

【7】 プログラムの説明を読んで、プログラムの(1)～(5)を答えなさい。

<プログラムの説明>

処理内容

店舗データおよび売上データを読み、指定した番号の処理結果を表示するプログラムである。

入力データ

店舗データ (ファイル名: shop.csv)

店舗コード	店舗名	前月利用者数	前月売上金額
×××	××～××	××××	×××××××

(第1図)

売上データ (ファイル名: selling.csv)

売上番号	顧客コード	値引金額	店舗コード	売上金額
××××	××～××	××××	×××	×××××

(第2図)

実行結果

```

処理番号を入力してください。
1:前月売上 2:今月売上 3:前月比売上 0:終了 =>1
(店舗コード) (店 舗 名) (前月利用者数) (前月売上金額)
 153      ××～××      58人      60,474円
  }      }      }      }
 168      ××～××      10人      10,362円
1:前月売上 2:今月売上 3:前月比売上 0:終了 =>2
(店舗コード) (店 舗 名) (今月利用者数) (今月売上金額) (値引金額計) (値引人数)
 284      ××～××      61人      81,519円      1,500円      5人
  }      }      }      }      }      }
 142      ××～××      12人      11,290円      0円      0人
1:前月売上 2:今月売上 3:前月比売上 0:終了 =>3
  }
    
```

(第3図)

処理条件

- 第1図の店舗データを読み、店舗情報を管理する ShopInfo クラスをインスタンス化し、配列 shop に記憶する。なお、店舗データは100件以内である。

配列

shop	(0)	(1)	~	(98)	(99)

- 第2図の売上データを読み、次の処理を行う。
 - 値引金額に 0 が入力されている場合、値引きは行わない。
 - 店舗コードをもとに配列 shop を探索し、店舗別に今月の利用者数と売上金額を集計する。なお、値引きした場合は、売上金額から値引きするとともに、値引金額計と値引人数を集計する。
 - 店舗別に、前月比の売上金額を求める。
- データを読み終えたあと、キーボードから入力された処理番号に応じて配列 shop を降順に並べ替えて、第3図のようにディスプレイに表示する。なお、処理番号は次のとおりである。

1:前月売上 2:今月売上 3:前月比売上 0:終了

< J a v a プ ロ グ ラ ム >

//クラスShopInfo

```

public class ShopInfo {
    private int code, countBefore, countPeople,
               salesBefore, sales, discount,
               compare, discountPeople;
    private String name;
    public ShopInfo(int code, String name,
                   int countBefore, int salesBefore) {
        this.code = code;
        this.name = name;
        this.countBefore = countBefore;
        this.salesBefore = salesBefore;
        countPeople = 0;
        sales = 0;
        discount = 0;
        compare = salesBefore * (-1);
        discountPeople = 0;
    }
    public int getCode() {
        (1);
    }
    public int getItem(int number) {
        int item = 0;
        switch(number) {
            case 1:
                item = salesBefore;
                break;
            case 2:
                item = sales;
                break;
            case 3:
                item = compare;
                break;
        }
        return item;
    }
}
    
```

```

public void calcTotal(int sales) {
    countPeople = countPeople + 1;
    this.sales = this.sales + sales;
    compare = compare + sales;
}
public void calcTotal(int sales, int discount) {
    countPeople = countPeople + 1;
    this.sales = this.sales + sales - discount;
    this.discount = this.discount + discount;
    compare = compare + sales - discount;
    discountPeople = discountPeople + 1;
}
public void outList(int number) {
    System.out.printf("¥t%3d ¥t%-10s", code, name);
    switch(number) {
        case 1:
            System.out.printf(" ¥t%4d人 ¥t%, 7d円¥n",
                               countBefore, salesBefore);
            break;
        case 2:
            System.out.printf(" ¥t%4d人 ¥t%, 7d円",
                               countPeople, sales);
            System.out.printf(" ¥t%, 7d円 ¥t%4d人¥n",
                               discount, discountPeople);
            break;
        case 3:
            System.out.printf(" ¥t%4d人 ¥t%, 7d円¥n",
                               countPeople - countBefore, compare);
            break;
    }
}
    
```

```

//クラスSalesAnalysis
import java.io.BufferedReader;
import java.io.FileReader;
import java.util.Scanner;

public class SalesAnalysis {
    private static ShopInfo[] shop = new ShopInfo[100];
    private static int index;
    private static void sort(int number) {
        for(int k = 1; k <= index; k++) {
            ShopInfo hoz = shop[k];
            int m;
            for(m = k - 1; m >= 0 && shop[m].getItem(number) < hoz.getItem(number); m--) {
                (2);
            }
            if(m != k - 1) {
                shop[m + 1] = hoz;
            }
        }
    }
    public static void main(String[] args) {
        index = 0;
        try {
            BufferedReader fileIn = new BufferedReader(new FileReader("shop.csv"));
            String line;
            while((line = fileIn.readLine()) != null) {
                String[] str = line.split(",");
                int code = Integer.parseInt(str[0]);
                String name = str[1];
                int countBefore = Integer.parseInt(str[2]);
                int salesBefore = Integer.parseInt(str[3]);
                shop[index] = new ShopInfo(code, name, countBefore, salesBefore);
                (3);
            }
            index = index - 1;
            fileIn.close();
            fileIn = new BufferedReader(new FileReader("selling.csv"));
            while((line = fileIn.readLine()) != null) {
                String[] str = line.split(",");
                int no = Integer.parseInt(str[0]);
                int customerCode = Integer.parseInt(str[1]);
                int discount = Integer.parseInt(str[2]);
                int code = Integer.parseInt(str[3]);
                int sales = Integer.parseInt(str[4]);
                int j = 0;
                while(shop[j].getCode() != code) {
                    j = j + 1;
                }
                if(discount == 0) {
                    shop[j].calcTotal(sales);
                } else {
                    (4);
                }
            }
            fileIn.close();
        } catch(Exception e) {
            System.out.println("エラーが発生しました" + e);
        }
        System.out.println("処理番号を入力してください。");
        Scanner keyboardIn = new Scanner(System.in);
        System.out.print("1：前月売上    2：今月売上    3：前月比売上    0：終了 =>");
        int number = keyboardIn.nextInt();
        while(number != 0) {
            (5);
            for(int j = 0; j <= index; j++) {
                shop[j].outList(number);
            }
            System.out.print("1：前月売上    2：今月売上    3：前月比売上    0：終了 =>");
            number = keyboardIn.nextInt();
        }
        keyboardIn.close();
    }
}

```