код відповідає всім основним вимогам, поставленим у завданні лабораторної роботи. Ось детальний аналіз:

**1. Створення класів та їх функціональність**  
• **Product:**  
– Має атрибути product\_id, name та price.  
– Реалізований метод get\_info, який повертає інформацію про продукт.  
– Окрім того, переозначені методи **eq** та **hash** дозволяють використовувати об'єкти цього класу як ключі в словнику, що дуже зручно для зберігання товарів у замовленні.

• **Order:**  
– Зберігає order\_id, посилання на користувача та словник з товарами (де кожен ключ — об'єкт продукту, а значення — кількість).  
– Методи add\_product і remove\_product забезпечують можливість додавання та видалення товарів із замовлення.  
– Метод calculate\_total правильно підраховує загальну суму замовлення, множачи ціну кожного продукту на кількість.

• **User:**  
– Містить основні атрибути: user\_id, name, email, список замовлень, а також додаткове поле is\_logged\_in для демонстрації процесу логування.  
– Реалізовані методи register, place\_order, view\_orders і login.  
– Метод login додає додаткову функцію, що перевіряє правильність email, що є корисним розширенням стандартних вимог.

**2. Модульні тести**  
• Використовуючи модуль unittest, ви створили серію тестів, які охоплюють: – Реєстрацію користувача (test\_user\_register).  
– Функціональність оформлення замовлення (test\_place\_order, test\_view\_orders).  
– Додавання та видалення товарів із замовлення (test\_add\_product\_to\_order, test\_add\_product\_multiple\_times, test\_remove\_product, test\_remove\_product\_completely).  
– Підрахунок загальної суми замовлення (test\_calculate\_total).  
– Перевірка методу get\_info та порівняння об'єктів Product (test\_get\_info, test\_product\_equality).  
– Функціональність логування (test\_login\_success, test\_login\_failure).