**BA/DA TRAINING**

**UML Diagram Assignment – 6(July12,2019)**

**SHAFALI GUPTA**

1. **Explain Component diagram, object diagram and deployment diagram with example.**

**Ans:- Component diagram-** Component diagram shows components, provided and required interfaces, ports, and relationships between them. The artifacts that implement component are intended to be capable of being deployed and re-deployed independently, for instance to update an existing system.

Components in UML could represent:

* **logical components** (business components, process components)
* **physical components** (e.g. CORBA components, EJB components, COM+ and .NET components, WSDL components, etc.)

Use component diagrams when you are dividing your system into components and want to show their interrelationships through interfaces or the breakdown of components into a lower-level structure.

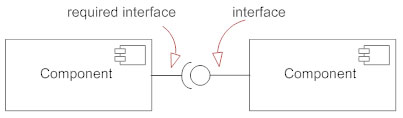
**Component**

A component is a logical unit block of the system, a slightly higher abstraction than classes. It is represented as a rectangle with a smaller rectangle in the upper right corner with tabs or the word written above the name of the component to help distinguish it from a class.



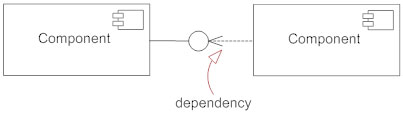
**Interface**

An interface (small circle or semi-circle on a stick) describes a group of operations used (required) or created (provided) by components. A full circle represents an interface created or provided by the component. A semi-circle represents a required interface, like a person's input.



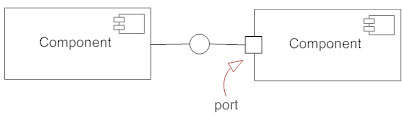
**Dependencies**

Draw dependencies among components using dashed arrows.



**Port**

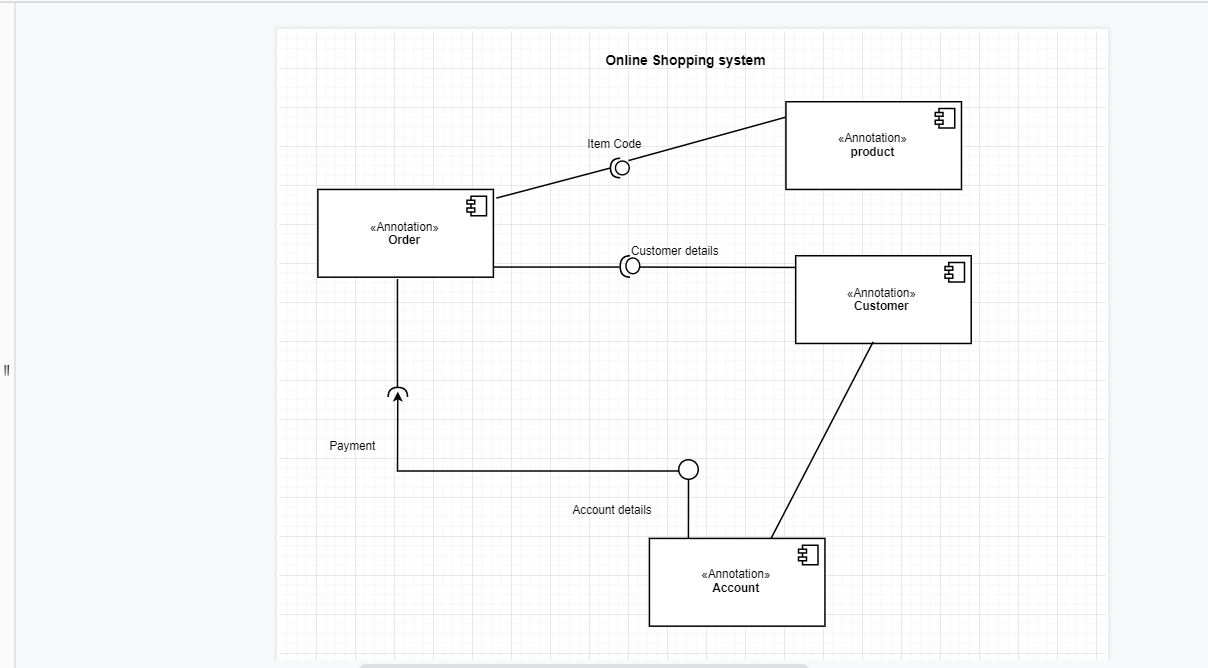
Ports are represented using a square along the edge of the system or a component. A port is often used to help expose required and provided interfaces of a component.



**How to Draw a Component Diagram**

* Take stock of everything needed to implement the planned system. For example, for a simple e-commerce system, you'll need components that describe products, orders, and customer accounts.
* Create a visual for each of the components.
* Describe the organization and relationships between components using interfaces, ports, and dependencies.

**Example of Component Diagram- Online shopping system**



**Object Diagram**

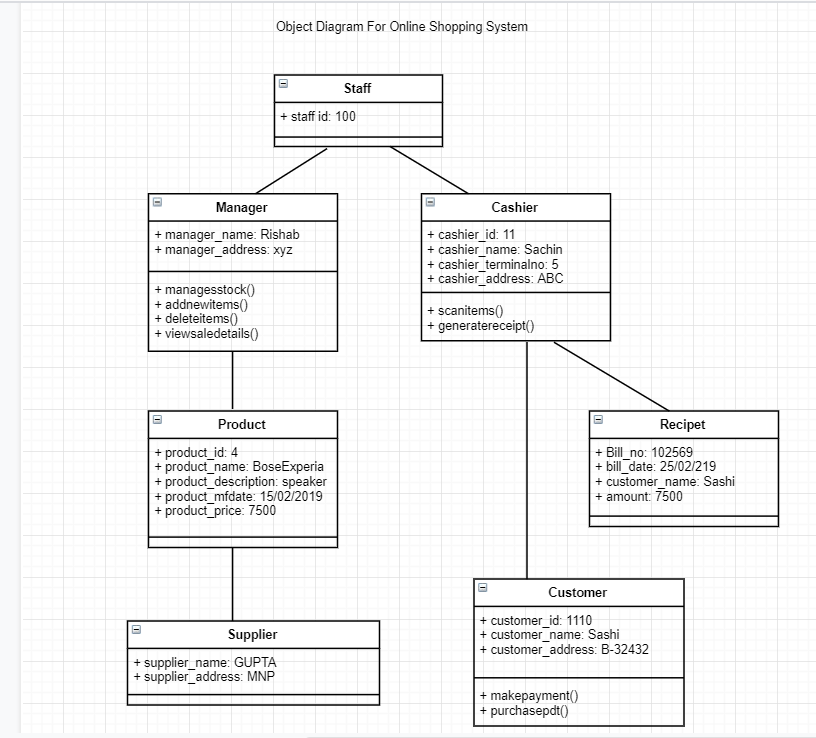
Object diagrams describe the static structure of a system at a particular time. They can be used to test class diagrams for accuracy. An object diagram in the Unified Modeling Language (UML), is a diagram that shows a complete or partial view of the structure of a modeled system at a specific time.

Object Diagrams use real world examples to depict the nature and structure of the system at a particular point in time. Since we can use data available within objects, Object diagrams provide a clearer view of the relationships that exist between objects. The purpose of the object diagram can be summarized as −

* Forward and reverse engineering.
* Object relationships of a system.
* Static view of an interaction.
* Understand object behavior and their relationship from practical perspective.

They are be useful to explain smaller portions of your system, when your system class diagram is very complex, and sometimes modeling recursive relationship in diagram.

**Example of Object Diagram- Online shopping system**



**deployment diagram**

Deployment diagrams depict the physical resources in a system, including nodes, components, and connections. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships. Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are useful for system engineers. An efficient deployment diagram is very important as it controls the following parameters-

* Performance
* Scalability
* Maintainability
* Portability

**Example of deployment Diagram- Vehicle Registration system**

