

+ clone() : item*

- insertionDateMin : QDate* - insertionDateMax : QDate* - publicationDateMin : QDate* - publicationDateMax : QDate* - genre : QString* - author : QString* - pagesMin : unsigned int* - pagesMax : unsigned int* - sizeMin : float* - sizeMax: float* - actor : QString* - durationMin : QTime* - durationMax : QTime* - artist : QString* - type : item* + filter() + ~filter() + getInsertionDateMin() : const QDate* + getInsertionDateMax() : const QDate* + getPublicationDateMin() : const QDate* + getPublicationDateMax() : const QDate* + getGenre() : const QString* + getSizeMin() : const float* + getSizeMax() : const float* + getAuthor() : const QString* + getPagesMin() : const unsigned int* + getPagesMax() : const unsigned int* + getActor() : const QString* + getDurationMin() : const QTime* + getDurationMax() : const QTime* + getArtist() : const QString* + getType() : const item* + setInsertionDateMin(iDateMin : const QDate&) : void + setInsertionDateMax(iDateMax : const QDate&) : void + setPublicationDateMin(pDateMin : const QDate&) : void + setPublicationDateMax(pDateMax : const QDate&) : void + setGenre(genr : const QString&) + setSizeMin(szMin : const float&) : void + setSizeMax(szMax : const float&) : void + setAuthor(aut : const QString&) + setPagesMin(pgMin : const unsigned int&)
+ setPagesMax(pgMax : const unsigned int&) + setActor(act : const QString&) : void + setDurationMin(durMin : const QTime&) : void + setDurationMax(durMax : const QTime&) : void + setArtist(art : const QString&) : void + setType(typ : item*) : void

collection

- items: QVector<item*>

+ collection()
+ collection(other: const collection&)
+ ~collection()
+ operator=(other: const colletion&): collection&

copy(target: QVector<item*>&, toCopy: const QVector<item*>&): void
destroy(vec: QVector<item*>&)

+ addItem(item: item*): void
+ removeItem(item: item*): void
+ clearAll(): void

+ getItems(): const QVector<item*>&

+ applyFilter(const filter&): QVector<item*>
+ makeSearch(const QString& keyword): QVector<item*>
+ saveCollectionAs(): bool
+ loadCollection(): bool