**Files:-**

1. mediaItemClass.h
2. itemClasses.h
3. libraryClass.h
4. defineClass.cpp
5. main.cpp

**mediaItemClass.h**

**#include <iostream>**

**using namespace std;**

**#ifndef DEF**

**#define DEF**

**class mediaItem**

**{**

**string title;**

**string creater;**

**int r\_year;**

**public:**

**mediaItem(string, string, int);**

**void setTitle(string);**

**void setCreater(string);**

**void setYear(int);**

**string getTitle() const;**

**string getCreater() const;**

**int getYear() const;**

**};**

**#endif**

**itemClasses.h**

#include "mediaItemClass.h"

class Book : virtual public mediaItem

{

string author;

int nbr\_pages;

public:

Book(string, string, int, int);

void setAuthor(string);

void setNbrPages(int);

string getAuthor() const;

int getNbrPages() const;

};

class Movie : virtual public mediaItem

{

string director;

int duration;

public:

Movie(string, string, int, int);

void setDirector(string);

void setDuration(int);

string getDirector() const;

int getDuration() const;

};

class musicAlbum : virtual public mediaItem

{

string artist;

int nbr\_track;

public:

musicAlbum(string, string, int, int);

void setArtist(string);

void setNbrTrack(int);

string getArtist() const;

int getNbrTrack() const;

};

**libraryClass.h**

#include "itemClasses.h"

#include <vector>

class Library

{

vector<Book> books;

vector<Movie> movies;

vector<musicAlbum> musics;

public:

// Library();

void addBooks(string, string, int, int);

void addMovies(string, string, int, int);

void addMusics(string, string, int, int);

void displayBooks();

void displayMovies();

void displayMusics();

};

**defineClass.cpp**

#include "libraryClass.h"

// mediaItem

mediaItem::mediaItem(string title = "", string creater = "", int r\_yrear = 0)

{

this->title = title;

this->creater = creater;

if (r\_year > 0)

this->r\_year = r\_year;

}

void mediaItem::setTitle(string title)

{

this->title = title;

}

void mediaItem::setCreater(string creater)

{

this->creater = creater;

}

void mediaItem::setYear(int year)

{

if (year > 0)

this->r\_year = year;

}

string mediaItem::getTitle() const

{

return this->title;

}

string mediaItem::getCreater() const

{

return this->creater;

}

int mediaItem::getYear() const

{

return this->r\_year;

}

// Book

Book::Book(string title = "", string author = "", int r\_year = 0, int nbr\_pages = 0) : mediaItem(title, author, r\_year)

{

this->author = author;

if (nbr\_pages > 0)

this->nbr\_pages = nbr\_pages;

}

void Book::setAuthor(string author)

{

setCreater(author);

this->author = author;

}

void Book::setNbrPages(int pages)

{

if (pages >= 0)

this->nbr\_pages = pages;

}

string Book::getAuthor() const

{

return author;

}

int Book::getNbrPages() const

{

return nbr\_pages;

}

// Movie

Movie::Movie(string title = "", string director = "", int r\_year = 0, int duration = 0) : mediaItem(title, director, r\_year)

{

this->director = director;

if (duration >= 0)

this->duration = duration;

}

void Movie::setDirector(string director)

{

setCreater(director);

this->director = director;

}

void Movie::setDuration(int duration)

{

if (duration >= 0)

this->duration = duration;

}

string Movie::getDirector() const

{

return director;

}

int Movie::getDuration() const

{

return duration;

}

// musicAlbum

musicAlbum::musicAlbum(string title = "", string artist = "", int r\_year = 0, int nbr\_track = 0) : mediaItem(title, artist, r\_year)

{

this->artist = artist;

if (nbr\_track >= 0)

this->nbr\_track = nbr\_track;

}

void musicAlbum::setArtist(string artist)

{

setCreater(artist);

this->artist = artist;

}

void musicAlbum::setNbrTrack(int track)

{

if (track >= 0)

this->nbr\_track = track;

}

string musicAlbum::getArtist() const

{

return artist;

}

int musicAlbum::getNbrTrack() const

{

return nbr\_track;

}

// library

void Library::addBooks(string title, string author, int r\_year, int pages)

{

Book book(title, author, r\_year, pages);

books.push\_back(book);

}

void Library::addMovies(string title, string director, int r\_year, int duration)

{

Movie movie(title, director, r\_year, duration);

movies.push\_back(movie);

}

void Library::addMusics(string title, string artist, int r\_year, int nbr\_track)

{

musicAlbum music(title, artist, r\_year, nbr\_track);

musics.push\_back(music);

}

void Library::displayBooks()

{

cout<<"Books "<<"Title "<<"Author "<<"Rease Year"<<"Pages"<<endl;

for(auto b:books)

{

cout<<b.getTitle()<<""<<b.getAuthor()<<""<<b.getYear()<<""<<b.getNbrPages()<<endl;

}

}

void Library::displayMovies()

{

cout<<"Movies "<<"Title "<<"Director "<<"Rease Year"<<"Duration"<<endl;

for(auto m:movies)

{

cout<<m.getTitle()<<""<<m.getDirector()<<""<<m.getYear()<<""<<m.getDuration()<<endl;

}

}

void Library::displayMusics()

{

cout<<"Musics "<<"Title "<<"Artist "<<"Rease Year"<<"Tracks"<<endl;

for(auto m:musics)

{

cout<<m.getTitle()<<""<<m.getArtist()<<""<<m.getYear()<<""<<m.getNbrTrack()<<endl;

}

}

**main.cpp**

#include "libraryClass.h"

int main()

{

Library lib;

string title;

string creater;

int r\_year;

int nbr;

int ch;

while (ch != 7)

{

cout << endl

<< "Library:- "

<< "\t1. Add\_Books" << endl

<< "\t2. Add\_Movies" << endl

<< "\t3. Add\_Musics" << endl

<< "\t4. Display Books List" << endl

<< "\t5. Display Movies List" << endl

<< "\t\t6. Display Musics List" << endl

<< "\t7. exit" << endl

<< "choose option:- ";

cin >> ch;

switch (ch)

{

case 1:

cout << "Title:- ";

;

cin >> title;

cout << "Author:- ";

;

cin >> creater;

cout << "Release Year:- ";

;

cin >> r\_year;

cout << "Number of Pages:- ";

;

cin >> nbr;

lib.addBooks(title, creater, r\_year, nbr);

break;

case 2:

cout << "Title:- ";

;

cin >> title;

cout << "Director:- ";

;

cin >> creater;

cout << "Release Year:- ";

;

cin >> r\_year;

cout << "Duration(hr):- ";

;

cin >> nbr;

lib.addMovies(title, creater, r\_year, nbr);

break;

case 3:

cout << "Title:- ";

;

cin >> title;

cout << "Artist:- ";

;

cin >> creater;

cout << "Release Year:- ";

;

cin >> r\_year;

cout << "Number of Tracks:- ";

;

cin >> nbr;

lib.addMusics(title, creater, r\_year, nbr);

break;

case 4:

lib.displayBooks();

case 5:

lib.displayMovies();

case 6:

lib.displayMusics();

case 7:

cout << endl

<< "You are exit" << endl;

break;

default:

cout << endl

<< "Wrong input !please try again" << endl;

break;

}

}

return 0;

}

//output







