**Exercise: Restaurant Ordering System**

Description:

You are tasked with developing a simple restaurant ordering system using Spring dependency injection.

Requirements:

1. Create a **MenuItem** class with attributes such as **name**, **description**, and **price**.
2. Create a **SimpleMenu** class. This class should manage a list of menu items.
   1. It has a list of menu items and methods.
      1. addItemToMenu
      2. RemoveItemFromMenu
3. Use constructor injection to inject the **SimpleMenu** dependency into a **Restaurant** class
4. The **Restaurant** class has fields restaurantName and SimpleMenu.
5. Implement a **Main** class to demonstrate the functionality of the restaurant ordering system. Create menu items, add them to the restaurant's menu, and display the menu.

Here you will see the implementation without spring.

Your task is to change the implementation to use spring – (no new should be called) all classes created by container with application-context.xml

// MenuItem.java

public class MenuItem {

private String name;

private String description;

private double price;

// Constructor, getters, setters

}

public **SimpleMenu** {

void addItem(MenuItem item);

void removeItem(String name);

List<MenuItem> menuItems ();

public void addItem(MenuItem item) {

menuItems.add(item);

}

public void removeItem(String name) {

menuItems.removeIf(item -> item.getName().equals(name));

}

}

// Restaurant.java

public class Restaurant {

private **SimpleMenu** simpleMenu;

public Restaurant(**SimpleMenu** simpleMenu) {

this.simpleMenu = simpleMenu;

}

public void displayMenu() {

List<MenuItem> menu = simpleMenu.getMenuItems();

for (MenuItem item : menu) {

System.out.println(item.getName() + " - " + item.getDescription() + " - $" + item.getPrice());

}

}

}

// Main.java

public class Main {

public static void main(String[] args) {

SimpleMenu simpleMenu = new SimpleMenu ();

Restaurant restaurant = new Restaurant(menuService);

MenuItem burger = new MenuItem("1", "Burger", "Juicy beef patty on a sesame seed bun", 8.99);

MenuItem pizza = new MenuItem("2", "Pizza", "Thin crust pizza with assorted toppings", 10.99);

restaurant.getSimpleMenu.addItem(burger);

restaurant.getSimpleMenu.addItem(pizza);

restaurant.displayMenu();

}

}

The target is to not create any new class in the application.

You should create Restaurant and simple Menu and Menu Items in application context

You should inject Simplemenu into restaurant in the app context

Etgar -> inject menu items into SimpleMenu with app context