Платформи корпоративних ІС

Завдання 1. Консольний проект. Повторення

Розробити консольну програму для реалізації вказаного завдання

Варіант 1.

1) Визначити інтерфейс IShape з методами обчислення площі фігури, периметра фігури, інтерфейс IFileManager з методами введення-виведення з файлу. Класи Circle (координати центру та радіус) , Square (координати верхнього лівого кутка та нижнього правого кутка), та Triangle реалізують ці інтерфейс. В текстовому файлі задані дані про геометричні фігури.

2) Зчитати дані у колекцію List. Посортувати колекцію по зростанню площ і вивести результат у файл1.

Знайти фігури, які лежать у третій чверті координатної площини і записати їх в окрему колекцію, яку необхідно посортувати за спаданням периметрів і вивести у файл2.

3) Перехоплення та опрацювання відповідних винятків

4) Використання Linq

5) Весь код повинен бути покритий юніт тестами

Варіант 2.

1) Визначити абстрактний клас Contact з полем name, властивістю доступу до цього поля, інтерфейс IFileManager з методами введення-виведення з файлу. Визначити класи PhoneContact (містить номер телефону), MailContact (містить мейл) та SkypeContact (ім’я в скайпі), які наслідують цей клас та реалізують інтерфейс. В текстовому файлі задані дані про різні контакти.

2) Зчитати дані у колекцію ArrayList. Посортувати колекцію в алфавітному порядку і вивести результат у файл1.

3) Переписати дані у новий контейнер пар: name - номер телефону+мейл+скайп. Вивести результат у файл2. Вивести окремо імена тих людей, що мають лише телефон

4) Перехоплення та опрацювання відповідних винятків

4) Використання Linq

5) Весь код повинен бути покритий юніт тестами

Варіант 3.

1) Визначити інтерфейс IColor, який містить властивість доступу до кольору. Визначити структуру ColoredSide (містить поля Колір та Довжина), яка реалізує цей інтерфейс і представляє кольрову сторону. Визначити клас КольоровийТрикутик, який містить 3 кольорові сторони, методи введеня-виведення. В текстовому файлі задані дані про кольорові трикутники.

2) Зчитати дані в колекцію SortedList кольорових трикутників. Фігури повинні бути посортовані за периметрами. Вивести результат у файл1

3) Визначити ті трикутники, в яких всі сторони одного кольору і переписати інформацію про них у колекцію пар: колір-кількість трикутників такого кольору.

4) Перефарбувати третю сторону трикутників, в яких дві сторони одного кольору, в цей колір, щоб вони стали одноколірні

5) Перехоплення та опрацювання відповідних винятків

6) Використання Linq

7) Весь код повинен бути покритий юніт тестами

Варіант 4.

1) Визначити інтерфейс IReadable з методом Read() та клас для представлення валюти Currency, який містить поля CurrencyName та Amount. Цей клас повинен наслідувати інтерфейс IReadable і реалізувати метод Read() для зчитування з файлу певних сум. У текстовому файлі задані дані у вигляді: 123 грн, 34 $, 23 Є, 459 грн і т.д. Зчитати дані з файлу у колекцію List з допомогою методу Read().

2) З колекції List вибрати лише гривневі суми і видрукувати результат на консоль, використавши метод ToString(). Для цього в класі Currency перевизначити метод ToString для форматного виведення.

3) Сформувати контейнер пар: назва валюти-загальна сума і вивести результат у файл

4) Здійснити конвертацію всіх сум у вказану з консолі валюту, результат записати в контейнер і вивести у файл.

5) Перехоплення винятків. При зчитуванні даних передбачити виняткові ситуації через неправильні вхідні дані та їх перехоплення у відповідних місцях

6) Використання Linq

7) Весь код повинен бути покритий юніт тестами

Варіант 5.

1) Визначити клас HTTPError для представлення HTTP помилок з полями Code (код помилки), Description (пояснення) (Наприклад: 400 - Bad Request, 401- Unauthorized, 403 – Forbidden і ін.) та дата. Цей клас повинен реалізувати інтерфейс IComparable та перевизначити метод Equal(), реалізацію яких узгодити. У текстовому файлі1 задані  дані про такі HTTP помилки, які сталися протягом певного періоду часу. Зчитати ці дані у колекцію.

2) текстовому файлі2 задано текст, який містить коди помилок і інший текст. Зчитати текст і, використовуючи отриману в попередньому завданні колекцію, замінити в цьому тексті всі коди помилок на їх пояснення і дату. Результат вивести на консоль.

3) Створити новий масив пар: код помилки – список дат цих помилок. Вивести у файл.

4) Перехоплення винятків. При роботі з текстовими файлами передбачити виняткові ситуації через відсутність файлів, або неможливість їх зчитування і передбачити їх перехоплення у відповідних місцях. І інші винятки

5) Використання Linq

6) Весь код повинен бути покритий юніт тестами

Варіант 6.

1) Вимоги до продукту можуть бути описані у вигляді набору UserStory (наприклад: Pr1Us1 Text of UserStory, де Pr1 – назва продукту, Us1 – UserStory1 ), набору Прототипів (малюнків, діаграм і ін. графічних матеріалів із зазначенням назви файлу та його розміру, наприклад: Pr1Prot1.jpg 450KB) та документу Специфікації (багаторядковий текст, який починається з назви продукту, а також може містити набори UserStory та набори Прототипів). Визначити інтерфейс IRequirement з методами ReadRequirement(), WriteRequirement(). Також класи UserStory, Prototype та Specification які реалізують цей інтерфейс. Реалізувати порівняння та глибоке копіювання (Clone()) у кожному класі

2) В текстовому файлі задані різні вимоги до декількох продуктів – спочатку всі UserStory, потім усі Прототипи, а далі тексти Специфікацій. Зчитати дані в колекції IRequirement. Вивести UserStory, посортовані за назвою, Прототипи - за розмірами файлів та Специфікації – за назвою продукту.

3) Колекцію Специфікацій доповнити UserStory та Прототипами, які відносяться до того самого продукту. Результат вивести у файл.

4) Перехоплення винятків.

5) Використання Linq

6) Весь код повинен бути покритий юніт тестами

Варіант 7.

1) Створити ієрархію класів Person-Student-Teacher. В кожного Teacher повинен бути список Students, якими він керує, а в кожного Student - Teacher, який ним керує. В кожному класі повинні бути віртуальні функції Input() та Print() і перевизначена функція ToString(). Для класів Person-Student-Teacher реалізувати Equals() та GetHashCode().

2) В текстовому файлі задано дані про студентів та викладачів. Ввести дані в колекцію Person. Проілюструвати роботу Clone на даній колекції – створити колекцію клонованих об’єктів. Вивести результат у файл.

3) Порахувати скільки в колекції є студентів і скільки викладачів. Продемонструвати роботу Equals() – утворивши з даної унікальну колекцію Person без повторів.

4)Перехоплення винятків.

5) Використання Linq

6) Весь код повинен бути покритий юніт тестами