

A photograph of a wooden fence with several colorful U.S. Mailboxes mounted on it. The mailboxes are in two rows. The top row has seven mailboxes: grey, yellow, blue, green, orange, teal, and red. The bottom row has seven mailboxes: green, red, teal, yellow, grey, blue, and orange. The mailboxes are arranged in a slightly staggered fashion. The background is a dark, textured wall. The overall lighting is somewhat dim, giving the image a slightly moody feel.

# Poczta kwiatowa

Anna Paziewska, Mateusz Plichta, Jakub Trzcinka, Karol Żelazowski

# Wykorzystane biblioteki

JDBC  
xerial.sqlite.jdbc

Pozwala na  
komunikację z  
bazą danych.

ASCII table  
de.vandermeer.asciitable

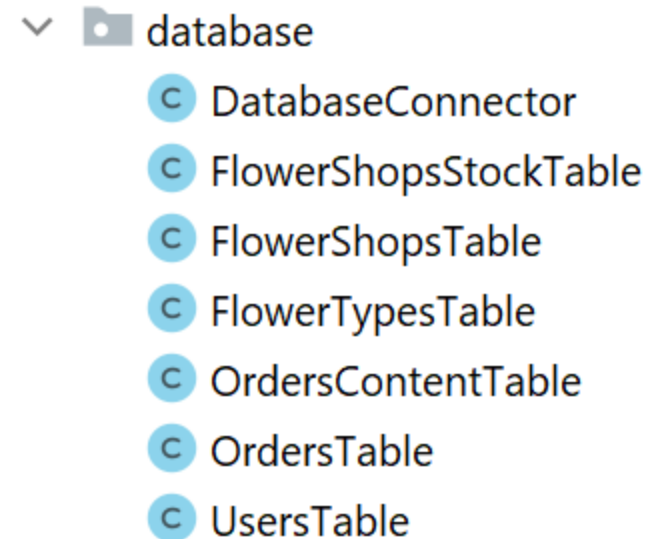
Narzędzie do  
prostego  
tworzenia tabel

JUnit Jupiter API  
junit.jupiter.api

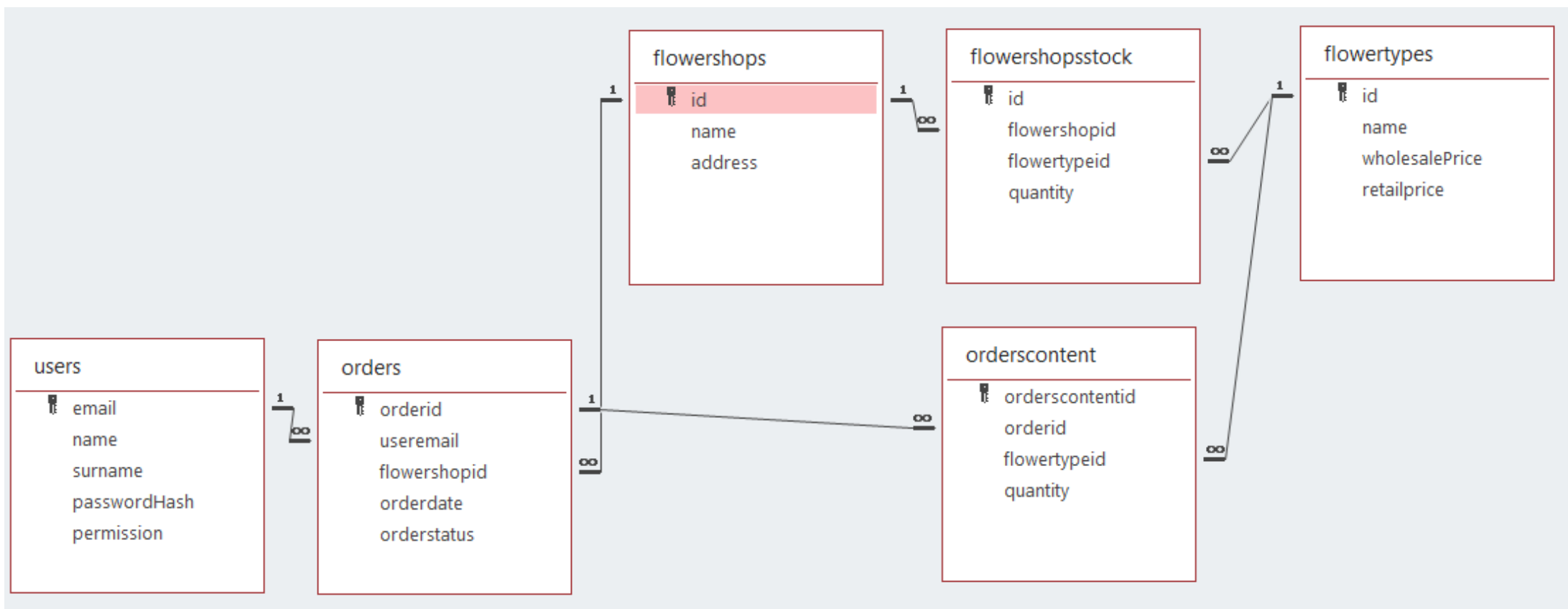
Umożliwia  
pisanie testów  
jednostkowych

# Pakiet database

- W pakiecie znajdują się klasy, które odpowiadają za komunikację z bazą danych
- Osobne klasy na poszczególne tabele




# Struktura bazy danych



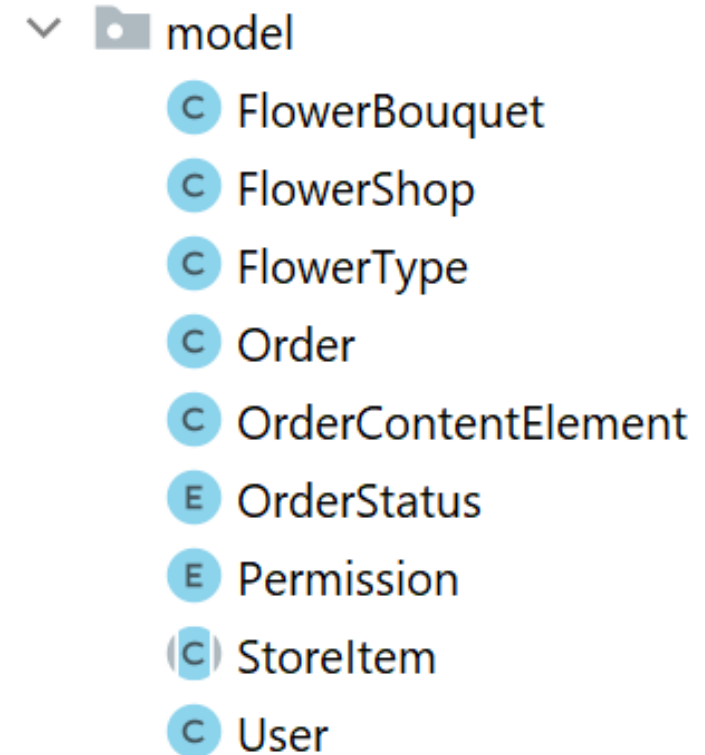
# Pakiet exception

- Używamy wyjątków aby obsługiwać błędy, które może napotkać użytkownik w trakcie korzystania z programu

- ▼  exception
  - Ⓢ FlowerShopNotFoundException
  - Ⓢ ItemNotFoundException
  - Ⓢ NotInTheOrderException
  - Ⓢ OrderNotFoundException
  - Ⓢ StoreItemNotAvailableException
  - Ⓢ TooMuchFlowerDeletedException
  - Ⓢ UnknownErrorException
  - Ⓢ UserAlreadyExistsException
  - Ⓢ UserNotFoundException

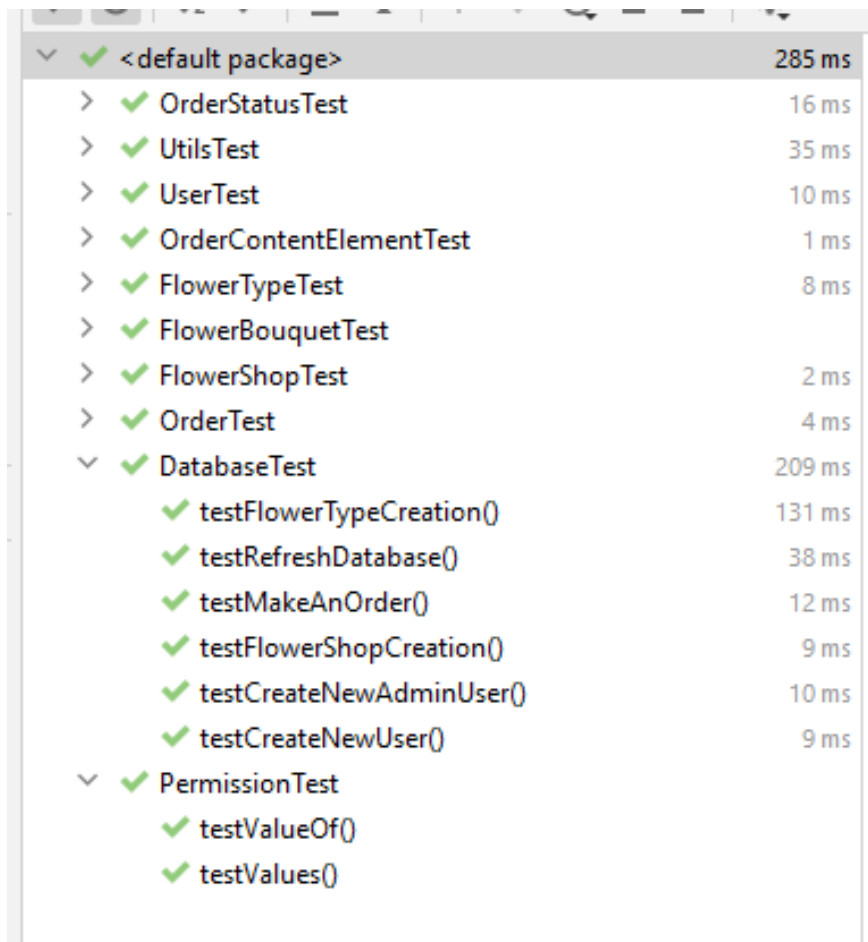
# Pakiet model

- Pakiet zawiera klasy reprezentujące obiekty takie jak Kwiat, Kwiaciarnia, Zamówienie, Użytkownik

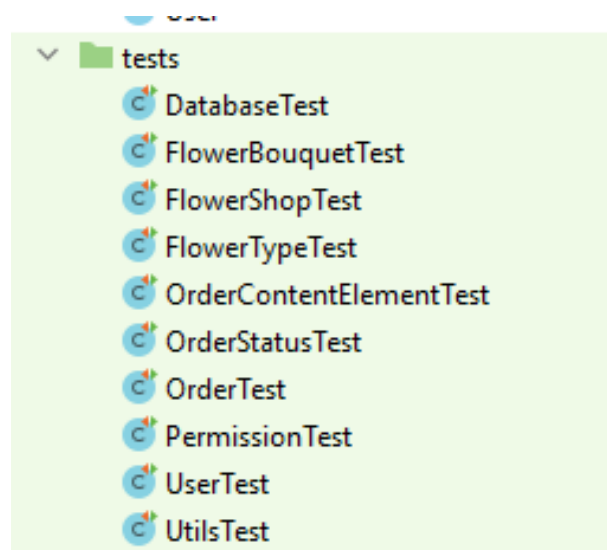


# Pakiet tests

- Pakiet zawiera testy jednostkowe
- Wszystkie testy są zdane pomyślnie

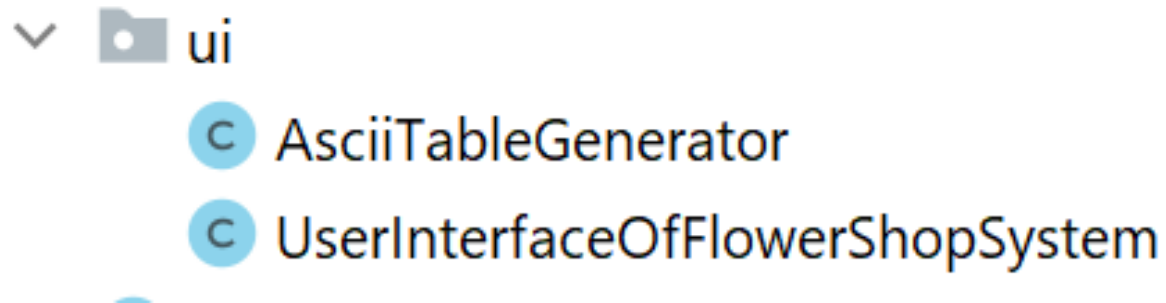


✓ <default package>	285 ms
> ✓ OrderStatusTest	16 ms
> ✓ UtilsTest	35 ms
> ✓ UserTest	10 ms
> ✓ OrderContentElementTest	1 ms
> ✓ FlowerTypeTest	8 ms
> ✓ FlowerBouquetTest	
> ✓ FlowerShopTest	2 ms
> ✓ OrderTest	4 ms
✓ DatabaseTest	209 ms
✓ testFlowerTypeCreation()	131 ms
✓ testRefreshDatabase()	38 ms
✓ testMakeAnOrder()	12 ms
✓ testFlowerShopCreation()	9 ms
✓ testCreateNewAdminUser()	10 ms
✓ testCreateNewUser()	9 ms
✓ PermissionTest	
✓ testValueOf()	
✓ testValues()	



## Pakiet UI

- Klasa AsciiTableGenerator odpowiada za konstruowanie i wyświetlanie tabel z danymi kwiaciarni, użytkowników, rodzajów kwiatów i składanych zamówień
- UserInterfaceOfFlowerShopSystem odpowiada za wszelkie interakcje pomiędzy użytkownikiem a systemem poczty kwiatowej, pozwalając intuicyjnie poruszać się po systemie





# Pozostałe klasy

- FlowerPostSystem - to główna klasa programu
- Constans - zawiera wartości stałe
- PaymentSystemSimulator – symuluje system płatności
- Utils - zawiera różne użyteczne metody, np. hashowanie haseł, czyszczenie konsoli, ukrywanie wpisywanego hasła

- Ⓒ Constants
- Ⓒ FlowerPostSystem
- Ⓒ PaymentSystemSimulator
- Ⓒ Utils

# Prezentacja działania programu

