Design Pattern
Proxy

```
public class Employee {
    private String name;
    private int role;

    public Employee(String name,int role) {
        this.name = name;
        this.role = role;
    }

    public String getName() {
        return name;
    }

    public int getRole() {
        return role;
    }

    public String toString() {
        StringBuilder stringBuilder = new StringBuilder();
        stringBuilder.append("{ Employee : ");
        stringBuilder.append("name = ").append(name).append(",");
        stringBuilder.append("role = ").append(role).append(" }");
        return stringBuilder.toString();
    }
}

public interface IAccess {
    void grant();
}

public class RealAccess implements IAccess {
    private Employee employee;

    public RealAccess(Employee employee) {
        this.employee = employee;
    }

    public void grant() {
        System.out.println("access granted for employee : " + employee);
    }
}
```

```
public class ProxyAccess implements IAccess {
    private Employee employee;
    private RealAccess realAccess;

    public ProxyAccess(Employee employee) {
        this.employee = employee;
    }

    public void setEmployee(Employee employee) {
        this.employee = employee;
    }

    public void grant() {
        if (employee.getRole() > 4) {
            realAccess = new RealAccess(employee);
            realAccess.grant();
        } else {
            System.out.println(employee +
                  " - no Internet access granted - role level is below 5");
        }
    }
}

public class Application {
    public static void main(String... args) {
        Employee employee1 = new Employee("employee 01",6);
        ProxyAccess proxyAccess = new ProxyAccess(employee1);
        proxyAccess.grant();

        System.out.println();

        Employee employee2 = new Employee("employee 02",3);
        proxyAccess.setEmployee(employee2);
        proxyAccess.grant();
    }
}
```