***Лабараторна робота №3***

***Варіант №14***

**Завдання №1**

14. Застосувати патерн Абстрактна фабрика для сімейств класів Основні страви та Десерти, створивши фабрики їжі для української кухні та східної кухні.

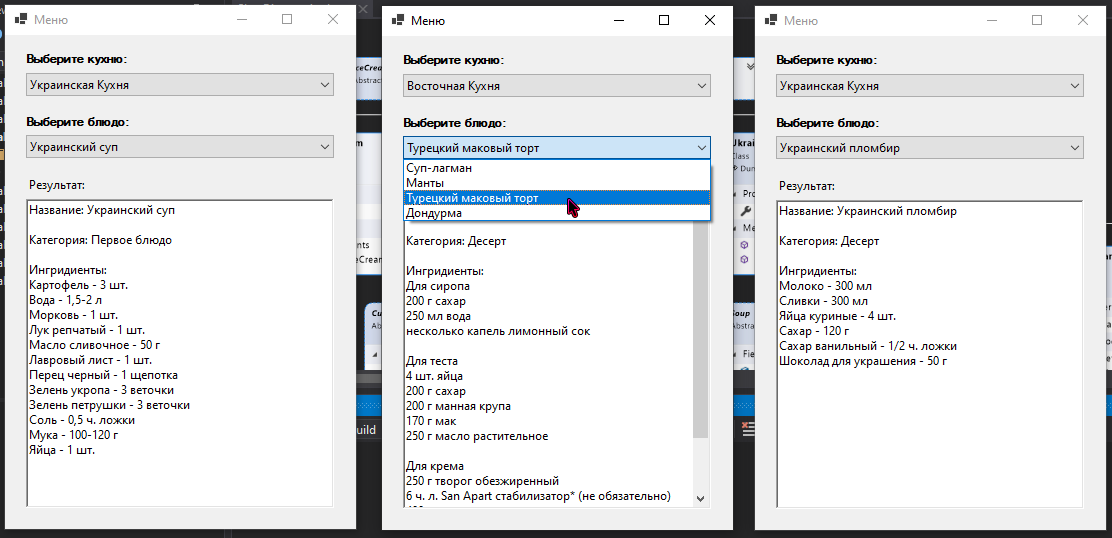
***Результати виконання завдання 1:***

*Діаграма класів додатку (рис.1):*



**Рис. 1 Діаграма класів додатку**

*Результати роботи додатку (рис.2)*



**Рис. 2 Результат виконаня додатку**

*Лістинг програмного коду:*

*\*\* MainForm.cs \*\**

using System;

using System.Windows.Forms;

namespace Lab\_3\_1

{

public partial class MainForm : Form

{

СuisineFactory \_cuisineFactory;

public MainForm()

{

InitializeComponent();

chooseCuisineBox.Items.Add("Украинская Кухня");

chooseCuisineBox.Items.Add("Восточная Кухня");

chooseEatBox.Items.Clear();

chooseEatBox.Enabled = false;

}

private void chooseCuisineBox\_SelectedIndexChanged(object sender, EventArgs e)

{

\_cuisineFactory = chooseCuisineBox.SelectedIndex switch

{

0 => new UkrainianСuisine(),

1 => new EasternСuisine(),

\_ => \_cuisineFactory

};

FillChooseEatBox(\_cuisineFactory);

}

private void chooseEatBox\_SelectedIndexChanged(object sender, EventArgs e)

{

string text = "";

switch (chooseEatBox.SelectedIndex)

{

case 0:

var soup = \_cuisineFactory.CreateSoup();

text = $"Название: {soup.Name}\n\n";

text += $"Категория: {soup.category}\n\n";

text += $"Ингридиенты:\n{soup.GetIngredients()}";

break;

case 1:

var dumplings = \_cuisineFactory.CreateDumplings();

text = $"Название: {dumplings.Name}\n\n";

text += $"Категория: {dumplings.category}\n\n";

text += $"Ингридиенты:\n{dumplings.GetIngredients()}";

break;

case 2:

var cake = \_cuisineFactory.CreateCake();

text = $"Название: {cake.Name}\n\n";

text += $"Категория: {cake.category}\n\n";

text += $"Ингридиенты:\n{cake.GetIngredients()}";

break;

case 3:

var icecream = \_cuisineFactory.CreateIceCream();

text = $"Название: {icecream.Name}\n\n";

text += $"Категория: {icecream.category}\n\n";

text += $"Ингридиенты:\n{icecream.GetIngredients()}";

break;

}

resultTextBox.Text = text;

}

private void FillChooseEatBox(СuisineFactory cuisineFactory)

{

chooseEatBox.Items.Clear();

chooseEatBox.Items.Add(cuisineFactory.CreateSoup().Name);

chooseEatBox.Items.Add(cuisineFactory.CreateDumplings().Name);

chooseEatBox.Items.Add(cuisineFactory.CreateCake().Name);

chooseEatBox.Items.Add(cuisineFactory.CreateIceCream().Name);

chooseEatBox.Enabled = true;

chooseEatBox.SelectedIndex = 0;

}

}

}

*\*\* CuisineFactory.cs \*\**

namespace Lab\_3\_1

{

abstract class СuisineFactory

{

public abstract Soup CreateSoup();

public abstract Dumplings CreateDumplings();

public abstract Cake CreateCake();

public abstract IceCream CreateIceCream();

}

class UkrainianСuisine : СuisineFactory

{

public override Soup CreateSoup()

{

return new UkrainianSoup();

}

public override Dumplings CreateDumplings()

{

return new UkrainianDumplings();

}

public override Cake CreateCake()

{

return new UkrainianCake();

}

public override IceCream CreateIceCream()

{

return new UkrainianIceCream();

}

}

class EasternСuisine : СuisineFactory

{

public override Soup CreateSoup()

{

return new EasternSoup();

}

public override Dumplings CreateDumplings()

{

return new EasternDumplings();

}

public override Cake CreateCake()

{

return new EasternCake();

}

public override IceCream CreateIceCream()

{

return new EasternIceCream();

}

}

abstract class Soup

{

public string category = "Первое блюдо";

public abstract string Name { get; }

public abstract string GetIngredients();

}

abstract class Dumplings

{

public string category = "Второе блюдо";

public abstract string Name { get; }

public abstract string GetIngredients();

}

abstract class Cake

{

public string category = "Десерт";

public abstract string Name { get; }

public abstract string GetIngredients();

}

abstract class IceCream

{

public string category = "Десерт";

public abstract string Name { get; }

public abstract string GetIngredients();

}

class UkrainianSoup : Soup

{

public override string Name { get; }

public UkrainianSoup()

{

Name = "Украинский суп";

}

public override string GetIngredients()

{

return "Картофель - 3 шт.\n" +

"Вода - 1,5-2 л\n" +

"Морковь - 1 шт.\n" +

"Лук репчатый - 1 шт.\n" +

"Масло сливочное - 50 г\n" +

"Лавровый лист - 1 шт.\n" +

"Перец черный - 1 щепотка\n" +

"Зелень укропа - 3 веточки\n" +

"Зелень петрушки - 3 веточки\n" +

"Соль - 0,5 ч. ложки\n" +

"Мука - 100-120 г\n" +

"Яйца - 1 шт.";

}

}

class UkrainianDumplings : Dumplings

{

public override string Name { get; }

public UkrainianDumplings()

{

Name = "Украинские пельмени";

}

public override string GetIngredients()

{

return "Для начинки:\n" +

"Фарш мясной (курица + свинина) – 700 г\n" +

"Лук репчатый – 1 шт.\n" +

"Чеснок – 2 зубчика\n" +

"Зелень укропа – 1 небольшой пучок\n" +

"Перец чёрный молотый – по вкусу\n" +

"Соль – по вкусу\n\n" +

"Для теста (стакан - 250 мл):\n" +

"Мука – 3 стакана\n" +

"Яйцо – 1 шт.\n" +

"Масло растительное – 3 ст. ложки\n" +

"Соли – 0,5 ч. ложки\n" +

"Кипяток – 1 стакан";

}

}

class UkrainianCake : Cake

{

public override string Name { get; }

public UkrainianCake()

{

Name = "Киевский торт";

}

public override string GetIngredients()

{

return "Для коржей:\n" +

"Белки яичные – 12 шт.\n" +

"Мука – 100 г\n" +

"Сахар – 500 г\n" +

"Орехи – 350 г\n" +

"Ванильный сахар – 20 г\n\n" +

"Для крема:\n" +

"Масло сливочное – 500 г\n" +

"Желтки – 12 шт.\n" +

"Молоко – 350 г\n" +

"Сахар – 300 г\n" +

"Какао – 20 г\n" +

"Ванильный сахар – 10 г\n" +

"Мука – 4 ст. л.\n" +

"Коньяк – 2 ст. ложки\n" +

"Для украшения:\n" +

"Шоколад - 1 плитка";

}

}

class UkrainianIceCream : IceCream

{

public override string Name { get; }

public UkrainianIceCream()

{

Name = "Украинский пломбир";

}

public override string GetIngredients()

{

return "Молоко - 300 мл\n" +

"Сливки - 300 мл\n" +

"Яйца куриные - 4 шт.\n" +

"Сахар - 120 г\n" +

"Сахар ванильный - 1/2 ч. ложки\n" +

"Шоколад для украшения - 50 г\n";

}

}

class EasternSoup : Soup

{

public override string Name { get; }

public EasternSoup()

{

Name = "Суп-лагман";

}

public override string GetIngredients()

{

return "Мякоть говяжья - 400 г\n" +

"Лук - 1 шт.\n" +

"Редька - 1 шт.\n" +

"Помидоры - 2 шт.\n" +

"или томат-паста - 1-2 ст. ложки\n" +

"Вода - 2,5 стакана\n" +

"Чеснок - 6 зубчиков\n" +

"Масло растительное - 30 г\n" +

"Соль - по вкусу\n" +

"Перец красный молотый - по вкусу\n" +

"Зелень - по вкусу\n\n" +

"Для лапши:\n" +

"Мука - 2 стакана\n" +

"Вода - 0,7-1 стакан\n" +

"Соль - 1 щепотка\n" +

"Масло растительное - 1-2 ч. ложки\n" +

"или готовая лапша - 400 г\n";

}

}

class EasternDumplings : Dumplings

{

public override string Name { get; }

public EasternDumplings()

{

Name = "Манты";

}

public override string GetIngredients()

{

return "Телятина - 1 кг\n" +

"Лук репчатый - 3 шт.\n" +

"Чеснок - 3 зубчика\n" +

"Мука - 0,5 кг\n" +

"Яйцо – 1 шт.\n" +

"Вода - сколько возьмет тесто\n" +

"Соль - по вкусу\n" +

"Смесь свежемолотых перцев – 1 ч.л. (по вкусу)\n" +

"Черный свежемолотый перец – 1 ч.л. (по вкусу)\n";

}

}

class EasternCake : Cake

{

public override string Name { get; }

public EasternCake()

{

Name = "Турецкий маковый торт";

}

public override string GetIngredients()

{

return "Для сиропа\n" +

"200 г сахар\n" +

"250 мл вода\n" +

"несколько капель лимонный сок\n\n" +

"Для теста\n" +

"4 шт. яйца\n" +

"200 г сахар\n" +

"200 г манная крупа\n" +

"170 г мак\n" +

"250 г масло растительное\n\n" +

"Для крема\n" +

"250 г творог обезжиренный\n" +

"6 ч. л. San Apart стабилизатор\* (не обязательно)\n" +

"400 г сливки охлажденные\n\n" +

"Для украшения\n" +

"кокосовая стружка\n";

}

}

class EasternIceCream : IceCream

{

public override string Name { get; }

public EasternIceCream()

{

Name = "Дондурма";

}

public override string GetIngredients()

{

return "Молоко (лучше всего козье, но и коровье подойдет) – 500 мл.\n" +

"Сахар — 170 гр.\n" +

"Салеп – 1,5 чайной ложки.\n" +

"Ванильный сахар — 1 пакетик .";

}

}

}

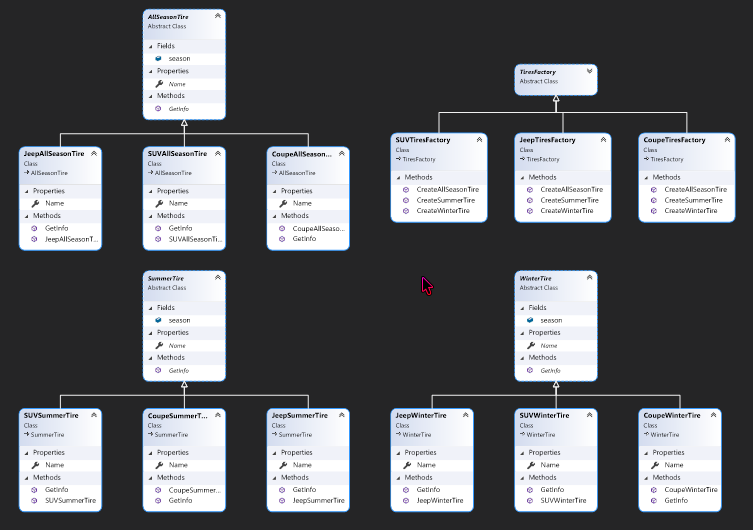
**Завдання №2**

Застосувати патерн Абстрактна фабрика для предметної області за варіантами.

(14. Шиномонтаж)

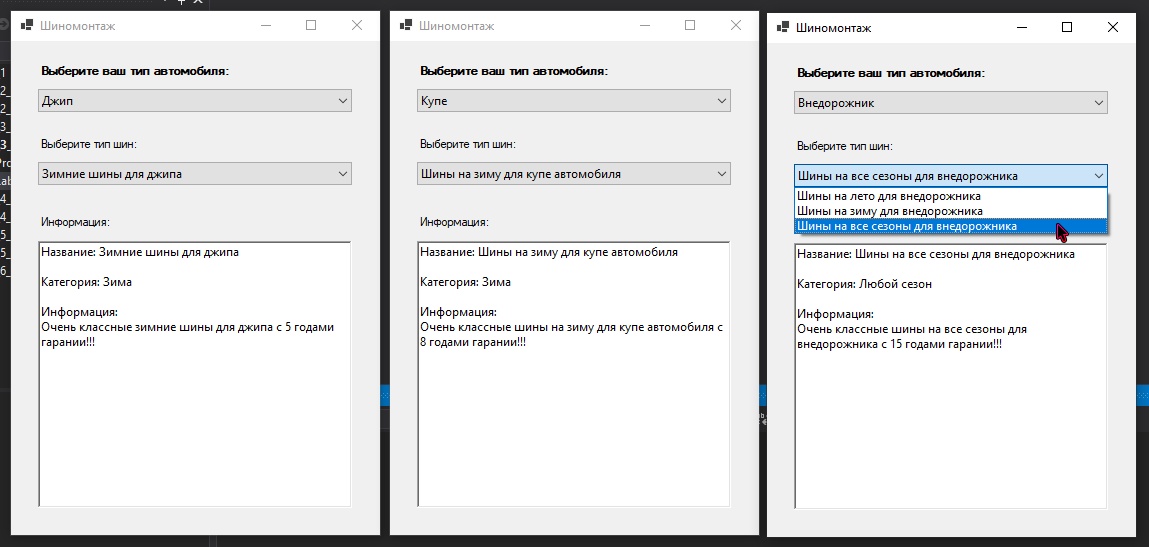
***Результати виконання завдання 2:***

*Діаграма класів додатку (рис.3):*



**Рис. 3 Діаграма класів додатку**

*Результати роботи додатку (рис.4)*



**Рис. 4 Результат виконаня додатку**

*Лістинг програмного коду:*

*\*\* MainForm.cs \*\**

using System;

using System.Windows.Forms;

namespace Lab\_3\_2

{

public partial class MainForm : Form

{

TiresFactory \_tiresFactory;

public MainForm()

{

InitializeComponent();

carTypeBox.Items.Add("Джип");

carTypeBox.Items.Add("Купе");

carTypeBox.Items.Add("Внедорожник");

carTypeBox.SelectedIndex = 0;

}

private void carTypeBox\_SelectedIndexChanged(object sender, EventArgs e)

{

\_tiresFactory = carTypeBox.SelectedIndex switch

{

0 => new JeepTiresFactory(),

1 => new CoupeTiresFactory(),

2 => new SUVTiresFactory(),

\_ => new JeepTiresFactory()

};

FillTiresTypeBox(\_tiresFactory);

}

private void tiresTypeBox\_SelectedIndexChanged(object sender, EventArgs e)

{

string text = "";

switch (tiresTypeBox.SelectedIndex)

{

case 0:

var summerTires = \_tiresFactory.CreateSummerTire();

text = $"Название: {summerTires.Name}\n\n";

text += $"Категория: {summerTires.season}\n\n";

text += $"Информация:\n{summerTires.GetInfo()}";

break;

case 1:

var winterTires = \_tiresFactory.CreateWinterTire();

text = $"Название: {winterTires.Name}\n\n";

text += $"Категория: {winterTires.season}\n\n";

text += $"Информация:\n{winterTires.GetInfo()}";

break;

case 2:

var allSeasonTires = \_tiresFactory.CreateAllSeasonTire();

text = $"Название: {allSeasonTires.Name}\n\n";

text += $"Категория: {allSeasonTires.season}\n\n";

text += $"Информация:\n{allSeasonTires.GetInfo()}";

break;

}

resultTextBox.Text = text;

}

private void FillTiresTypeBox(TiresFactory tiresFactory)

{

tiresTypeBox.Items.Clear();

tiresTypeBox.Items.Add(tiresFactory.CreateSummerTire().Name);

tiresTypeBox.Items.Add(tiresFactory.CreateWinterTire().Name);

tiresTypeBox.Items.Add(tiresFactory.CreateAllSeasonTire().Name);

tiresTypeBox.Enabled = true;

tiresTypeBox.SelectedIndex = 0;

}

}

}

*\*\* TiresFactory.cs \*\**

namespace Lab\_3\_2

{

abstract class TiresFactory

{

public abstract SummerTire CreateSummerTire();

public abstract WinterTire CreateWinterTire();

public abstract AllSeasonTire CreateAllSeasonTire();

}

class JeepTiresFactory : TiresFactory {

public override SummerTire CreateSummerTire()

{

return new JeepSummerTire();

}

public override WinterTire CreateWinterTire()

{

return new JeepWinterTire();

}

public override AllSeasonTire CreateAllSeasonTire()

{

return new JeepAllSeasonTire();

}

}

class CoupeTiresFactory : TiresFactory {

public override SummerTire CreateSummerTire()

{

return new CoupeSummerTire();

}

public override WinterTire CreateWinterTire()

{

return new CoupeWinterTire();

}

public override AllSeasonTire CreateAllSeasonTire()

{

return new CoupeAllSeasonTire();

}

}

class SUVTiresFactory : TiresFactory {

public override SummerTire CreateSummerTire()

{

return new SUVSummerTire();

}

public override WinterTire CreateWinterTire()

{

return new SUVWinterTire();

}

public override AllSeasonTire CreateAllSeasonTire()

{

return new SUVAllSeasonTire();

}

}

abstract class SummerTire

{

public string season = "Лето";

public abstract string Name { get; }

public abstract string GetInfo();

}

abstract class WinterTire

{

public string season = "Зима";

public abstract string Name { get; }

public abstract string GetInfo();

}

abstract class AllSeasonTire

{

public string season = "Любой сезон";

public abstract string Name { get; }

public abstract string GetInfo();

}

class JeepSummerTire : SummerTire

{

public override string Name { get; }

public JeepSummerTire()

{

Name = "Летние шины для джипа";

}

public override string GetInfo()

{

return "Очень классные летние шины для джипа с 10 годами гарании!!!";

}

}

class JeepWinterTire : WinterTire

{

public override string Name { get; }

public JeepWinterTire()

{

Name = "Зимние шины для джипа";

}

public override string GetInfo()

{

return "Очень классные зимние шины для джипа с 5 годами гарании!!!";

}

}

class JeepAllSeasonTire : AllSeasonTire

{

public override string Name { get; }

public JeepAllSeasonTire()

{

Name = "Шины на все сезоны для джипа";

}

public override string GetInfo()

{

return "Очень классные шины на все сезоны для джипа с 2 годами гарании!!!";

}

}

class CoupeSummerTire : SummerTire {

public override string Name { get; }

public CoupeSummerTire()

{

Name = "Шины на лето для купе автомобиля";

}

public override string GetInfo()

{

return "Очень классные шины на лето для купе автомобиля с 3 годами гарании!!!";

}

}

class CoupeWinterTire : WinterTire {

public override string Name { get; }

public CoupeWinterTire()

{

Name = "Шины на зиму для купе автомобиля";

}

public override string GetInfo()

{

return "Очень классные шины на зиму для купе автомобиля с 8 годами гарании!!!";

}

}

class CoupeAllSeasonTire : AllSeasonTire {

public override string Name { get; }

public CoupeAllSeasonTire()

{

Name = "Шины на все сезоны для купе автомобиля";

}

public override string GetInfo()

{

return "Очень классные шины на все сезоны для купе автомобиля с 4 годами гарании!!!";

}

}

class SUVSummerTire : SummerTire

{

public override string Name { get; }

public SUVSummerTire()

{

Name = "Шины на лето для внедорожника";

}

public override string GetInfo()

{

return "Очень классные шины на лето для внедорожника с 6 годами гарании!!!";

}

}

class SUVWinterTire : WinterTire

{

public override string Name { get; }

public SUVWinterTire()

{

Name = "Шины на зиму для внедорожника";

}

public override string GetInfo()

{

return "Очень классные шины на зиму для внедорожника с 4 годами гарании!!!";

}

}

class SUVAllSeasonTire : AllSeasonTire

{

public override string Name { get; }

public SUVAllSeasonTire()

{

Name = "Шины на все сезоны для внедорожника";

}

public override string GetInfo()

{

return "Очень классные шины на все сезоны для внедорожника с 15 годами гарании!!!";

}

}

}