

```
// -----media.h-----
// Adam Ali / Adam Mirza CSS 343 A
// Created: 05/21/17
// Modified: 05/23/17
// -----
// Describes the abstract ADT Media such that any kind of item can be held
// in the store. All Media items have stock counts and formats (DVD, etc).
// Derived classes of Media simply add any additional attributes that pertain
// to them. They are all equipped with operations to maintain the stocks,
// display the details, and make comparisons between one another.
//
// This class is an interface for any type of Media in a store. It serves as
// a guide for being able to keep track of the inventory of a Media and what
// format that media is. There are methods to checkout, return, and compare
// Media.
// -----
// Functionality includes:
//      - create a Media item
//      - copy an exist Media item
//      - destruct a Media item
//      - retrieve the format
//      - retrieve stock counts
//      - increment/decrement stock counts
//      - display details
//      - comparison operators (==, <)

#pragma once

#include <string>
using namespace std;

class Media {
public:
    // -----Media-----
    // Media: creates a Media object with specified stock and format.
    // preconditions: format non-empty string, e.g. "DVD"
    //                  stock >= 0
    // postconditions: specified Media item created.
    // -----
    Media(int, string);

    // -----Media-----
    // Media: creates a copy of the Media item.
    // preconditions: none.
    // postconditions: a copy of the Media is created.
    // -----
    Media(const Media&);

    // -----~Media-----
    // ~Media: destructs the Media and frees any assoc. memory.
    // preconditions: none.
    // postconditions: any assoc. memory is freed, object inaccessible.
    // -----
    virtual ~Media();

    // -----getFormat-----
    // getFormat: obtains format, e.g. "DVD", "VHS", etc.
    // preconditions: none.
    // postconditions: format is returned.
    // -----
    string getFormat() const;
}
```

```
// -----getCount-----
// getCount: obtains stock count available for check out.
// preconditions: none.
// postconditions: stock is returned.
// -----
int getCount() const;
```

```
// -----increaseCount-----
// increaseCount: increments stock count, typically after a return.
// preconditions: none.
// postconditions: stock = stock + 1.
// -----
void increaseCount();
```

```
// -----decreaseCount-----
// decreaseCount: decrements stock count, typically after a borrow.
// preconditions: none.
// postconditions: stock = stock - 1.
// -----
void decreaseCount();
```

```
// -----display-----
// display: outputs Media details, pertaining to particular style.
// preconditions: none.
// postconditions: Media details output to console.
// -----
virtual void display() const = 0;
```

```
// -----operator==-----
// operator==: determines if both Media items are identical.
// preconditions: none.
// postconditions: true if identical, otherwise false.
// -----
virtual bool operator==(const Media&) const = 0;
```

```
// -----operator<-----
// operator<: determines if this Media precedes the other Media.
// preconditions: none.
// postconditions: true if preceding, otherwise false.
// -----
virtual bool operator<(const Media&) const = 0;
```

private:

```
int count;           // amount available for checkout
string format; // DVD, VHS, Book, Magazine, etc.
```

```
};
```