Comparable <T>

`<<Interface>>`
Comparable <T>

+ *compareTo(T o): int*

Movie

- title: String
- year: String
- form: String
- multiple: String
- serialVersionUID: long

+ Movie()
+ Movie(t: String, y: String, f: String, m: String)
+ getTitle(): String
+ getYear(): String
+ getForm(): String
+ getMultiple(): String
+ parse(line: String): void
+ compareTo(m: Movie): int
+ toString(): String

Series

- title: String
- startYear: String
- yearRange: String
- episodes: ArrayList<Episode>
- serialVersionUID: long

+ Series()
+ Series(t: String, sY: String, yR: String)
+ parse(line: String): void
+ getTitle(): String
+ getStartYear(): String
+ getYearRange(): String
+ getEpisodes(): ArrayList<Episode>
+ addEpisode(e: Episode): void
+ compareTo(s: Series): int
+ toString(): String

Episode

- episodeTitle: String
- number: String
- releaseYear: String
- serialVersionUID: long

+ Episode()
+ Episode(t: String, n: String, y: String)
+ parse(line: String): void
+ getTitle(): String
+ getNumber(): String
+ getYear(): String
+ toString(): String
+ compareTo(e: Episode): int

MediaMaker

- lastName: String
- firstName: String
- acting: ArrayLIst<Actor>
- directing: ArrayList<Director>
- producing: ArrayList<Producer>
- serialVersionUID: long

+ MediaMaker()
+ MediaMaker(l: string, f: String)
+ parseActorInfo(info: String): void
+ parseDirectorInfo(info: String): void
+ parseProducerInfo(info: String): void
+ compareTo(mm: MediaMaker): int

«interface»
Serializable

DatabaseModel

- actionListenerList: ArrayList<ActionListener>

+ addActionListener(l: ActionListener): void
+ removeActionListener(l: ActionListener): void
+ processEvent(e: ActionEvent): void

DatabaseController

+ field: type

+ DatabaseController()

DatabaseView

- model: DatabaseModel

+ DatabaseView()

Database

+ movies: ArrayList<Movie>
+ series: ArrayList<Series>
+ mediaMakers: LinkedHashMap<String, MediaMaker>
- serialVersionUID: long

+ Database()
+ insertMovie(m: Movie): void
+ insertSeries(s: Series): void
+ insertMediaMaker(mm: MediaMaker): void
+ writeDatabase(d: Database, file String): void
+ readDatabase(file: String): Database

---

java.awt.event.ActionListener

`<<Interface>>`
java.awt.event.ActionListener

+ actionPerformed(e: ActionEvent): void

Actor

- actorInfo:
ArrayList\<String\>
- moviesIn:
ArrayList\<Movie\>
- seriesIn: ArrayList\<Series\>
- serialVersionUID: long

+ Actor()
+ parseActorInfo(info:
String): void

Director

- DirectorInfo:
ArrayList\<String\>
- moviesDirected:
ArrayList\<Movie\>
- seriesDirected:
ArrayList\<Series\>
- serialVersionUID: long

+ Director()
+ parseDirectorInfo(info:
String): void

Producer

- producerInfo:
ArrayList\<String\>
- moviesProduced:
ArrayList\<Movie\>
- seriesProduced:
ArrayList\<Series\>
- serialVersionUID: long

+ Producer()
+ parseProducerInfo(info:
String): void

Driver

+ mdb: Database

+ main(args: String[]): void
+ loadBinary(br:
BufferedReader): void
+ save(br: BufferedReader):
void