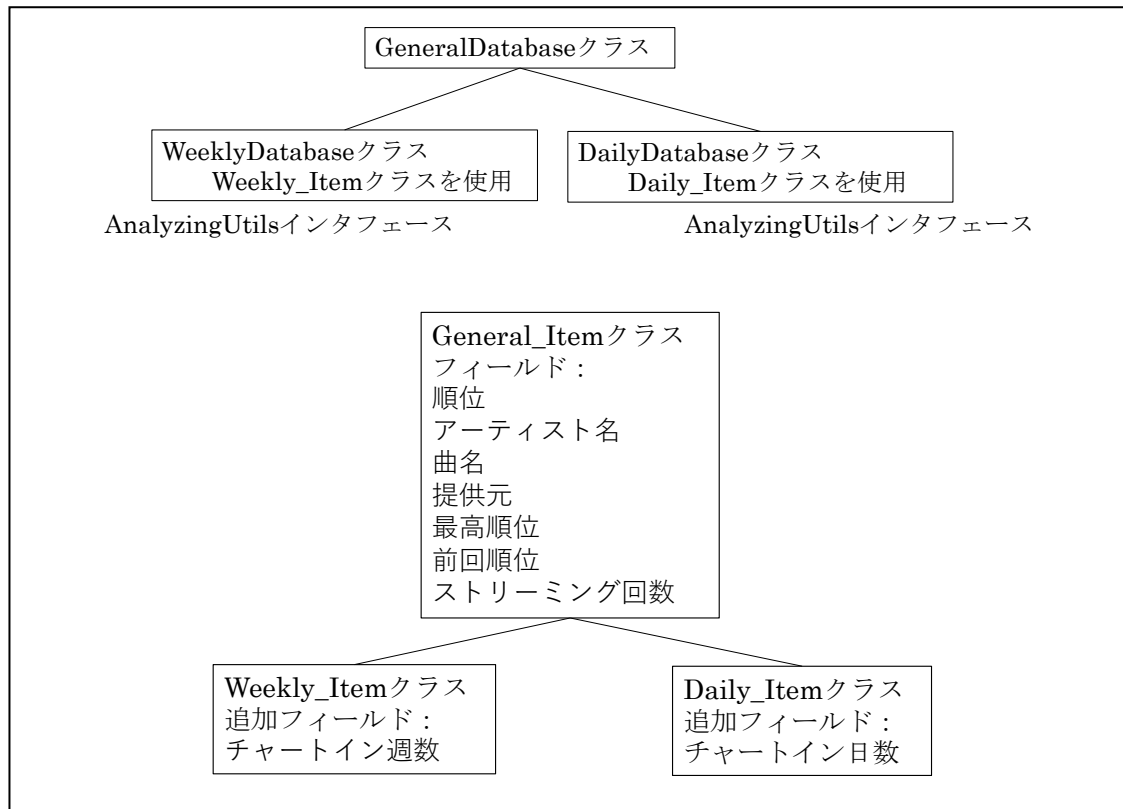


1. 課題

Spotify の週のランキングデータと今日(本日)のランキングデータのファイルを読み込み、様々な集計を行うプログラムを作成したい。週のデータと本日のデータは多少内容が異なっているので、以下のようなクラスを設計した。



各クラスの仕様を以下に与える。

GeneralDatabase クラス

フィールド名	型	説明
DataList	ArrayList <General_Item>	データのリスト

メソッド名	戻り値	引数	説明
printDatabase	void	ArrayList <General_Item>	データ内容を表示
getTrackRanking	int	String	曲の順位を戻す
getTop20Artists	void	なし	ストリーミング回数 上位 20 のアーティスト を表示

WeeklyDatabase クラス (GeneralDatabase クラスを拡張, AnalyzingUtils インタフェースを実装)

フィールド名	型	説明
DataList	ArrayList <General_Item>	データのリスト

メソッド名	戻り値	引数	説明
loadData	void	String	データファイルの読み込み
getChartIn	int	String	チャートイン週数を戻す
printDatabase	void	ArrayList <General_Item>	データ内容を表示
getTrackRanking	int	String	曲の順位を戻す
getTop20Artists	void	なし	ストリーミング回数上位 20 のアーティストを表示

DailyDatabase クラス (GeneralDatabase クラスを拡張, AnalyzingUtils インタフェースを実装)

フィールド名	型	説明
DataList	ArrayList <General_Item>	データのリスト

メソッド名	戻り値	引数	説明
loadData	void	String	データファイルの読み込み
getChartIn	int	String	チャートイン日数を戻す
printDatabase	void	ArrayList <General_Item>	データ内容を表示
getTrackRanking	int	String	曲の順位を戻す
getTop20Artists	void	なし	ストリーミング回数上位 20 のアーティストを表示

AnalyzingUtils インタフェース

メソッド名	戻り値	引数	説明
loadData	void	String	データファイルの読み込み
getChartIn	int	String	チャートイン日数を戻す

General_Item クラス (抽象クラス)

フィールド名	型	説明
rank	int	順位
artist_names	String	アーティスト名
track_name	String	曲名
source	String	提供元
peak_rank	int	最高順位
previous_rank	int	前回順位
streams	int	ストリーミング回数

メソッド名	戻り値	引数	説明
printData (抽象メソッド)	void	なし	曲データの表示

Weekly_Item.クラス (General_Item クラスを拡張)

フィールド名	型	説明
rank	int	順位
artist_names	String	アーティスト名
track_name	String	曲名
source	String	提供元
peak_rank	int	最高順位
previous_rank	int	前回順位
streams	int	ストリーミング回数
weeks_on_chart	int	チャートイン週数

メソッド名	戻り値	引数	説明
printData	void	なし	曲データの表示

General_Item クラス (General_Item クラスを拡張)

フィールド名	型	説明
rank	int	順位
artist_names	String	アーティスト名
track_name	String	曲名
source	String	提供元
peak_rank	int	最高順位
previous_rank	int	前回順位
streams	int	ストリーミング回数
days_on_chart	int	チャートイン日数

メソッド名	戻り値	引数	説明
printData	void	なし	曲データの表示

また, AnalyzingUtils インタフェースと General_Item クラスの定義を以下に与える.

AnalyzingUtils インタフェース

```
public interface AnalyzingUtils {  
    public abstract void loadData(String filename);  
    public abstract int getChartIn(String kyoku);  
}
```

General_Item クラス

```
abstract public class General_Item {  
    int rank;  
    String artist_names;  
    String track_name;  
    String source;  
    int peak_rank;  
    int previous_rank;  
    int streams;  
  
    General_Item() {  
        this.rank = -1;  
        this.artist_names = "noname";  
        this.track_name = "noname";  
        this.source = "nosource";  
        this.peak_rank = -1;  
        this.previous_rank = -1;  
        this.streams = -1;  
    }  
  
    abstract void printData();  
}
```

これらのクラスを利用して、様々な集計を行うプログラム、SpotifyAnalysis.java を以下に示す。このファイルは CoursePower にアップロードされている。

```
import java.io.BufferedReader;
import java.io.InputStreamReader;

public class SpotifyAnalysis {

    public static void main(String[] args) {
        String filename_w = args[0];
        String filename_d = args[1];
        WeeklyDatabase wd = new WeeklyDatabase();
        wd.loadData(filename_w);
        wd.printDatabase();
        DailyDatabase dd = new DailyDatabase();
        dd.loadData(filename_d);
        dd.printDatabase();

        System.out.println("***** 今週のストリーミング回数トップ20のアーティスト *****");
        wd.getTop20Artists();
        System.out.println("***** 今日のストリーミング回数トップ20のアーティスト *****");
        dd.getTop20Artists();
        String input;

        while(true) {
            System.out.println("何を調べますか？ 1:週間ランキング
2:本日ランキング 3:チャートイン週数 4:チャートイン日数 5:終了");
            try {
                BufferedReader reader = new BufferedReader(new
InputStreamReader(System.in));
                input=reader.readLine();
                if (input.equals("5")) {
                    break;
                } else if (input.equals("1")) {
                    System.out.println("トラック名を入力
->");

                    input = reader.readLine();
                    int num = wd.getTrackRanking(input);
                    System.out.println(input+" の週間ランキ
ング:"+num);
                } else if (input.equals("2")) {
                    System.out.println("トラック名を入力
->");

                    input = reader.readLine();
                    int num = dd.getTrackRanking(input);
                    System.out.println(input+" の本日ランキ
ング:"+num);
                } else if (input.equals("3")) {
                    System.out.println("トラック名を入力
->");

                    input = reader.readLine();
                    int num = wd.getChartIn(input);
```

```

        System.out.println(input+" のチャートイ
ン週数:"+num);
    } else if (input.equals("4")) {
        System.out.println("トラック名を入力
->");
        input = reader.readLine();
        int num = dd.getChartIn(input);
        System.out.println(input+" のチャートイ
ン日数:"+num);
    } else {
        System.out.println("入力エラー");
    }
} catch (Exception e) {
    e.printStackTrace();
}
}
}
}

```

SpotifyAnalysis.java をテスト用プログラムとし、出力例に示すような出力が得られるプログラムを完成させよ。

(ヒント)

- GeneralDatabase クラスで作成される DataList は Genral_Item のリストとして宣言されているが、GeneralDatabase クラスを継承している WeeklyDatabase クラスや DailyDatabase クラスでは、DataList を Weekly_Item や Daily_Item のリストとして作成することができる。

実行例（下線部は入力を示す）

```

ランキング：1 アーティスト名："Official HIGE DANdism" 楽曲名："Subtitle" チャートイン週数：9 ス
トリーミング回数：3231883
ランキング：2 アーティスト名："Kenshi Yonezu" 楽曲名："KICK BACK" チャートイン週数：9 ストリ
ーミング回数：2422613
ランキング：3 アーティスト名："natori" 楽曲名："Overdose" チャートイン週数：13 ストリーミング回
数：1770534
（途中省略...）
ランキング：197 アーティスト名："Lilas Ikuta" 楽曲名："JUMP" チャートイン週数：1 ストリーミン
グ回数：241330
ランキング：198 アーティスト名："yama" 楽曲名："春を告げる" チャートイン週数：135 ストリーミン
グ回数：241165
ランキング：199 アーティスト名："back number" 楽曲名："ハッピーエンド" チャートイン週数：146 ス
トリーミング回数：240962
ランキング：200 アーティスト名："RM" 楽曲名："Wild Flower (with youjeen)" チャートイン週数：1 ス
トリーミング回数：238848
ランキング：1 アーティスト名："Official HIGE DANdism" 楽曲名："Subtitle" チャートイン週数：64 ス
トリーミング回数：477336
ランキング：2 アーティスト名："Kenshi Yonezu" 楽曲名："KICK BACK" チャートイン週数：64 ストリ
ーミング回数：336818
ランキング：3 アーティスト名："natori" 楽曲名："Overdose" チャートイン週数：93 ストリーミング回
数：239680
ランキング：4 アーティスト名："Tani Yuuki" 楽曲名："W / X / Y" チャートイン週数：353 ストリーミ
ング回数：227727

```

ランキング：5 アーティスト名："Ado" 楽曲名："新時代" チャートイン週数：189 ストリーミング回数：198168

(途中省略...)

ランキング：197 アーティスト名："King Gnu" 楽曲名："BOY" チャートイン週数：381 ストリーミング回数：34009

ランキング：198 アーティスト名："Official HIGE DANDism" 楽曲名："宿命" チャートイン週数：1035 ストリーミング回数：33943

ランキング：199 アーティスト名："Vaundy" 楽曲名："裸の勇者" チャートイン週数：330 ストリーミング回数：33555

ランキング：200 アーティスト名："Kep1er" 楽曲名："WA DA DA" チャートイン週数：320 ストリーミング回数：33452

***** 今週のストリーミング回数トップ 20 のアーティスト *****

1:"Official HIGE DANDism"：7068043

2:"back number"：6283592

3:"Ado"：5546276

(途中省略...)

17:"Aimyon"：1618498

18:"IVE"：1532941

19:"RADWIMPS"：1409875

20:"LE SSERAFIM"：1408328

***** 今日のストリーミング回数トップ 20 のアーティスト *****

1:"Official HIGE DANDism"：1039394

2:"back number"：913903

3:"Ado"：691814

(途中省略...)

17:"Utada Hikaru"：213481

18:"Aimyon"：208353

19:"IVE"：191999

20:"LE SSERAFIM"：187582

何を調べますか？ 1:週間ランキング 2:本日ランキング 3:チャートイン週数 4:チャートイン日数 5:終了

1

トラック名を入力->

"Lemon"

"Lemon" の週間ランキング:93

何を調べますか？ 1:週間ランキング 2:本日ランキング 3:チャートイン週数 4:チャートイン日数 5:終了

2

トラック名を入力->

"Lemon"

"Lemon" の本日ランキング:88

何を調べますか？ 1:週間ランキング 2:本日ランキング 3:チャートイン週数 4:チャートイン日数 5:終了

3

トラック名を入力->

"Lemon"

"Lemon" のチャートイン週数:123

何を調べますか？ 1:週間ランキング 2:本日ランキング 3:チャートイン週数 4:チャートイン日数 5:終了

4

トラック名を入力->

"Lemon"

"Lemon" のチャートイン日数:862

何を調べますか？ 1:週間ランキング 2:本日ランキング 3:チャートイン週数 4:チャートイン日数 5:終了

5

★入力データは以下の2つのファイルである。これらも **CoursePower** にアップロードされている。

regional-jp-daily-2022-12-13.csv

regional-jp-weekly-2022-12-08.csv

2. レポートの形式

レポートには少なくとも以下の項目を記載すること。用紙のサイズはA4とする。

1. 表紙（学籍番号、氏名、レポート提出日）
2. 課題内容（簡潔に示す）
3. 解決手順（詳細に説明）
4. 実行結果
5. 考察
6. プログラムリスト
7. 参考文献(ある場合のみ)
8. その他(例えば感想など。ある場合のみ)

3. 提出方法

提出期限：1月13日（金）

作成したソースコードはzipファイルにまとめて、**CoursePower**にアップロードすること。
また、レポートのファイルは別のメニューを用意するので、そこにアップロードすること。