

- **Testy dla klasy `Product`**

- **Test tworzenia produktu:**

- **Cel:** Sprawdzić, czy produkt jest tworzony poprawnie.
 - **Oczekiwane wyniki:**
 - Nazwa produktu powinna być taka sama, jak ta podana podczas tworzenia.
 - Cena powinna być zgodna z podaną wartością.

- **Testy dla klasy `OrderItem`**

- **Test obliczania całkowitej ceny:**

- **Cel:** Sprawdzić, czy całkowita cena zamówienia (ilość * cena jednostkowa) jest poprawnie obliczana.
 - **Oczekiwane wyniki:**
 - Całkowita cena powinna odpowiadać iloczynowi ceny jednostkowej i ilości.

- **Testy dla klasy `Order`**

- **Test dodawania produktu:**

- **Cel:** Sprawdzić, czy produkt zostaje dodany do zamówienia.
 - **Oczekiwane wyniki:**
 - Po dodaniu produktu, liczba pozycji w zamówieniu powinna wzrosnąć.
 - Jeśli ten sam produkt jest dodawany ponownie, jego ilość powinna wzrosnąć, a liczba pozycji w zamówieniu nie powinna wzrosnąć.

- **Test usuwania produktu:**

- **Cel:** Sprawdzić, czy produkt jest usuwany z zamówienia zgodnie z oczekiwaniami.
 - **Oczekiwane wyniki:**
 - Jeśli ilość produktu wynosi 1, powinien zostać usunięty z zamówienia.
 - Jeśli ilość produktu jest większa niż 1, jego ilość powinna być zmniejszona o 1.

- **Test obliczania wartości zamówienia:**

- **Cel:** Sprawdzić, czy metoda `GetTotalOrderValue()` poprawnie oblicza wartość zamówienia, uwzględniając ewentualne zniżki.
 - **Oczekiwane wyniki:**
 - Bez zniżek: całkowita wartość powinna być równa sumie wartości wszystkich pozycji.
 - Przy dodaniu zniżek na drugi i trzeci najtańszy produkt, wartość zamówienia powinna być odpowiednio obniżona.
 - Dodatkowo, zniżka 5% powinna być zastosowana dla zamówień powyżej 5000 PLN.

- **Testy integracyjne**

- **Test całego przepływu zamówienia:**

- **Cel:** Sprawdzić, czy wszystkie operacje na zamówieniu (dodawanie, usuwanie) są poprawnie obsługiwane.
 - **Oczekiwane wyniki:**
 - Dodanie kilku produktów i następne ich usunięcie powinno prowadzić do poprawnej aktualizacji stanu zamówienia.

- **Testy błędów**

- **Test niepoprawnych danych:**

- **Cel:** Sprawdzić, jak aplikacja obsługuje nieprawidłowe dane wejściowe.
 - **Oczekiwane wyniki:**
 - Próba dodania produktu, który nie istnieje, powinna skutkować odpowiednim komunikatem o błędzie.
 - Wprowadzenie ujemnej lub zerowej ilości produktu powinno prowadzić do wyświetlenia komunikatu o błędzie.

- **Testy wydajnościowe**

- **Test obciążeniowy:**

- **Cel:** Sprawdzić, jak aplikacja radzi sobie z dużą liczbą produktów w zamówieniu.
 - **Oczekiwane wyniki:**
 - Aplikacja powinna działać płynnie, a obliczenia powinny być wykonane w rozsądnym czasie.

- **Testy interfejsu użytkownika**

- **Test interakcji użytkownika:**

- **Cel:** Sprawdzić, czy wszystkie opcje w interfejsie działają zgodnie z oczekiwaniami.
 - **Oczekiwane wyniki:**
 - Wszystkie opcje w menu powinny być dostępne i funkcjonalne.
 - Komunikaty wyświetlane przez aplikację powinny być zrozumiałe i pomocne dla użytkownika.