

# Ocp\_Act32

## ✅ OCP Activity 32: Message Sender System

### 📌 Objective:

Understand the **Open/Closed Principle**:

“Software entities (classes, modules, functions) should be **open for extension**, but **closed for modification**.”

---

### 🧠 Scenario:

You’re creating a system that sends messages. Right now, it only sends emails. Soon, you’ll need to support more methods like SMS or push notifications — but you don’t want to keep changing the original message-sending logic every time.

---

### 🔧 Instructions:

1. Create an **abstract class or interface** called `MessageSender` :
    - Include a method `send(message)`
  2. Create a class `EmailSender` that:
    - Inherits from `MessageSender`
    - Implements the `send()` method to print:  
`"Sending email: [message]"`
  3. Create another class `SMSSender` that:
    - Inherits from `MessageSender`
    - Implements `send()` to print:  
`"Sending SMS: [message]"`
  4. Create a function `send_notification(sender, message)` that:
    - Calls `sender.send(message)`
  5. Test by calling the function with both sender types.
-

*Note:*

- *Give me your best output*
- *Add what you think is needed*