maintain Student’s address, College Address and Staff’s address we don’t need to use the same code again and again. We just have to use the reference of Address class while defining each of these classes like:

Student Has-A Address (Has-a relationship between student and address)

College Has-A Address (Has-a relationship between college and address)

Staff Has-A Address (Has-a relationship between staff and address)

Hence we can improve code re-usability by using Aggregation relationship.