Software Engineering
Prof. Rajib Mall
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

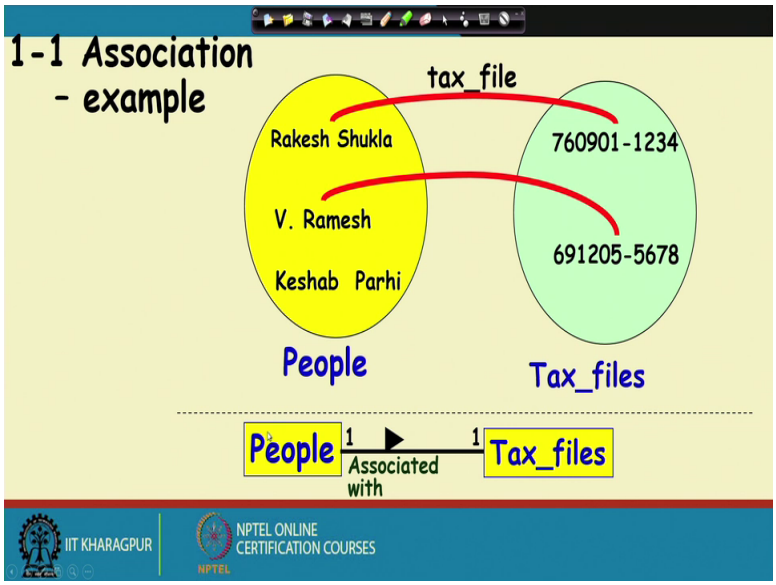
Lecture - 35
Association Relationship

Welcome to this lecture. In the last lecture we had discussed about the different types of relationship that can exist among the classes. And, then we said that there are 4 types of relationship that can exist among classes, that is inheritance, association, aggregation, composition and dependency.

We, look that the inheritance relationship with some examples and we saw that inheritance is a powerful mechanism and also easily implemented in code, just by using a keyword we can implement the inheritance relationship. And, then we are started to look at the association relationship, we are looking at how to represent that in the UML syntax and also how to implement it in code and what is the implication of the association relationship between classes?

Now, let us proceed further understanding the association relationship.

(Refer Slide Time: 01:33)



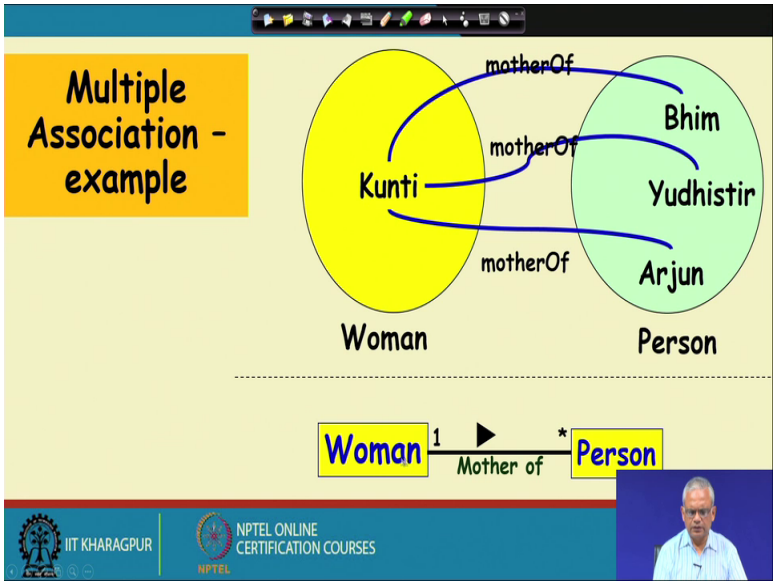
The association relationship is represented by a line here and sometimes by an arrow will see later when and we write a multiplicity number here. So, we read here and this is the

name of the association, the person or the people associated with the tax file. One person associated with one tax file or one tax file associated with one person.

If, we look at the corresponding objects of this people, we will see that there are many objects Rakesh Shukla. V. Ramesh, Keshab Parhi etcetera, and then the tax files are having different numbers. So, each person here each object people object here is associated with the tax file.

But, then there can be some objects here, which do not have which are not associated with any tax file so that is permitted by this relationship. As you proceed it becomes clear that why that is the case that some of the objects, people object is associated with a tax file, but some are not.

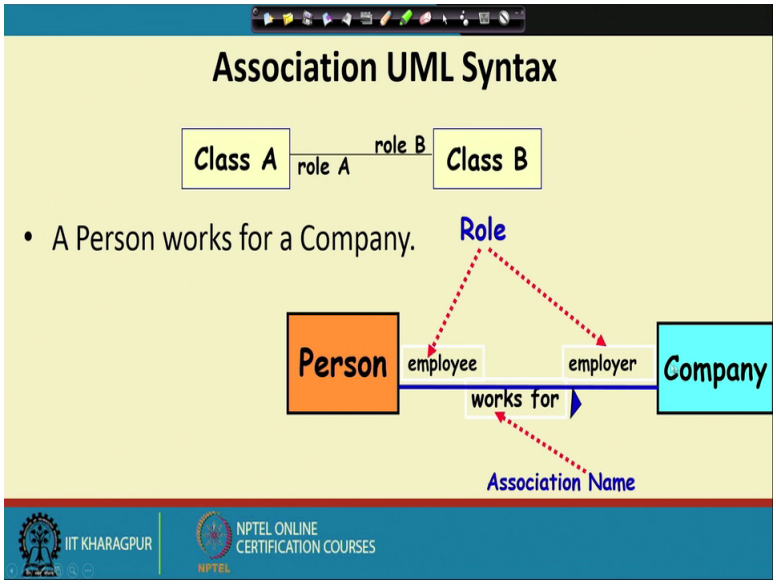
(Refer Slide Time: 03:03)



Here, woman a woman is mother of many persons, but we can also read it as a person has a woman as mother. Please pay attention we will I will read it again, because this is important. We read this relationship, this association relationship between woman and a person in the name of the association relationship is mother of. We can read it as a woman is the mother of many persons a woman is the mother of many persons, that is the implication of the star.

We can also read it as a person has one woman as his mother. If, we look at the corresponding object diagram that becomes clear the woman class is instantiated into many woman.

(Refer Slide Time: 04:24)



And, one of the woman here is Kunti who is the mother of many persons here Bhim, Yudhistir, Arjun etcetera, but Arjun has exactly one mother, Yudhistir also has exactly one mother.

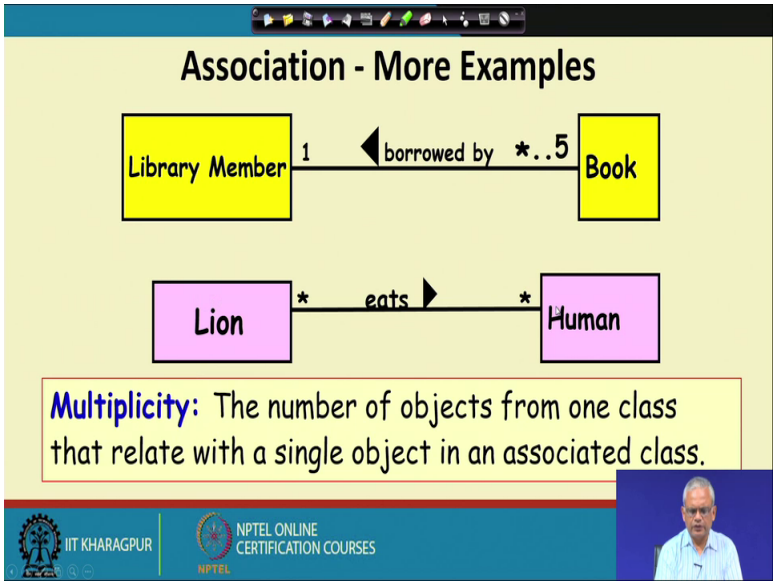
So, that is what it says a person has one woman as mother, but woman can have many persons woman is the mother of many persons, this is the syntax in UML. We write the 2 classes which are related by the association relationship, we draw a line here indicating the association relationship and we write the role here, we will see, what exactly is the role?

For example, a person and a company are associated, because the person works for the company the name of the association relationship is works for. We represented in the form person works for this the reading direction; name of the association is works for these are the 2 classes. And, the role of the person is the employee and the role of the company is the employer.

We can write that or we may not write, but then in the case tools we will see that if we write this 2 it helps this become attributes of the corresponding class. The person works

for a company and the role of the person is employee and the role of the company is the employer.

(Refer Slide Time: 06:22)

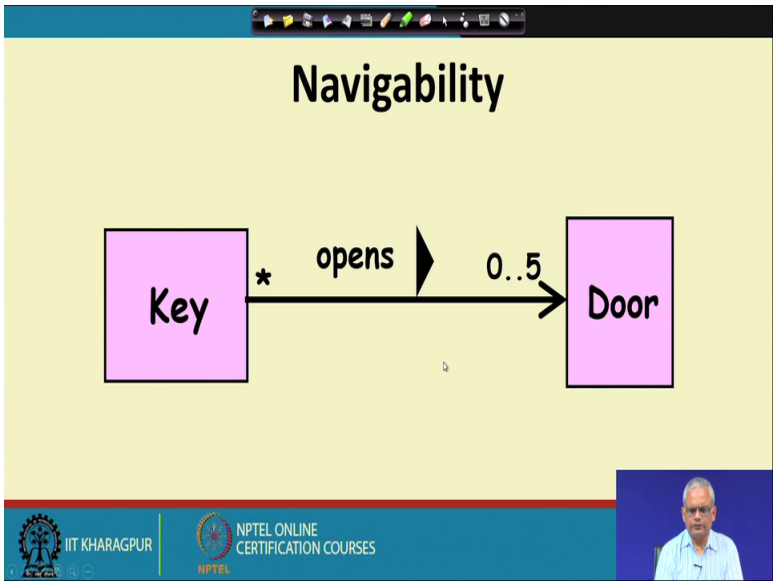


These are more examples of association, how do we read this association? A member borrows up to 5 books. A book is borrowed by a library member; a member borrows up to 5 books ok. I should have written 0 here instead of star let me just change that it should be 0, 0 to 5. A member borrows up to 5 books or we can read in this direction, that a book is borrowed by exactly 1 library member.

Now this one, a lion eats many humans, a human is eaten by many Lions. So, the multiplicity that we write here, it indicates a single object here is associated with how many objects on the other side? This is important; the multiplicity indicates one object here is associated with how many objects and this side?

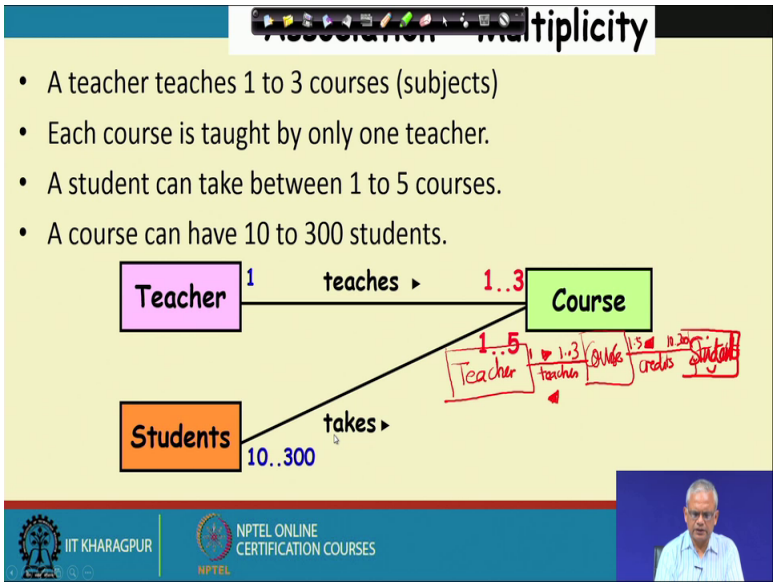
A, lion eats many humans, a single object on this side a human is eaten by many lions. Similarly, here the multiplicity here is 1 and here it is 0 to 5. A, library member that is a single object here borrows up to 5 books, a single object here a book is borrowed by a member.

(Refer Slide Time: 08:37)



Now, let us look at the arrow here if we put arrow here; that means, that the key can invoke a method on the door, but the door cannot it is navigable from key to door, but not the vice versa. If, it is a simple line here or a bidirectional arrow the Navigability's on both directions. Now, let us see how you can read here, a key that is a single object here, a key opens up to 5 doors, a door is opened by many keys.

(Refer Slide Time: 09:34)



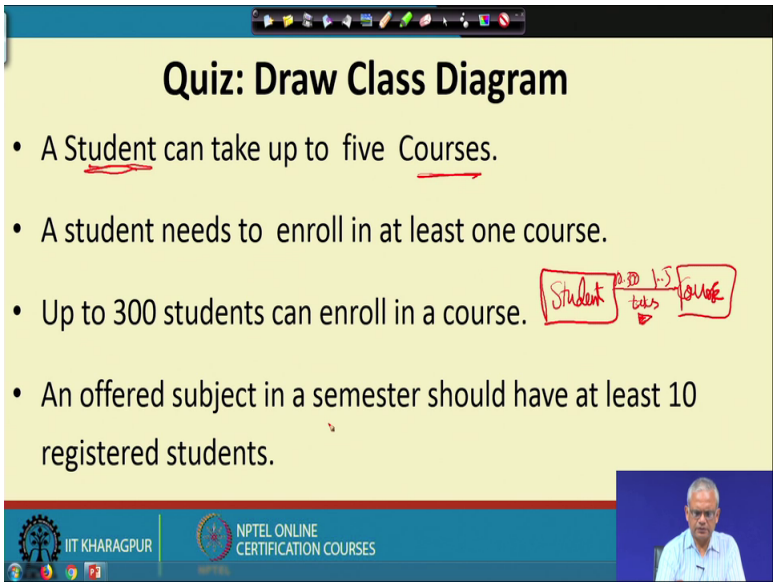
Now, let us see how to draw a class diagram given this description. A, teacher teachers 1 to 3 courses, each course is taught by one teacher. A student can take between 1 to 5 course and a course can have 10 to 300 student.

So, the classes here a teacher, course and subject, a teacher teaches 1 to 3 courses. A, course is taught by exactly one teacher. Teacher teachers up to 1 to 3 course this the reading direction, to write the reading direction with this direction, then we write course taught by to write this is the direction, if we give will write the relation the association is taught by and the next one is student can take between 1 to 5 courses.

So, it is the student, this the student a student can take up to 1 to 5 courses and a course can have 10 to 300 students. And, we can write the name of the association relationship a student credits 1 to 5 courses, let me put the reading direction and this credits.

So, that is the same diagram here a teacher teaches 1 to 3 courses. And a student takes 1 to 5 courses or a course is taken by 10 to 300 students. Given a description we should be able to identify the classes and the association relationship, write the multiplicity and the association in the reading direction.

(Refer Slide Time: 12:27)



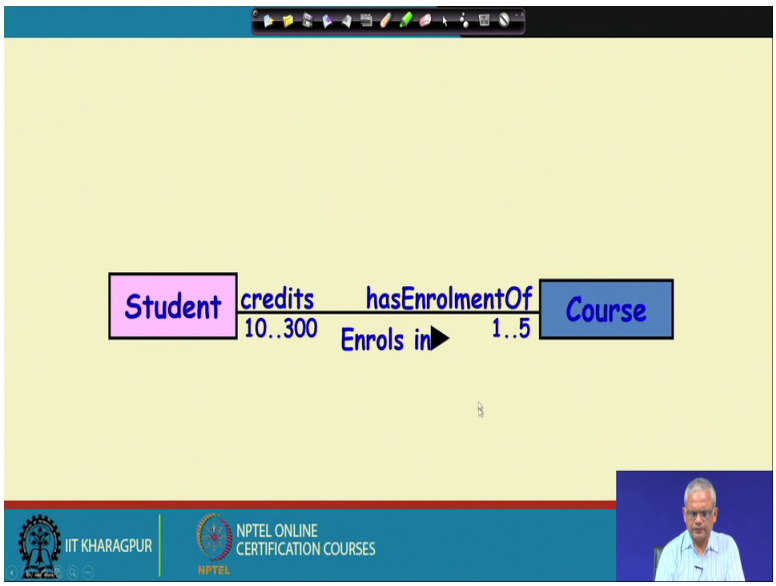
Now, please do a small quiz, we need to draw the class diagram a student can take up to 5 courses. Student names to enroll in at least one course, up to 300 students can enroll in

a course. And, an offer subject in a semester should have at least 10 registered students please try to draw the class diagram for this.

The, first thing is to identify what are the classes here. Student is a class and course is a class and we represent them a student and the course, a student can take up to 5 courses and need to at least take one course. So, we write 1 to 5 and student takes write the reading direction here, up to 300 students can enroll in a course and it must have at least 10 students.

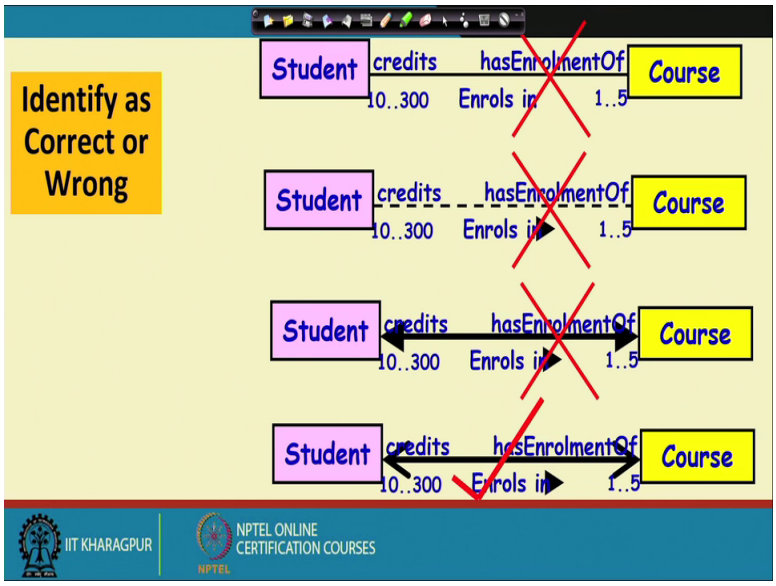
So, we write the multiplicity 10 to 300. So, this will be the class diagram for this description.

(Refer Slide Time: 14:11)



So, that is what we have written student takes are enrolls in 1 to 5 courses. And, a course has enrollment of 10 to 300 students. And, we can also write the role the student credits and the course has enrollment of.

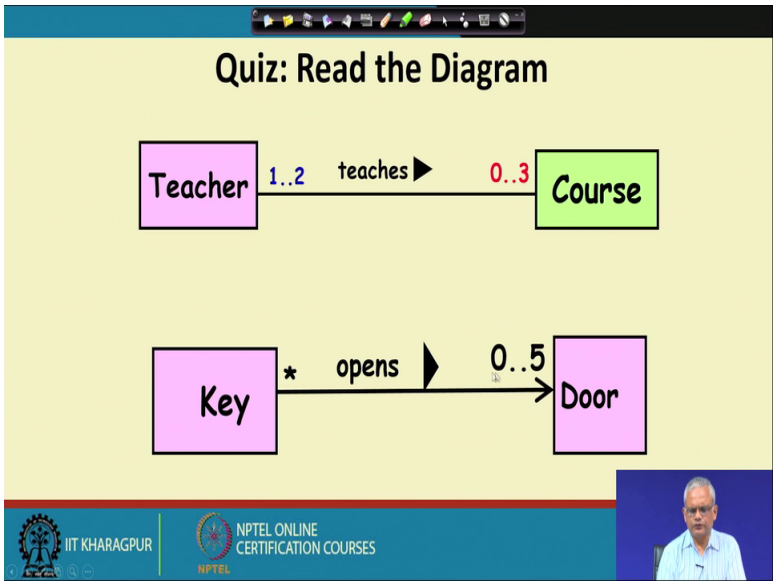
(Refer Slide Time: 14:32)



How do I identify, which diagram is correct? Let us look at this diagram this is a round diagram, what is the mistake in this diagram? The mistake in this diagram is that we have not given the reading direction and in this diagram, what is the mistake here given the reading direction here, but then the line association line is round used a dotted line.

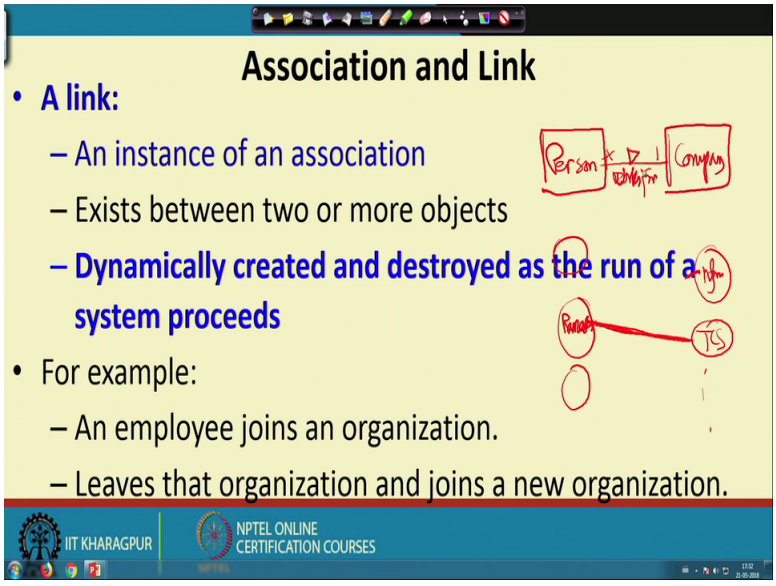
What about this diagram this is a round diagram, we have used a wrong arrow type. This is a correct diagram is a bidirectional arrow, which is also equal to just as, which is just equivalent to a simple line, a simple line connecting to classes it is same as their being connected by a bidirectional arrow of this form.

(Refer Slide Time: 15:57)



Now, there is another quiz here. Please read this diagram, 2 classes teacher and course and we have the multiplicity that are written here and the association relationship and reading direction, how do we read this diagram? We, can read as a teacher teaches up to 3 courses. A, course is taken by either one or 2 teachers, a teacher teaches up to 3 courses, a course it is taught by either 1 or 2 teachers. Now, let us try to read this diagram. A key opens up to 5 doors; a door is opened by many keys.

(Refer Slide Time: 17:05)



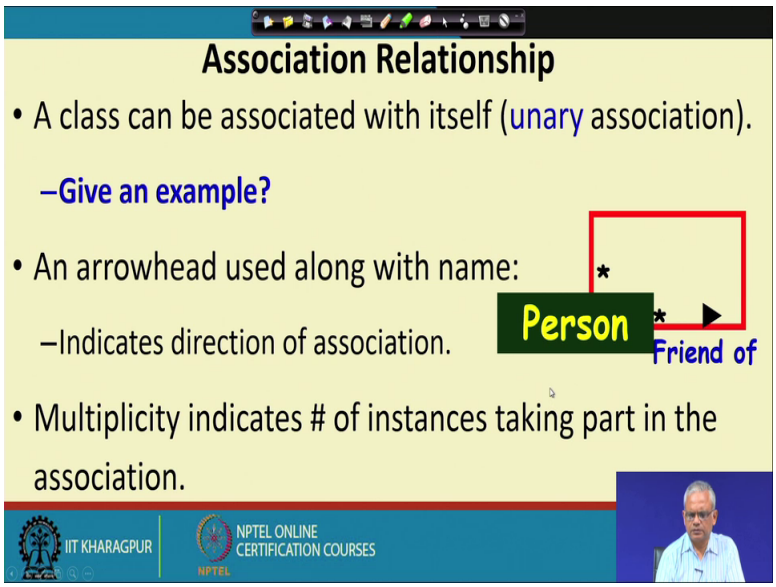
Now, let us look at the concept of a link, so far seen that the association relationship exists between classes. And, the objects of the 2 classes, which are associated a link exists between them. And therefore, we can say that a link is an instance of an association exists between 2 or more objects. The association relationship between classes is a static in nature, we always represent the association. Where, as the objects as the system runs the software executes objects may get the links between objects may get dynamically created or destroyed.

For example, let us say we have an employee class and a company class. Let us say person works for a company, a person works for a company and write the reading direction. And, company has many employees write the star here, but when we look at the object diagram let us an object let us say Ramesh and there are other objects and there are many companies let us say Infosys, TCS, etcetera.

Let us say the company the person Ramesh works for Infosys a link exist between these 2 objects, but as the system runs let us say Ramesh leaves Infosys and joins TCS. Then, this link gets dissolved and at the moment they have no link and let us say after few months he joints TCS. So, link gets formed between Ramesh and TCS.

So, the association relationship is static in nature between 2 classes, whereas the links are more dynamic in nature. And, these exist between the corresponding associated classes, objects of the associated classes.

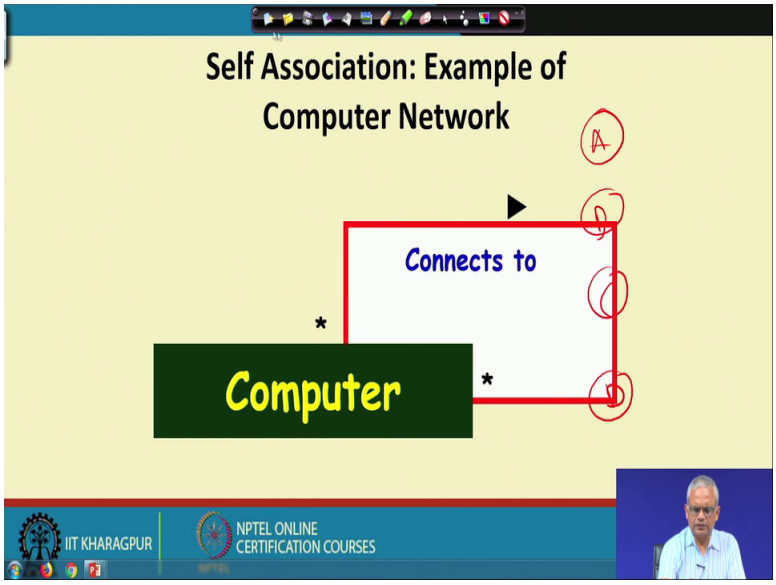
(Refer Slide Time: 20:01)



But, can association be unary that is we have a single class and association relationship is defined and that same class. A, single class and the association relationship is defined on that same class is it possible, if yes give an example ok. There are many examples possible, we can write the multiplicity and we can use the arrowhead to indicate the direction of the association. And, one example here is a person is a friend of many persons or a person has many persons as his friend. A, person has many persons as his friend or a person be friends many persons.

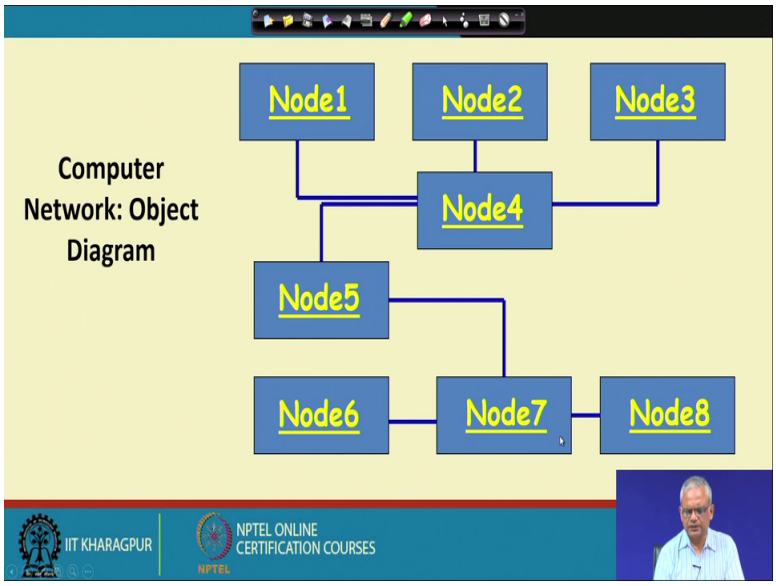
This is an example of a unary association; there can be many examples.

(Refer Slide Time: 21:24)



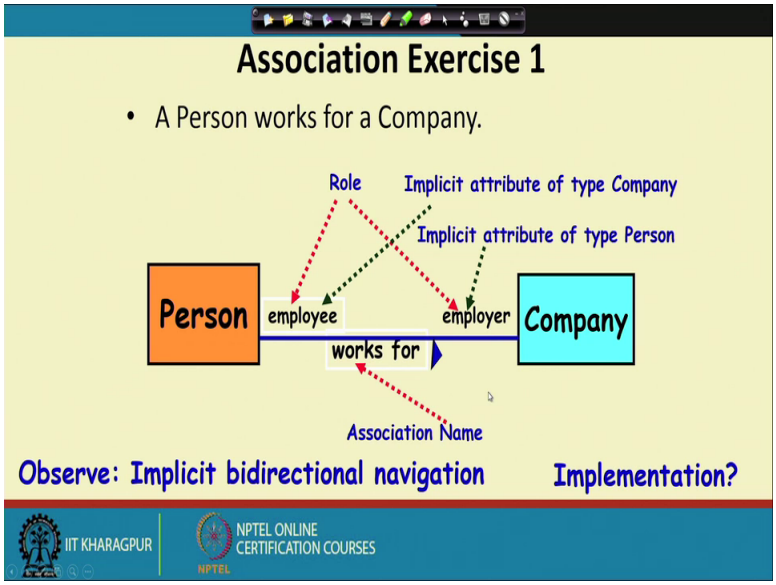
This is another example, a computer connects to many other computer or a computer is connected by many other computer, but how will the object diagram of this look like. A computer connects to many other computers, or a computer is connected by many other computers. How, will the object diagram, if you have many computers, these are the computers A B C D etcetera, how will the object diagram of this look like?

(Refer Slide Time: 20:20)



So, this is an object diagram, we have the computer name node 1, node 2, node 3, node 4, 5, 6, 7, 8. So, node 1 is connected to node 4, node 5 is connected to node 4. So, the node 4 is connected by 3 computers, whereas node 4 ok 4 computers, but node 6 is connected by 1 computer.

(Refer Slide Time: 22:56)

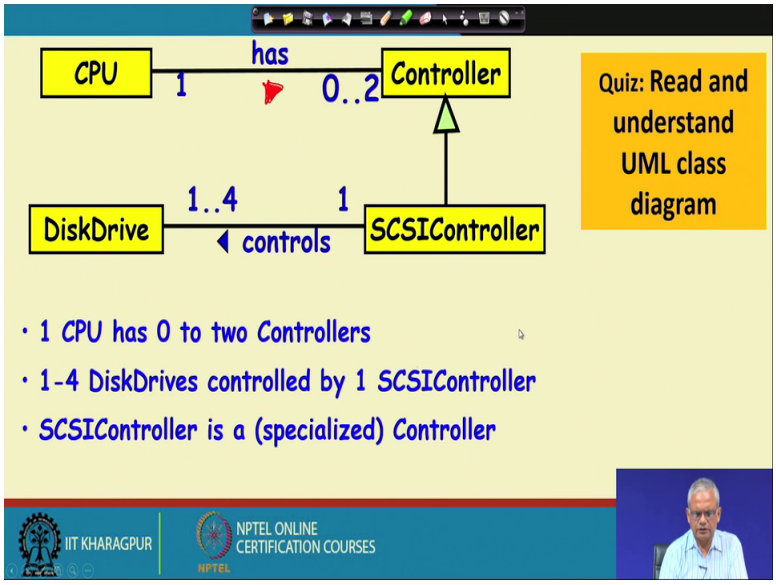


Now, let us do one excise, let us develop the class diagram. A, person works for a company withdraw the person class and the company class, and then draw the association relationship works for, and the role is employee here, the role of the company

is the employer, these are the implicit attributes as we will see that when there is a association relationship, we need to create some implicit attributes, which will look at the very soon and this is the name of the association.

Let us see the implementation of this that if this is the class diagram that a person and a company, works for his association relationship. We will see the implementation, that how we can write the code for such a diagram, then it will become clear, that the roles become the implicit attribute type of the company. So, this becomes a attribute type of company and this become the attribute type of the person.

(Refer Slide Time: 24:20)

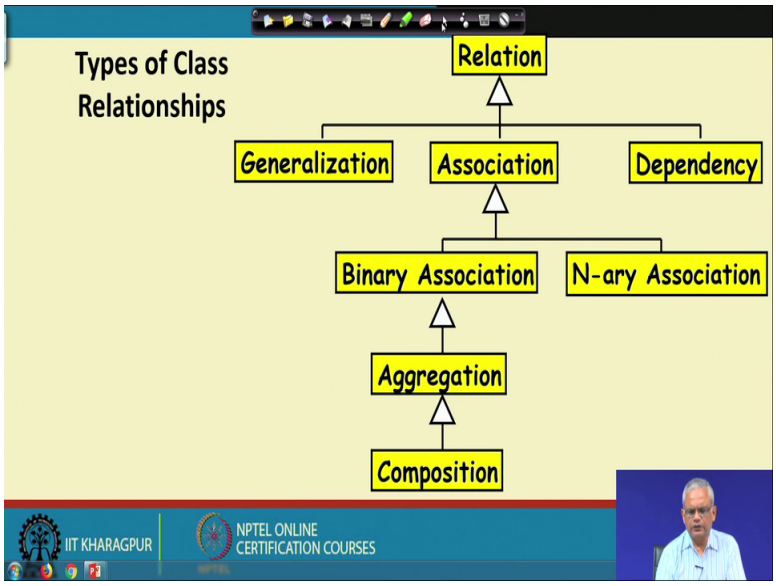


But, before that there is an another quiz here, please read the following diagram, if this is the diagram given how do you read it? Ok.

We can read here and not run the reading direction here missed it just a draw the reading direction here ok. So, a CPU has up to 2 controllers. A controller is connected to 1 CPU. The, controller can be a quasi-controller, if you had other inheritance relationship would have other types of controllers, but a quasi controller is a special type of controller.

And, a controller controls up to 1 to 4 diskdrives or we can also read a diskdrive is controlled by 1 quasi controller. A, CPU has 0 to 2 controllers 1 to 4 diskdrives are controlled by ones quasi controller, ones quasi controller, controls 1 to 4 diskdrives as quasi controller is a special type of controller.

(Refer Slide Time: 26:13)



So for we looked at the relations between classes, generalization, association, aggregation composition and dependency are not looked at and association can be binary association or n-ary association and a special type of binary association is aggregation and a special type of aggregation is composition. We are at the end of this lecture in the next lecture we look at aggregation composition and the dependency relations, we will stop here.

Thank you.