

Wojciech Jeziorowski

Nr albumu: 49892

Python Developer gr.1 2022/2023

Projekt dostępny na: <https://github.com/WojciechJeziorowski/ProjektZaliczeniowy>

Lista Zakupów

Projekt polega na utworzeniu listy zakupów oraz umożliwia przeprowadzenie różnych operacji na liście. W projekcie została wykorzystana wiedza nabyta z kursu.

1 etapem było utworzenie menu programu:

```
def wypisz_menu():  
    print()  
    print('1. Dodaj produkt do kupienia')  
    print('2. Oznacz produkt jako kupiony')  
    print('3. Pokaż produkty do kupienia i kupione')  
    print('4. Usuń produkt z listy')  
    print('5. Ustaw dany produkt na początek listy')  
    print('6. Oznacz kupiony produkt jako niekupiony')  
    print('7. Pokaż rachunek')  
    print('8. Wyjdź z programu')
```

```

def petla_glowna_programu():
    instancje_produktow_do_kupienia = ListaZakupow()
    while True:
        wypisz_menu()
        wybor = pobierz_wybor()
        if wybor == 1:
            instancje_produktow_do_kupienia.dodaj_produkt_do_listy()
        elif wybor == 2:
            instancje_produktow_do_kupienia.oznacz_produkt_jako_kupiony()
        elif wybor == 3:
            instancje_produktow_do_kupienia.pokaz_produkty_do_kupienia_i_kupione()
        elif wybor == 4:
            instancje_produktow_do_kupienia.usun_produkt_do_kupienia()
        elif wybor == 5:
            instancje_produktow_do_kupienia.ustaw_produkt_na_poczatek_listy()
        elif wybor == 6:
            instancje_produktow_do_kupienia.oznacz_produkt_kupiony_jako_niekupiony()
        elif wybor == 7:
            instancje_produktow_do_kupienia.pokaz_rachunek()
        elif wybor == 8:
            return
    petla_glowna_programu()

```

```

1. Dodaj produkt do kupienia
2. Oznacz produkt jako kupiony
3. Pokaż produkty do kupienia i kupione
4. Usuń produkt z listy
5. Ustaw dany produkt na początek listy
6. Oznacz kupiony produkt jako niekupiony
7. Pokaz rachunek
8. Wyjdź z programu
Co chcesz zrobić? (1-8): 8

```

W funkcji `pobierz_wybor` zabezpieczona jest możliwość popełnienia błędu przy wpisywaniu numeru menu:

```
def pobierz_wybor():
    while True:
        i = input('Co chcesz zrobić? (1-8): ')
        try:
            liczba = int(i)
        except ValueError:
            print('To musi być liczba')
            continue
        if 1 <= liczba <= 8:
            return liczba
    print('To musi być liczba od 1 do 8')
```

W projekcie definiowane są 3 listy:

- lista produktów do kupienia
- lista produktów kupionych
- rachunek

```
class ListaZakupow:
    def __init__(self):
        self.lista_produkow_do_kupienia = []
        self.lista_produkow_kupionych = []
        self.rachunek = []
```

Projekt umożliwia:

1. dodanie produktów do listy zakupów

```
def dodaj_produkty_do_listy(self):
    self.lista_produkow_do_kupienia.append(input('Podaj nazwę produktu do kupienia: '))
```

2. oznaczenie produktu jako kupionego (produkt jest przenoszony z listy produktów do kupienia na listę produktów kupionych (wybór konkretnego produktu odbywa się przez podanie indeksu produktu (pobierz_indeks)):

```
def oznacz_produkty_jako_kupione(self):
    indeks_kupionego_produkty = self._pobierz_indeks()
    self.lista_produkow_kupionych.append(self.lista_produkow_do_kupienia[indeks_kupionego_produkty])
    del self.lista_produkow_do_kupienia[indeks_kupionego_produkty]
    cena = self._podaj_cene()
    self.rachunek.append(cena)
```

```

def _pobierz_indeks(self, lista=None):
    if lista is None:
        lista = self.lista_produktow_do_kupienia
    if len(lista) == 0:
        print('Najpierw dodaj produkt do listy zakupów')
        petla_glowna_programu()
    while True:
        print()
        for indeks, produkt_do_kupienia in enumerate(lista):
            print(f'\t\t{indeks} [ ] {produkt_do_kupienia}')
        print()
        try:
            indeks_produktu = int(input('Który indeks: '))
        except ValueError:
            print('To musi być liczba')
            continue
        if indeks_produktu >= 0 and indeks_produktu < len(lista):
            return indeks_produktu
        else:
            print('Podana liczba jest spoza przedziału')

```

Funkcja `pobierz_indeks` ma zabezpieczenia przed podaniem złego indeksu produktu oraz odsyła nas do głównego menu w momencie gdy lista produktów do kupienia jest pusta (zabezpieczenie przed pustą listą funkcjonuje dla wszystkich logicznych przypadków gdyby użytkownik chciał operować na produktach które nie zostały zdefiniowane).

Po przeniesieniu produktu na drugą listę program pyta o cenę zakupu produktu (`podaj_cene`). Funkcja również jest zabezpieczona przed możliwymi błędami.

```

def _podaj_cene(self):
    while True:
        try:
            cena = float(input('Podaj cenę zakupionego produktu [PLN]:'))
        except ValueError:
            print('To musi być liczba')
            continue
        if cena >= 0:
            return cena
        else:
            print('Cena musi być liczbą nieujemną')

```

Po podaniu ceny, jej wartość wpisywana jest na 3 listę (rachunek).

3. Wyświetlenie list produktów do kupienia i kupionych:

```
def pokaz_produkty_do_kupienia_i_kupione(self):
    print()
    print('Do kupienia jest: ')
    for indeks, produkty_do_kupienia in enumerate(self.lista_produktow_do_kupienia):
        print(f'\t\t{indeks} [ ] {produkty_do_kupienia}')
    print()
    print('Kupiono: ')
    for indeks, produkty_kupione in enumerate(self.lista_produktow_kupionych):
        print(f'\t\t{indeks} [x] {produkty_kupione}')
```

```
Do kupienia jest:
    0 [ ] chleb
    1 [ ] jajko

Kupiono:
    0 [x] mleko
```

Wyświetlany jest indeks i nazwa produktu oraz znacznik że produkt został kupiony.

4. Usunięcie wybranego produktu z listy:

```
def usun_produkty_do_kupienia(self):
    indeks_produkty_do_usuniecia = self._pobierz_indeks()
    del self.lista_produktow_do_kupienia[indeks_produkty_do_usuniecia]
```

5. Ustawienie danego produktu na początek listy:

```
def ustaw_produkty_na_poczatek_listy(self):
    indeks_produkty_do_kupienia = self._pobierz_indeks()
    element_na_poczatek = self.lista_produktow_do_kupienia[indeks_produkty_do_kupienia]
    del self.lista_produktow_do_kupienia[indeks_produkty_do_kupienia]
    self.lista_produktow_do_kupienia.insert(0, element_na_poczatek)
```

6. Oznaczenie produktu kupionego jako niekupionego (np. w przypadku popełnienia błędu przez użytkownika):

```
def oznacz_produkty_kupiony_jako_niekupiony(self):
    indeks_produkty_kupionego = self._pobierz_indeks(self.lista_produktow_kupionych)
    self.lista_produktow_do_kupienia.append(self.lista_produktow_kupionych[indeks_produkty_kupionego])
    del self.lista_produktow_kupionych[indeks_produkty_kupionego]
    del self.rachunek[indeks_produkty_kupionego]
```

Jednocześnie wybrany produkt jest usuwany z rachunku (bo jednak nie został zakupiony).

7. Wyświetlenie rachunku:

```

def pokaz_rachunek(self):
    print()
    suma = 0
    print('Twój rachunek to: ')
    for indeks, cena in enumerate(self.rachunek):
        print(f'\t\t{indeks} -> {round(cena, 2)} PLN')
        suma += cena
    suma = round(suma, 2)
    print('Łącznie:', suma, 'PLN')
    print()
    while True:
        wybor = input('Czy chcesz zapisać rachunek do pliku? t/n')
        if wybor == 't' or wybor == 'T':
            while not os.path.exists('rachunek.txt'):
                print('Plik nie istnieje, utwórz plik rachunek.txt')
                time.sleep(5)
            rachunek_string = str(self.rachunek)
            with open('rachunek.txt', 'w') as fp:
                fp.writelines(rachunek_string)
            return
        elif wybor == 'n' or wybor == 'N':
            return
        else:
            print('Wpisz "t" lub "n" ')

```

```

Twój rachunek to:
    0 -> 10.0 PLN
    1 -> 5.56 PLN
    2 -> 13.47 PLN
Łącznie: 29.03 PLN

```

Wyświetlane są indeksy produktów oraz ich ceny oraz suma wszystkich zakupionych produktów. Ceny są zaokrąglane do 2 miejsca po przecinku. Dodatkowo program pyta czy zapisać rachunek do pliku (funkcja posiada zabezpieczenia przed udzieleniem nieprawidłowej odpowiedzi, oraz czeka na utworzenie pliku tekstowego w przypadku gdy nie został on utworzony oraz ponownie sprawdza jego stworzenie co 5 sekund. Do pliku zapisywany jest rachunek przekształcony wcześniej na stringa.

```

Czy chcesz zapisać rachunek do pliku? t/n

```

8. Wyjście z programu

Do utworzenia programu zaimportowałem biblioteki os.path oraz time:

```
import os.path  
import time
```

W programie utworzona została klasa, funkcje, funkcje pomocnicze (z podkreślnikiem) wykorzystano obiektowość Pythona, pętle, instrukcje warunkowe, podstawowe operacje na listach, operacje na plikach, utworzono zabezpieczenia przed błędami.