

```

1 from product import Product
2 from order import Order
3 from customer import Customer
4 from discount import Discount
5
6
7 """
8 1. Создаются товары, заказы и клиенты.
9 2. Демонстрируется функциональность добавления
   заказов к клиенту.
10 3. Демонстрируется применение различных видов скидок
    к заказам.
11 4. Проводится подсчёт общего количество заказов и
    общей сумму всех заказов для всех клиентов.
12 5. Выводится информация о клиентах, заказах и
    продуктах с использованием дандер методов.
13 6. Проверяются статический метод (Discount)и методы
    класса (Order)
14 """
15 # Товары
16 print('Товары: создание и вывод информации
    -----')
17 car_charger = Product('Car Charger 100W', 1228.0)
18 print(car_charger)
19 usb_type_C_charging = Product('Зарядка USB + Type-C'
    , 769.0)
20 print(usb_type_C_charging)
21 print('Цены равны: ', car_charger ==
    usb_type_C_charging)
22 print('цена USB < car_charger :', usb_type_C_charging
    < car_charger)
23 fm_transmitter = Product('FM модулятор с Bluetooth в
    машину', 1194.0)
24 print(fm_transmitter)
25 bluetooth_modulator = Product('модулятор блютуз в
    машину', 1439.0)
26 print(bluetooth_modulator)
27 laptop_work_14_SSD = Product('Ноутбук для работы SD-
    256GB', 15400.0)
28 print(laptop_work_14_SSD)
29

```

```
30 #Заказы
31 print('Заказы: создание, проверка функциональности и
      вывод информации')
32 #Заказ №1
33 order_1 = Order ('a+106z', [car_charger,
      usb_type_C_charging])
34 print('Создание первого заказа
      -----')
35 print(order_1)
36 #Товары заказа
37 print('Первый товар заказа')
38 print(order_1.get_products()[0])
39 #Добавление товаров в заказ
40 print('Проверка функциональности')
41 print('Добавление товаров в заказ
      -----')
42 order_1.add_product([fm_transmitter,
      laptop_work_14_SSD ])
43 print(order_1)
44 #Товары заказа
45 print('Первый товар заказа')
46 print(order_1.get_products()[0])
47 #Удаление товаров из заказа
48 print('Удаление товаров из заказа
      -----')
49 order_1.del_product([car_charger])
50 print(order_1)
51 #Товары заказа
52 print('Первый товар заказа')
53 print(order_1.get_products()[0])
54
55 #Заказ №2
56 order_2 =Order ('a+116z', [laptop_work_14_SSD,
      bluetooth_modulator])
57 print('Создание второго заказа
      -----')
58 print(order_2)
59 #Товары заказа
60 print('Последний товар заказа')
61 print(order_2.get_products()[-1])
62
```

```
63 #Заказ №3
64 order_3 = Order('a+116z', [laptop_work_14_SSD,
    fm_transmitter, bluetooth_modulator])
65 print('Создание третьего заказа
    -----')
66 print(order_3)
67 #Товары заказа
68 print('Последний товар заказа')
69 print(order_3.get_products()[-1])
70
71 # Клиенты
72 print('Клиенты: создание, проверка функциональности
    и вывод информации')
73 #Клиент №1
74 customer_1 = Customer('Иванов Иван Иванович', [
    order_1])
75 print('Создание первого клиента
    -----')
76 print(customer_1)
77 #Добавление заказов клиенту
78 print('Проверка функциональности')
79 print('Добавление заказов в список заказов клиента
    -----')
80 customer_1.add_order([order_2])
81 print(customer_1)
82 #Заказы клиента
83 print('Последний заказ клиента')
84 print(customer_1.get_orders()[-1])
85 #Удаление товаров из заказа
86 print('Удаление заказов из списка заказов клиента
    -----')
87 customer_1.del_order([order_2])
88 print(customer_1)
89
90 #Клиент №2
91 customer_2 = Customer('Мишулина Ольга Александровна'
    , [order_2, order_3])
92 print('Создание второго клиента
    -----')
93 print(customer_2)
94 #Товары заказа
```

```

95 print('Последний заказ клиента')
96 print(customer_2.get_orders()[-1])
97
98 # Скидки
99 print('Скидки: создание, проверка функциональности и
      вывод информации')
100 discount_list = {
101     'сезонная скидка' : 40.0,
102     'скидка по промокоду' : 30.0,
103     'пенсионная скидка' : 20.0
104 }
105 print('Создание скидок
      -----
      --')
106 discount = Discount(discount_list)
107 print(discount)
108 #Добавление скидок
109 print('Добавление скидок
      -----')
110 add_discounts = {
111     'скидка в Якузу' : 10.5,
112     'скидка в Лопасню' : 20.0
113 }
114 discount.add_discounts (add_discounts)
115 print(discount)
116 #Удаление скидок
117 print('Удаление скидок
      -----')
118 key_discounts = ['пенсионная скидка', 'скидка в
      Лопасню']
119 discount.del_discounts(key_discounts)
120 print(discount)
121 # Стоимость товаров первого заказа со скидкой '
      сезонная скидка'
122 print('Стоимость товаров первого заказа со скидкой "
      сезонная скидка"')
123 order_discont_cost_1 = discount.
      get_order_discounted_cost(order_1,'сезонная скидка')
124 print(order_1)
125 print(f'Стоимость товаров заказа со скидкой : {round
      (order_discont_cost_1,2)}')
```

```
126 # Стоимость товаров первого клиента со скидкой '
    скидка в Яузу'
127 print('Стоимость товаров первого заказа со скидкой "
    скидка в Яузу"')
128 customer_discont_cost_1 = discount.
    get_customer_discounted_cost (customer_1, 'скидка в
    Яузу')
129 print(customer_1)
130 print(f'Стоимость товаров клиента со скидкой : {
    round(customer_discont_cost_1,2)}')
131 #Проверка метода класса Order для подсчета общего
    количества заказов для всех клиентов
132 print('Проверка метода класса Order для подсчета
    общего количества заказов для всех клиентов')
133 customer_list = [customer_1, customer_2]
134 total_number_orders = Order.get_total_number_orders(
    customer_list)
135 print (f'Общее количество заказов по клиентам = {
    total_number_orders}')
136 #Проверка метода класса Order для подсчета общей
    стоимости заказов для всех клиентов
137 print('Проверка метода класса Order для подсчета
    общей стоимости заказов для всех клиентов')
138 total_cost_order = Order.get_total_cost_orders(
    customer_list, discount, 'сезонная скидка')
139 print(f'Общая стоимость заказов всех клиентов с
    скидкой "сезонная скидка": {round(total_cost_order,2
    )}')
140 # Проверка статического метода класса Discount для
    расчета цены с указанной скидкой
141 print('Проверка статического метода класса Discount
    для расчета цены с указанной скидкой')
142 discont_price = Discount.get_discount_price(1450.0,
    discount_list,'скидка по промокоду')
143 print (f'цена: {1450.0} \n Скидка по промокоду: {
    discount_list.get('скидка по промокоду')} \n цена со
    скидкой: {discont_price}' )
144
145
146
147
```

148

149