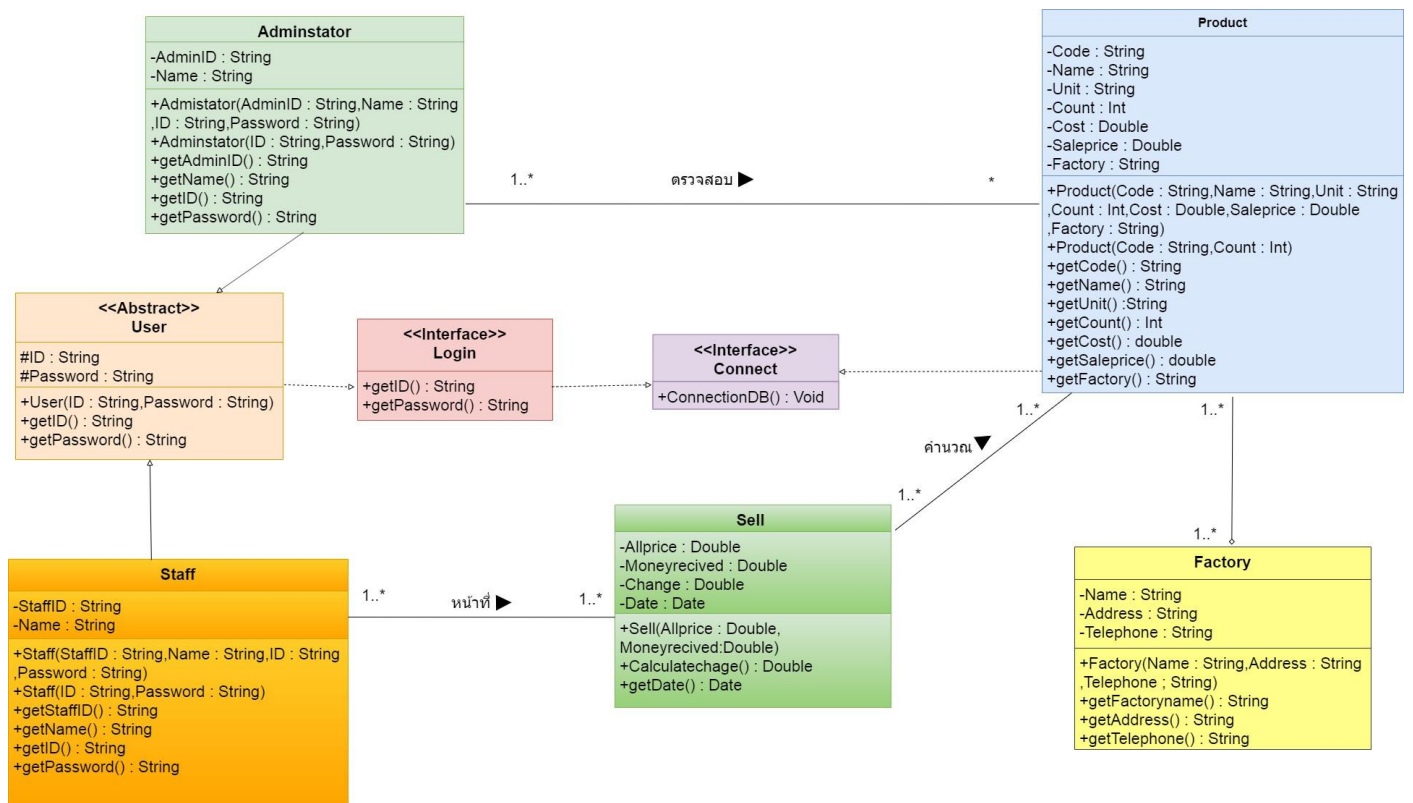
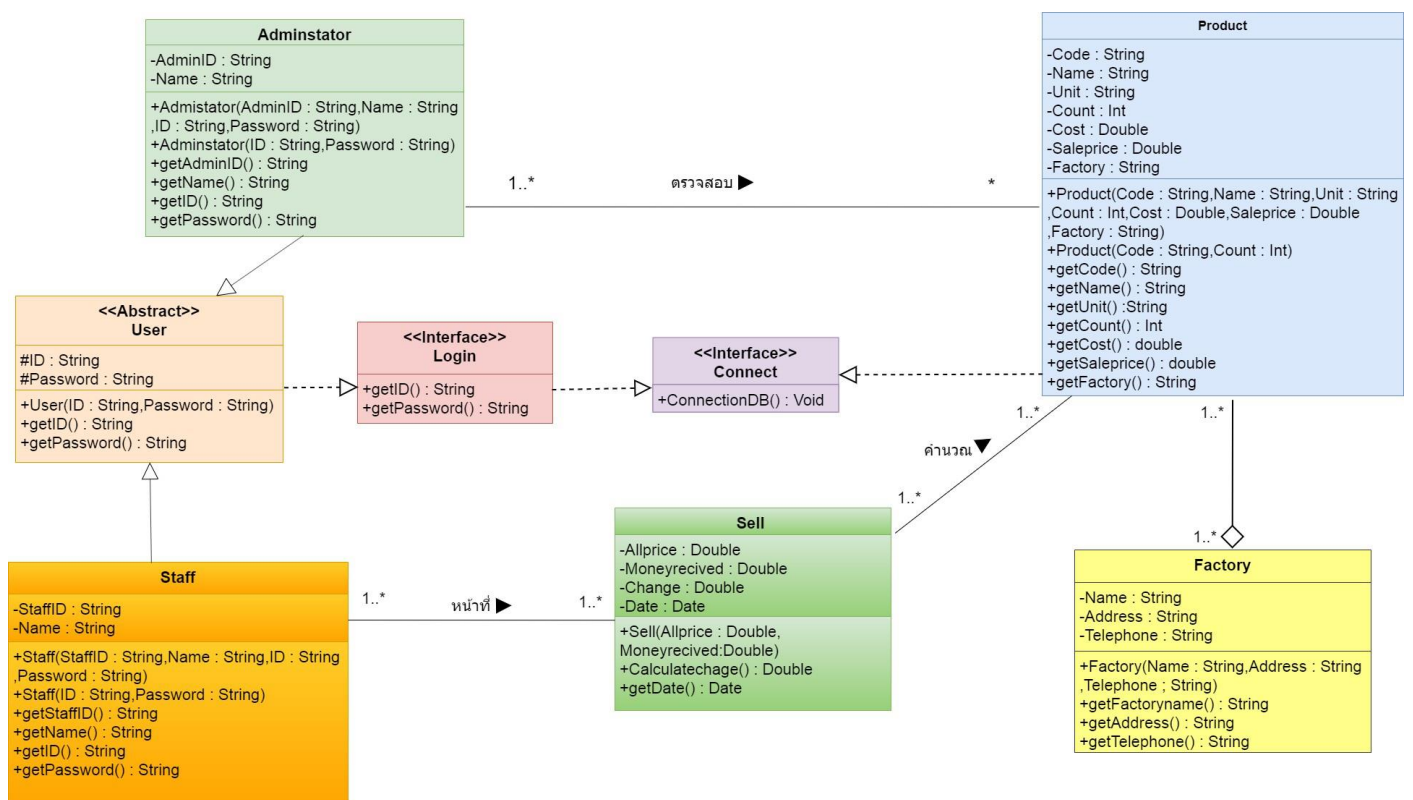


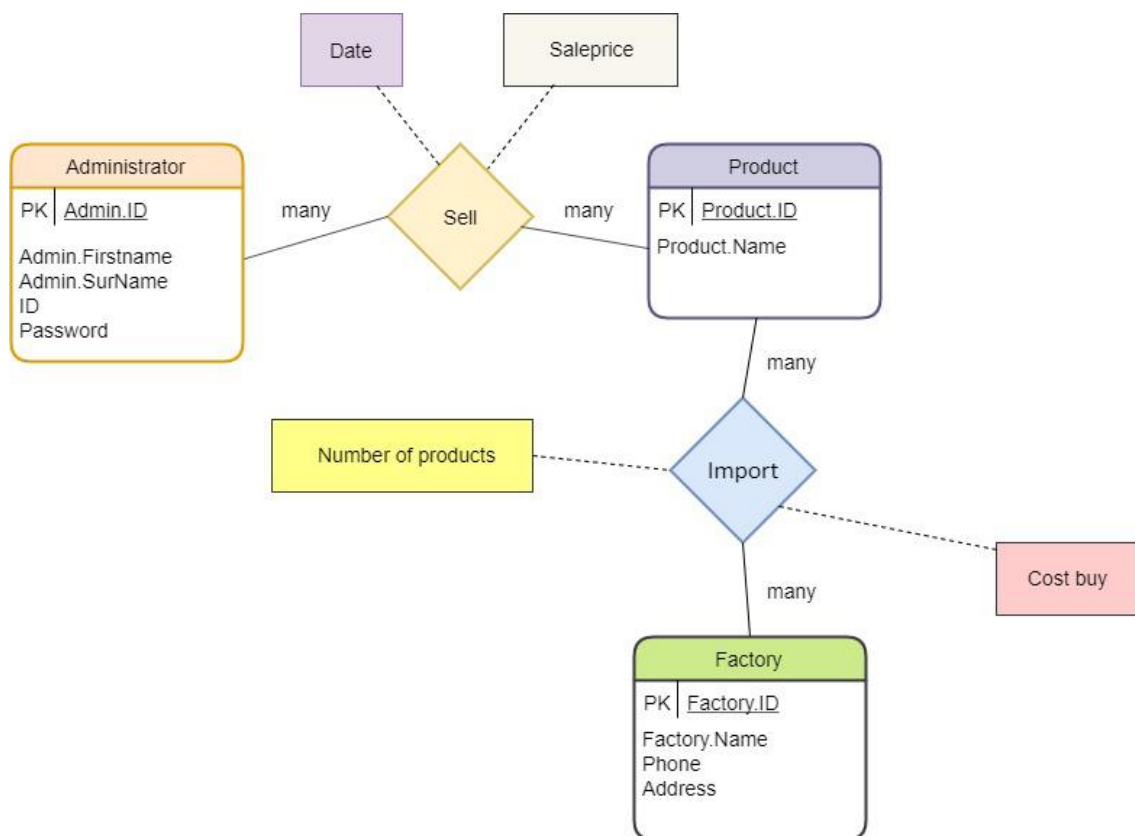
Class Diagram ก่อนแก้ไข



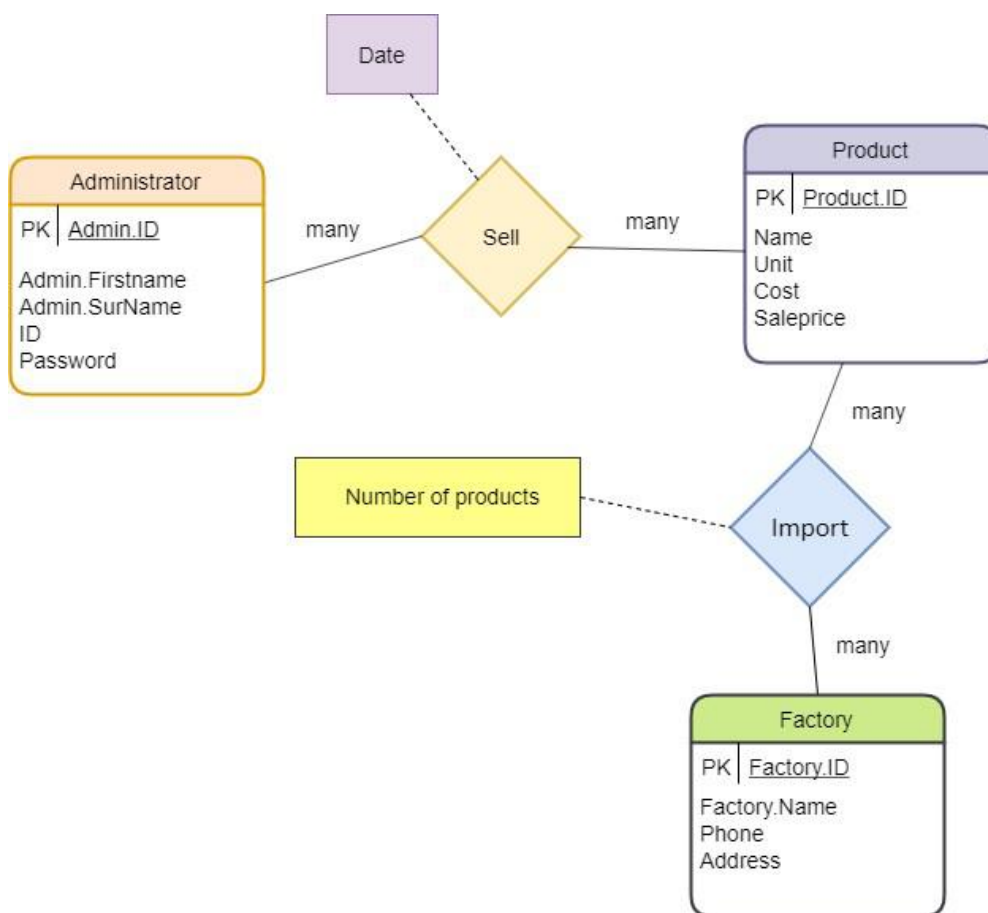
Class Diagram หลังแก้ไข



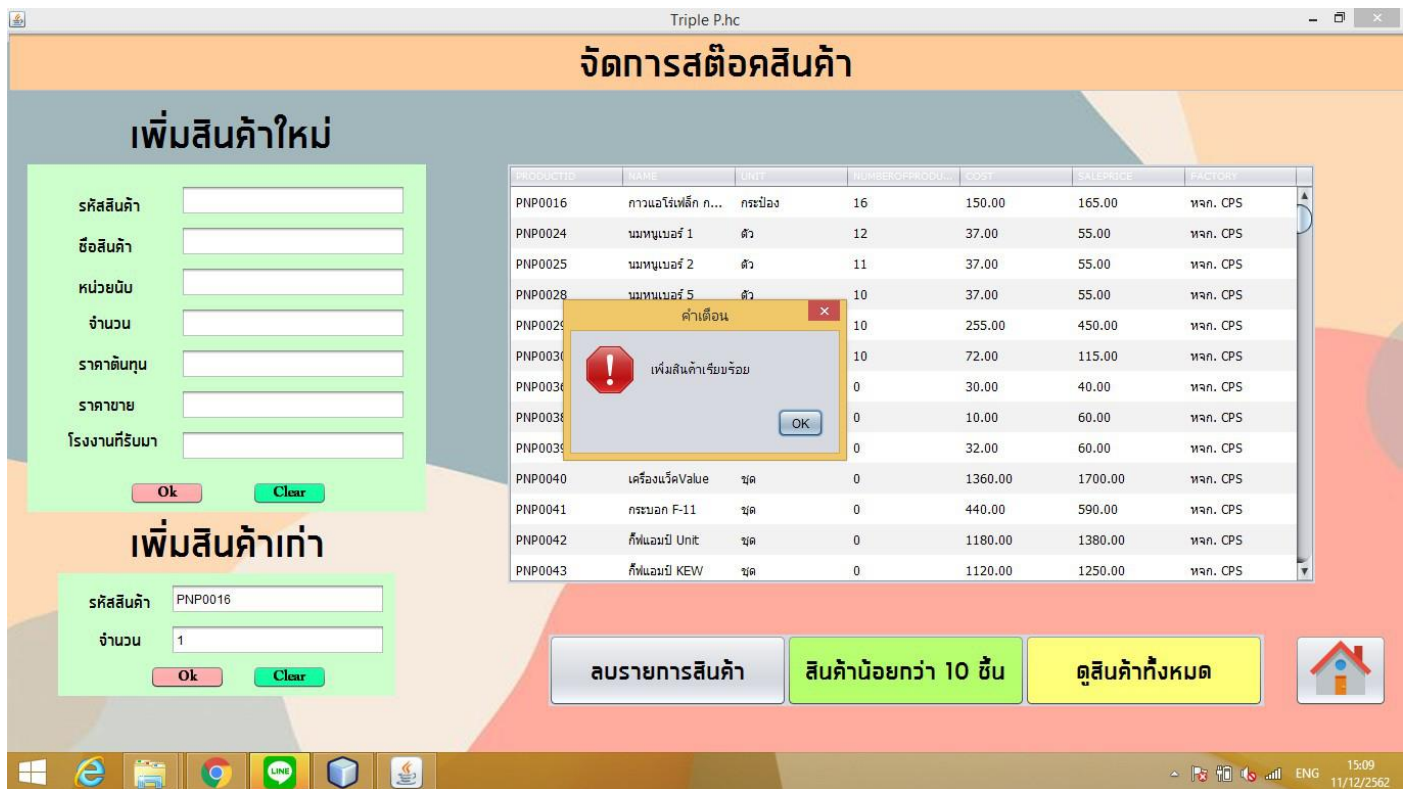
Er ก่อนแก้ไข



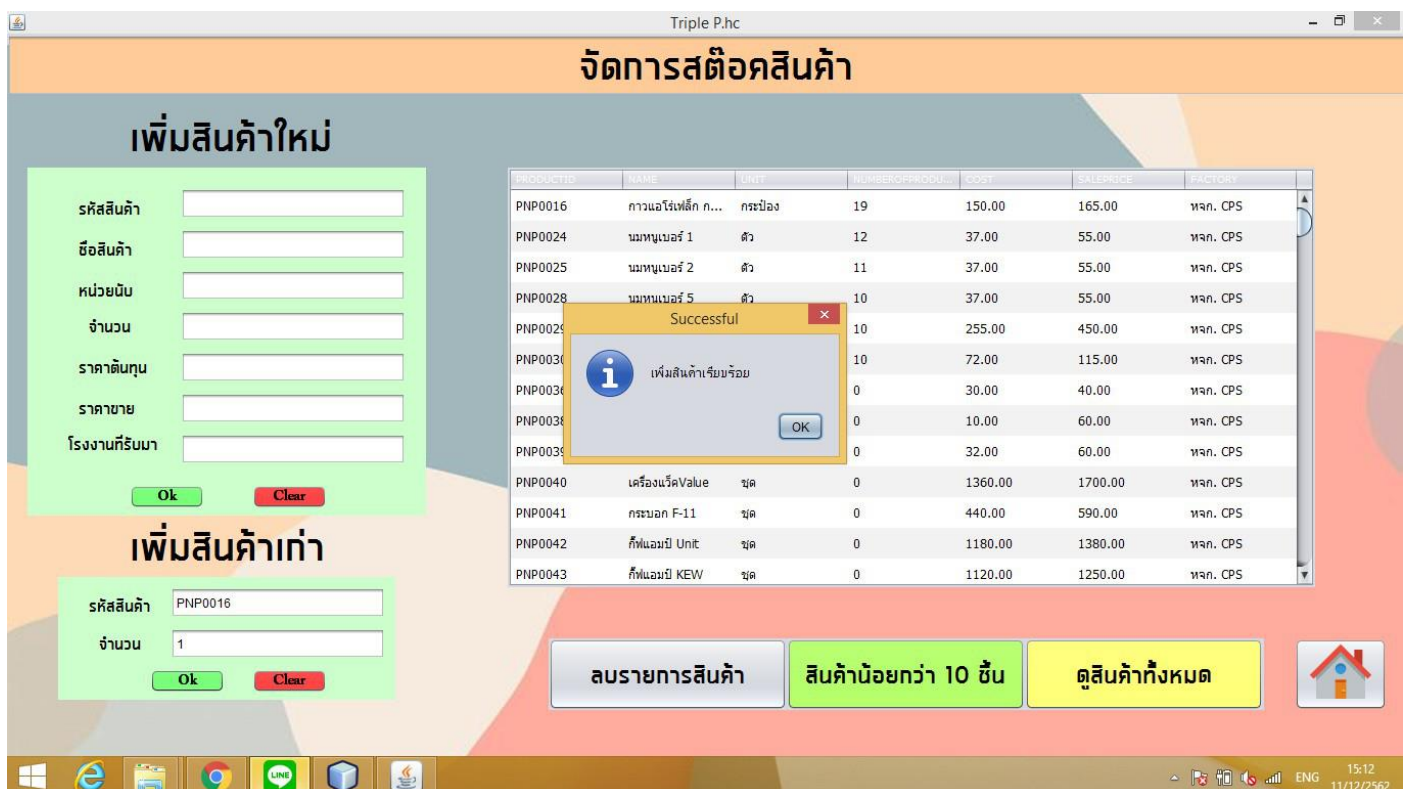
Er หลังแก้ไข



ก่อนแก้ไข message & สีปุ่ม ok clear



หลังแก้ไข message & สีปุ่ม ok clear



- ตรงสัญลักษณ์ **popup message** หน้าต่างอื่นๆที่ไม่ควรใช้เป็นสัญลักษณ์ **error** ได้แก้ไขเป็นสัญลักษณ์แบบสีฟ้าหมดแล้ว
- **encapsulation** ในโปรแกรมจะสังเกตว่าทุก **attribute** ใน **class** จะตั้งเป็น **private** และมี **method get** ที่ทำการส่งค่าตัวแปร แต่ใน **abstract user class** ที่เรากำหนด **attribute** เป็น **protected** เพราะเราต้องการให้ **subclass** สามารถใช้ **attribute** ได้
- **inheritance** ในโปรแกรมเราใช้ **abstract user class** ในการให้ **Adminstator** ในการสืบทอดคุณสมบัติ และในอนาคตเราอาจจะไปพัฒนาในส่วนของ **Staff** ให้สามารถ **login** ได้เหมือนกับ **adminstator** ตามใน **class diagram** ที่เราได้ทำการออกแบบเอาไว้
- **polymorphism** เนื่องจาก **abstract user class** ไม่สามารถสร้าง **object** ได้เราเลยต้องใช้ **polymorphism** ในการสร้าง **object** ไปยังลูกคลาสโดยประเภทของ **object** เป็นของ **abstract user class** ดังในรูปด้านล่าง

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    getpass = jPasswordField1.getPassword();
    pass = String.valueOf( getpass);
    user a = new Adminstator(id,pass);
    id = jID.getText();
    if(!a.getID().equals("") && !a.getPass().equals("") && setlogin() == true)
    {
        Admininterface form2 = new Admininterface();
        form2.setVisible(true);
        setVisible(false);
    }
    else if(setlogin() == false)
        JOptionPane.showMessageDialog(this, "คุณกรอก ID/Password ผิดพลาด", "คำเตือน", JOptionPane.ERROR_MESSAGE);
}
```