Day 26 - 13th Aug 2025

Name : Aravind Kasanagottu

Id: mvsnarav

Task 01:

**UML Class Diagram Relationships**

Here is a concise overview of the most common relationships used in UML class diagrams:

**1. Composition**

* **Symbol:** A solid diamond on a solid line.
* **Meaning:** A strong "has-a" relationship where the parts cannot exist without the whole. The lifetimes are dependent.
* **Example:** A House and a Room; if the house is destroyed, the room is too.

**2. Aggregation**

* **Symbol:** A hollow diamond on a solid line.
* **Meaning:** A weak "has-a" relationship where the parts can exist independently of the whole.
* **Example:** A School and a Teacher; the teacher can exist even if the school closes.

**3. Association**

* **Symbol:** A solid line.
* **Meaning:** A general connection between two classes.
* **Example:** A Student is associated with a Course.

**4. Directed Association**

* **Symbol:** A solid line with an arrowhead.
* **Meaning:** One class knows about or uses another class.
* **Example:** A Customer knows about and places an Order.

**5. Dependency**

* **Symbol:** A dashed line with an arrowhead.
* **Meaning:** A temporary or "uses" relationship where one class depends on another for some functionality.
* **Example:** A ReportGenerator depends on a Printer.

**6. Generalization (Inheritance)**

* **Symbol:** A solid line with a hollow arrowhead.
* **Meaning:** An "is-a" relationship where a subclass inherits from a superclass.
* **Example:** A Dog is an Animal.

**7. Realization (Implementation)**

* **Symbol:** A dashed line with a hollow arrowhead.
* **Meaning:** A class implements the methods of an interface.
* **Example:** A Car implements the Vehicle interface.

Task02:

## UML Class Diagram Relationships

Here is a concise overview of the most common relationships used in UML class diagrams:

### 1. Composition

* **Symbol:** A **solid diamond** on a solid line.
* **Meaning:** A strong "has-a" relationship where the parts cannot exist without the whole. The lifetimes are dependent.
* **Example:** A **House** and a **Room**; if the house is destroyed, the room is too.

### 2. Aggregation

* **Symbol:** A **hollow diamond** on a solid line.
* **Meaning:** A weak "has-a" relationship where the parts can exist independently of the whole.
* **Example:** A **School** and a **Teacher**; the teacher can exist even if the school closes.

### 3. Association

* **Symbol:** A **solid line**.
* **Meaning:** A general connection between two classes.
* **Example:** A **Student** is associated with a **Course**.

### 4. Directed Association

* **Symbol:** A solid line with an **arrowhead**.
* **Meaning:** One class knows about or uses another class.
* **Example:** A **Customer** knows about and places an **Order**.

### 5. Dependency

* **Symbol:** A **dashed line** with an arrowhead.
* **Meaning:** A temporary or "uses" relationship where one class depends on another for some functionality.
* **Example:** A **ReportGenerator** depends on a **Printer**.

### 6. Generalization (Inheritance)

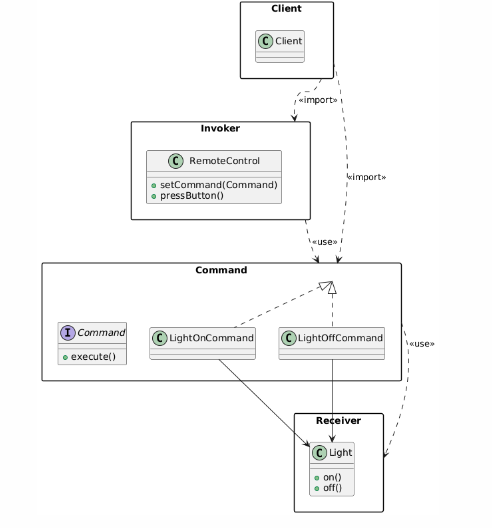
* **Symbol:** A solid line with a **hollow arrowhead**.
* **Meaning:** An **"is-a"** relationship where a subclass inherits from a superclass.
* **Example:** A **Dog** is an **Animal**.

### 7. Realization (Implementation)

* **Symbol:** A **dashed line** with a **hollow arrowhead**.
* **Meaning:** A class implements the methods of an interface.
* **Example:** A **Car** implements the **Vehicle** interface.

**Task3:**

package diagram representation of Command Design Pattern code

****

