

Курсова робота

з дисципліни Програмування -2. Структури даних та алгоритми
студента групи ІС-93, ФІОТ
Пінчука Артура Юрійовича

керівник: Проскура С.Л.

Постановка задачі

Власний бюджет складається з декількох рахунків, які поповнюються за заданими статтями прибутку. Гроші цих рахунків можуть бути переведені з одного на інший, можуть витрачатись за заданими статтями витрат. Підсумовуючи витрати та прибутки, можливо отримати інформацію, скільки було витрачено/отримано загалом/за певною статтею по заданому рахунку. Користувач може створювати рахунки, користуватися всіма функціями, які було перелічено вище.

Загальний опис проекту

Задля реалізації проекту було створено три класи: Account, Menu, Transfers.

ACCOUNTS

MENU



Transfers

Клас Account

Приватні поля:

- `account_name` типу `string` – унікальна назва кожного рахунку;
- `account_value` типу `decimal`. – унікальний баланс кожного рахунку.

Клас Account

Публічні методи:

- Account - конструктор класу Account
- Output - повертає назву обраного рахунку та його баланс
- Type_output - повертає звіт витрат та прибутків за заданими статтями
- Оператор + - для облегшення реалізації методу витрат/поповнення
- Adder - головний метод поповнення рахунку
- Transfer - переказ коштів з одного рахунку на інший
- Replenish - функція поповнення рахунку за заданими статтями
- Spend - функція витрат коштів за заданими статтями

Клас Menu

Клас Menu є класом, що реалізує головне меню.

Клас містить приватні поля:

- counter типу int – базовий лічильник для реалізації списку об'єктів;
- accounts типу array – список створених об'єктів класу Account;

Клас Menu

Публічні методи:

- MainMenu - викликає головне меню в консолі, є найголовнішим методом зв'язку між користувачем та програмою
- NewAcc - викликає меню для створення нових рахунків
- Accounts - повертає список усіх створених рахунків та їх баланси
- TransferMenu - викликає меню для здійснення усіх реалізованих типів транзакцій
- Summary - повертає звіт витрат та надходжень за заданими статтями для бажаного рахунку

Клас Transfers

Клас Transfers є похідним класом класу Menu. Він реалізує методи, потрібні для здійснення переказів, поповнення рахунку чи витрат коштів.

Клас містить приватні поля:

- Transfer типу delegate – делегат, потрібний для реалізації функцій витрат коштів та поповнення рахунку;
- to типу Transfer - витрати коштів;
- back типу Transfer - поповнення рахунку;

Клас Transfers

Публічні методи:

- SpendOrReplenish - викликає меню з вибором потрібного типу переказу (витрата чи поповнення). Виконує бажану дію.
- OneToAnother - викликає меню, що відповідає за переказ коштів з одного рахунку на інший.

Скріншоти програми

Оберіть, що Ви хочете зробити:

- 1 - Створити новий рахунок
 - 2 - Оглянути створені рахунки
 - 3 - Зробити переказ коштів
 - 4 - Оглянути звіт за статтями витрат та надходжень
 - 5 - Вихід
-

1

Ви можете створити до 5 рахунків.

Введіть ім'я для рахунку:

my_account

Введіть початковий баланс для рахунку:

500

Новий рахунок успішно створено!

2

Назва акаунту: my_account ||| Баланс акаунту: 500

Цей рахунок ще не створено.

Цей рахунок ще не створено.

Цей рахунок ще не створено.

Цей рахунок ще не створено.

3

Оберіть, що Ви хотіли б зробити:

- 1 - Витратити кошти / Поповнити рахунок
 - 2 - Перевести кошти з одного рахунку на інший
 - 3 - Повернутися до головного меню
-

Скріншоти програми

```
-----  
1  
-----
```

Що саме Ви хотіли б зробити?

- 1 - Витратити
 - 2 - Поповнити
 - 3 - Повернутися до головного меню
- ```

```

```

1

```

Введіть назву рахунку:

my\_account

Введіть кількість коштів, яку Ви хотіли б витратити:

400

```

```

Статті витрат:

їжа, одяг, розваги, позика

```

```

Введіть потрібну статтю витрат:

їжа

Операція успішна!

```

```

```

2

```

Введіть назву рахунку:

my\_account

Введіть кількість коштів, яку Ви хотіли б отримати:

250

```

```

Статті отримання коштів:

зарплата, позика, бізнес, подарунок

```

```

Введіть потрібну статтю отримання коштів:

зарплата

Операція успішна!

```

```

```

1

```

Ви можете створити до 5 рахунків.

Введіть ім'я для рахунку:

new\_account

Введіть початковий баланс для рахунку:

100

```

```

Новий рахунок успішно створено!

```

```

# Скріншоти програми

```

2

Назва акаунту: my_account ||| Баланс акаунту: 350

Назва акаунту: new_account ||| Баланс акаунту: 100

Цей рахунок ще не створено.

Цей рахунок ще не створено.

Цей рахунок ще не створено.

```

```

2

Введіть назву рахунку, з якого будуть зняті кошти:
my_account
Введіть назву рахунку, на який буде здійснено переказ коштів:
new_account
Введіть кількість коштів для переказу:
25
Операція успішна!

```

```

Оберіть, що Ви хочете зробити:
1 - Створити новий рахунок
2 - Оглянути створені рахунки
3 - Зробити переказ коштів
4 - Оглянути звіт за статтями витрат та надходжень
5 - Вихід

3

Оберіть, що Ви хотіли б зробити:
1 - Витратити кошти / Поповнити рахунок
2 - Перевести кошти з одного рахунку на інший
3 - Повернутися до головного меню

2

```

# Скріншоти програми

```

4

Введіть назву рахунку, для якого Ви хотіли б побачити звіт:
my_account
Звіт для: my_account
-----Надходження-----
Заробітня плата: 250
Позика: 0
Прибуток від бізнесу: 0
Подарунок: 0
-----Витрати-----
Їжа: -400
Одяг: 0
Розваги: 0
Позика: 0

```

# Використані знання та технології

- Фундаментальні поняття ООП
- Опис та використання класів
- Перевантаження функцій та операторів
- Механізми успадкування та поліморфізму класів
- Механізм обробки виняткових ситуацій
- Побудова структур даних
- Делегати

