

```
1 def menu_glowne():
2     print("Wybierz opcje: ")
3     print("1. Wpłata")
4     print("2. Wypłata")
5     print("3. Sprawdzenie stanu konta")
6     print("4. Zakończ")
7     # 6. Dodanie opcji
8     print("5. Wyświetlenie historii transakcji")
9
10 def pobierz_wybor_klienta():
11     return int(input("Podaj numer wybranej opcji: "))
12
13 def pobierz_kwote(tekst):
14     return float(input(tekst))
15
16 def pokaz_stan_konta(saldo):
17     print(f"Stan wynosi {saldo} złotych")
18
19 def wpłata(saldo):
20     kwota_wplaty = pobierz_kwote("Ile chcesz wpłacić? : ")
21     saldo = saldo + kwota_wplaty
22     print(f"Wpłacono {kwota_wplaty} złotych")
23     pokaz_stan_konta(saldo)
24     return saldo
25
26 def wypłata(saldo):
27     kwota_wyplaty = pobierz_kwote("Ile chcesz wypłacić? : ")
28     if kwota_wyplaty <= saldo:
29         saldo -= kwota_wyplaty
30         print(f"Wypłacono {kwota_wyplaty} złotych")
31         pokaz_stan_konta(saldo)
32         return saldo
33     else:
34         print("Operacja nieudana, za mało środków na koncie")
35         return saldo
36
37 def pokaz_stan_konta(saldo):
38     print(f"Stan wynosi {saldo} złotych")
39
40 def pobierz_dane(dana):
41     return input(f"Podaj numer {dana}: ")
42
43 def sprawdz_zgodnosc_danych(baza, karta, pin):
44     for klient in baza:
45         if klient[0] == karta:
46             return klient[1] == pin
47     return False
48
49 def pobierz_stan_konta(baza, karta):
50     for klient in baza:
51         if klient[0] == karta:
52             return klient[2]
53     return 0
```

```
54 #-----
55 # Etap 4 - Historia transakcji
56 # - dodanie do każdego użytkownika w liście pustej listy
57 #     * w tych listach będą przechowywane operacje wykonane na koncie
58 # - funkcja aktualizująca historię klienta
59 # - funkcja pokazująca całą historię klienta
60 # - dodanie do menu opcji '5-wyświetlenie historii'
61
62 # 2.
63 def aktualizuj_historie_klienta(baza, karta, operacja):
64     for klient in baza:
65         if klient[0] == karta:
66             klient[3].append(operacja)
67
68
69 # 3.
70 def pokaz_historie_klienta(baza, karta):
71     for klient in baza:
72         if klient[0] == karta:
73             for i in range(len(klient[3])):
74                 print(f'{i+1}. {klient[3][i]}')
75
76
77 wybor = 0
78 saldo = 0
79
80 # 1. Dodanie pustych list
81 KLIENCI = [#numer karty, PIN, saldo
82            # [0]      [1]      [2]
83            ["0001", "1234", 0,      []],  #[0]
84            ["0002", "1111", 120,    []],  #[1]
85            ["0003", "3232", 1000,   []],  #[2]
86            ["0004", "4450", 12.50, []]    #[3]
87 ]
88
89
90 podana_karta = pobierz_dane("karty")
91 podany_pin = pobierz_dane("PIN")
92
93
94 if sprawdz_zgodnosc_danych(KLIENCI, podana_karta, podany_pin):
95     saldo = pobierz_stan_konta(KLIENCI, podana_karta)
96
97     while wybor != 4:
98         menu_glowne()
99         wybor = pobierz_wybor_klienta()
100
101     # 4. Dodanie zapisywania operacji
102     if wybor == 1:
103         saldo = wplata(saldo)
104
105         aktualizuj_historie_klienta(KLIENCI, podana_karta, "Wpłata")
106
107     pass
```

```
108     elif wybor == 2:
109         saldo = wypłata(saldo)
110
111         aktualizuj_historie_klienta(KLIENCI, podana_karta, "Wypłata")
112
113         pass
114     elif wybor == 3:
115         pokaz_stan_konta(saldo)
116
117         aktualizuj_historie_klienta(KLIENCI, podana_karta, "Wyświetlenie salda")
118
119         pass
120     elif wybor == 4:
121         print("Wyłączanie bankomatu")
122         pass
123
124     # 5. Opcja wyświetlenia historii
125     elif wybor == 5:
126         pokaz_historie_klienta(KLIENCI, podana_karta)
127         pass
128     else:
129         print("niepoprawne dane")
130         pass
131     pass
132 else:
133     print("Błąd logowania")
```