University of British Columbia, Vancouver
Department of Computer Science

_____

CPSC 304 Project Cover Page

Milestone #: ____2____

Date: ___Mar 4th, 2021_____

Group Number: _____6____

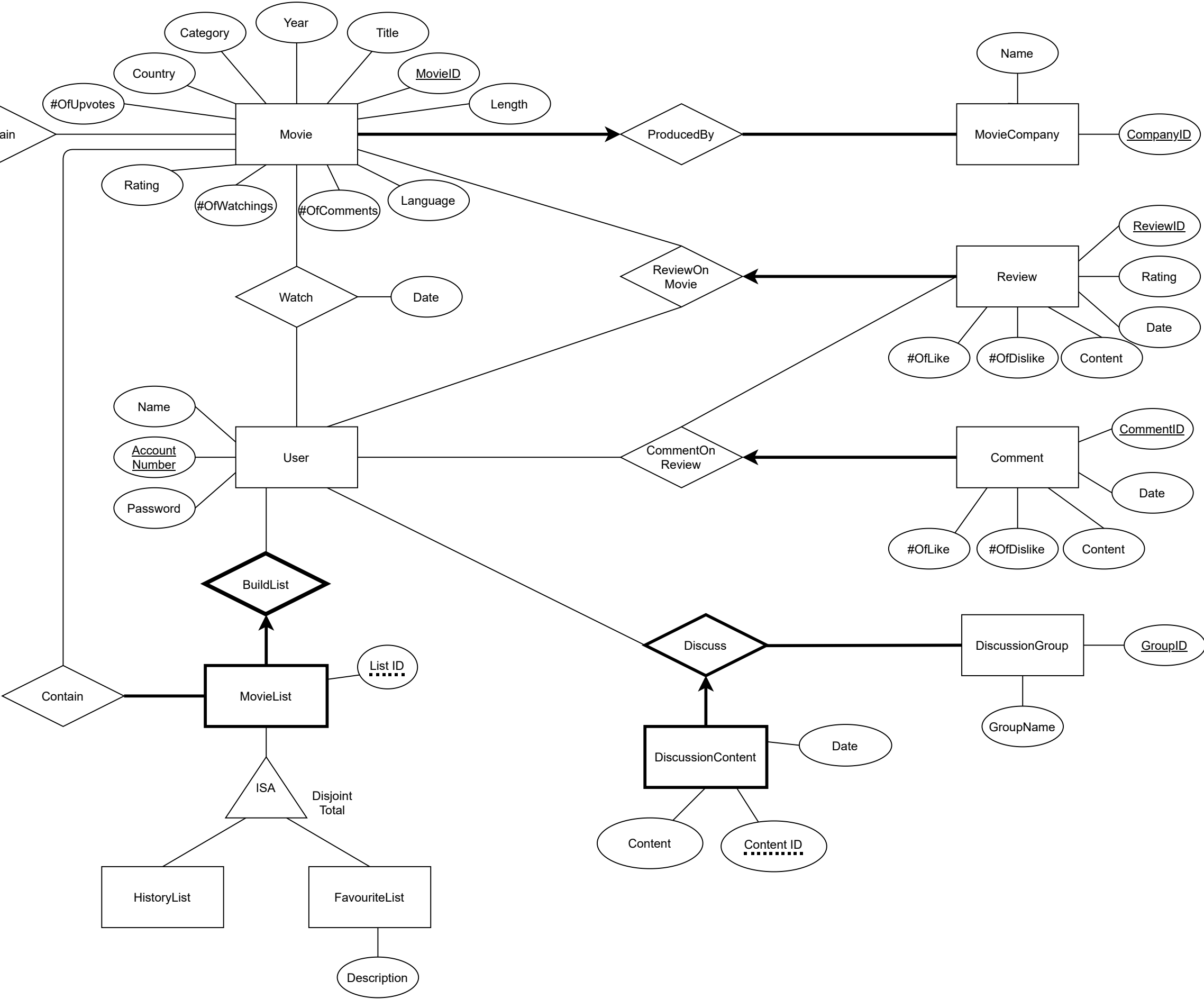| Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Runhe Guo (Josh)	17975517	z8x2b	grh20010102@gmail.com
Jia Jun Cheng Xian	55325294	t7z2b	anthonychengxian@gmail.com
Ruixin Yang	21315726	t7f3b	yangstarman@hotmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Part 2: ER diagram

Notes on Changes:

1. Add an attribute "Language" to Movie, in order to have an additional non-PK/CK Functional Depenency (Country->Language)

2. Add an attribute "Rating" to Movie. It's reasonale for a movie to have a rating. We missed that earlier somehow.

3. Remove "WatchList" which was previously a subclass of "MovieList", since it's very similar to "FavouriteList", and seems excessive.

4. Remove the attribute "ListName" of MovieList, and add an attribute "Description" to FavouriteList. The reason is that we think in our design the history list need not have a particular name, whereas the favourite lists that a user keep can have detailed descriptions, such as "10 favourite action movies"

RankingName

Ranking

RankingID

Contain

#OfUpvotes

Category · Year · Title

Country

MovieID

Movie

Length

ProducedBy

Name

MovieCompany

CompanyID

Rating

#OfWatchings · #OfComments · Language

Watch · Date

ReviewOn Movie

ReviewID

Review

Rating

Date

#OfLike · #OfDislike · Content

Name

Account Number

User

Password

CommentOn Review

CommentID

Comment

Date

#OfLike · #OfDislike · Content

BuildList

List ID

MovieList

Contain

Discuss

DiscussionGroup

GroupID

GroupName

DiscussionContent

Date

Content · Content ID

ISA · Disjoint Total

HistoryList · FavouriteList

Description

Part 3 and 4: Relational Schema and Functional Dependencies

underline is primary key and bold would be foreign key
trivial functional dependencies are ignored
Based on our design, for each table, the current primary key is the only possible candidate key. There are no additional CKs.

1. Ranking (RankingName: char(20), RankingID: int)
FD: RankingID -> RankingName

2. Movie (MovieID: int, Length: int, Title: char(20), Year: int, Category: char(20), Country: char(20), Rating: int, #ofUpvotes: int, #ofWatchings: int, #ofComments: int, **CompanyID**: int, Language: char(20))
        #Upvotes, #ofWatchings, #Comments DEFAULT 0
        Length, Title, Year, Category, Country, Rating, Language NOT NULL
FD:
MovieID -> Length, Title, Year, Category, Country, #ofUpvotes, #ofWatchings, #ofComments, CompanyID, Rating, Language
Country -> Language
Title, Year -> CompanyID

3. RMContain (**RankingID**: int, **MovieID**: int)

4. MovieCompany (Name: char(20), CompanyID: int)
        Name NOT NULL, Name UNIQUE
FD: CompanyID -> Name

5. User （Name: char(20), AccountNumber: int, Password: char(20)) Name UNIQUE
FD: AccoutNumber ->   Name, Password

6. Watch (**MovieID**: int, **AccountNumber**: int, Date: date)
FD: MovieID, AccountNumber -> Date

7. MovieList (ListID: int, **AccountNumber**: int)

8. HistoryList (ListID: int)

9. FavouriteList (ListID: int, Description: char(500))
FD: ListID -> Description

10. MMContain (**MovieID**: int, **ListID**: int, **AccountNumber**: int)

11. RReview (#ofLike: int, #ofDislike: int, Content: char(500), Date: date, Rating: int, ReviewID: int, **MovieID**: int, **AccountNumber**: int)
        Content NOT NULL, #ofLike DEFAULT 0, #ofDislike DEFAULT 0
FD: ReviewID ->#ofLike, #ofDislike, Content, Date, Rating, MovieID, AccountNumber

12. CComment (#ofLike: int, #ofDislike: int, Content: char(500), Date: date, CommentID: int, **ReviewID**: int, **AccountNumber**: int)

Content NOT NULL, #ofLike DEFAULT 0, #ofDislike DEFAULT 0
FD: CommentID ->#ofLike, #ofDislike, Content, Date,ReviewID, AccountNumber

13. DiscussionGroup (<u>GroupID</u>: int, GroupName: char(20)) GroupName NOT NULL
FD: GroupID -> GroupName

14. DiscussionContent (<u>ContentID</u>: int, **<u>GroupID</u>**: int, **<u>AccountNumber</u>**: int, Content: char(500),
     Date: date)
Content NOT NULL
FD: ContentID, GroupID, AccountNumber -> Content, Date

Part 5: Normalization

Ranking (RankingName, <u>RankingID</u>)

MovieCountryInfo (Language, <u>Country</u>)

MovieCompanyInfo (<u>Title, Year</u>, **CompanyID**)

MovieBasicInfo (**Title, Year**, <u>MovieID</u>, Length, Category, **Country**, Rating, #ofUpvotes, #ofWatchings, #ofComments)

RMContain (**<u>RankingID</u>, <u>MovieID</u>**)

MovieCompany (Name, <u>CompanyID</u>)

User (<u>AccountNumber</u>, Name, Password)

Watch (**<u>MovieID</u>**, **<u>AccountNumber</u>**, Date)

MovieList (<u>ListID</u>, **AccountNumber**)

HistoryList (<u>ListID</u>)

FavouriteList (<u>ListID</u>, Description)

MMContain (**<u>MovieID, ListID, AccountNumber</u>**)

RReview (#Like, #ofDislike, Content, Date, Rating, <u>ReviewID</u>, **MovieID, AccountNumber**)

CComment (#Like, #ofDisLike, Content, Date, <u>CommentID</u>, **ReviewID, AccountNumber**)

DiscussionGroup (GroupID, GroupName)

DiscussionContent (ContentID, **GroupID, AccountNumber**, Content, Date)

Part 6 SQL CREATE TABLE statements

```sql
CREATE TABLE Ranking(
    RankingName CHAR(20),
    RankingID   INT  PRIMARY KEY
 );
CREATE TABLE MovieCountryInfo(
    Language CHAR(20) NOT NULL,
    Country  CHAR(20) PRIMARY KEY
 );
CREATE TABLE MovieCompanyInfo(
    Title     CHAR(20) NOT NULL,
    Year      INT      NOT NULL,
    CompanyID INT,
    PRIMARY KEY (Title, Year),
    FOREIGN KEY (CompanyID) REFERENCES MovieCompany(CompanyID)
             ON DELETE CASCADE
 );

CREATE TABLE MovieBasicInfo(
   MovieID INT PRIMARY KEY,
   Title CHAR(20) NOT NULL,
   Year INT NOT NULL,
   Length INT NOT NULL,
   Category CHAR(20) NOT NULL,
   Country CHAR(20) NOT NULL,
   Rating INT NOT NULL,
   #ofUpvotes INT SET DEFAULT 0,
   #ofWatchings INT SET DEFAULT 0,
   #ofComments INT SET DEFAULT 0,
   FOREIGN KEY (Title, Year) REFERENCES MovieCompanyInfo(Title,
Year),
   FOREIGN KEY (Country) REFERENCES MovieCountryInfo(Country)
             ON DELETE CASCADE
             ON UPDATE CASCADE
);
```

```sql
CREATE TABLE RMContain(
    RankingID INT,
    MovieID INT,
    PRIMARY KEY (RankingID,MovieID),
    FOREIGN KEY (RankingID) REFERENCES Ranking(RankingID),
    FOREIGN KEY (MovieID) REFERENCES MovieBasicInfo(MovieID)
             ON DELETE CASCADE
);
CREATE TABLE MovieCompany(
    Name CHAR(20) NOT NULL,
    CompanyID INT PRIMARY KEY
    UNIQUE Name
);
CREATE TABLE User(
    AccountNumber INT PRIMARY KEY,
    Name CHAR(20),
    Password CHAR(20)
    UNIQUE Name
);

CREATE TABLE Watch(
    MovieID INT,
    AccountNumber INT,
    Date DATE,
    PRIMARY KEY(MovieID,AccountNumber)
);

CREATE TABLE MovieList(
    ListID INT,
    AccountNumber INT,
    PRIMARY KEY(ListID,AccountNumber),
    FOREIGN KEY(AccountNumber) REFERENCES User(AccountNumber)
);
CREATE TABLE HistoryList(
    ListID INT,
    PRIMARY KEY(ListID)
);
CREATE TABLE FavouriteList(
    ListID INT,
    Description CHAR(500),
    PRIMARY KEY(ListID)
);

CREATE TABLE MMContain(
    MovieID INT,
```

```sql
    ListID INT,
    AccountNumber INT,
    PRIMARY KEY(MovieID, ListID, AccountNumber),
    FOREIGN KEY(AccountNumber) REFERENCES User(AccountNumber),
    FOREIGN KEY(MovieID) REFERENCES MovieBasicInfo(MovieID),
    FOREIGN KEY(ListID) REFERENCES MovieList(ListID)
);

CREATE TABLE RReview(
    #ofLike INT SET DEFAULT 0,
    #ofDislike INT SET DEFAULT 0,
    Content CHAR(500) NOT NULL,
    Date DATE,
    Rating INT,
    ReviewID INT PRIMARY KEY,
    MovieID INT,
    AccountNumber INT,
    FOREIGN KEY(AccountNumber) REFERENCES User(AccountNumber),
    FOREIGN KEY(MovieID) REFERENCES MovieBasicInfo(MovieID)
);

CREATE TABLE CComment(
    #ofLike INT SET DEFAULT 0,
    #ofDislike INT SET DEFAULT 0,
    Content CHAR(500) NOT NULL,
    Date DATE,
    CommentID INT PRIMARY KEY,
    ReviewID INT,
    AccountNumber INT,
    FOREIGN KEY(AccountNumber) REFERENCES User(AccountNumber),
    FOREIGN KEY(ReviewID) REFERENCES RReview(ReviewID)
);

CREATE TABLE DiscussionGroup(
    GroupID INT PRIMARY KEY,
    GroupName CHAR(20) NOT NULL
);

CREATE TABLE DiscussionContent(
    ContentID INT,
    GroupID INT,
    AccountNumber INT,
    Content CHAR(500) NOT NULL,
    Date DATE,
    PRIMARY KEY(ContentID, GroupID, AccountNumber),
    FOREIGN KEY(AccountNumber) REFERENCES User(AccountNumber),
    FOREIGN KEY(GroupID) REFERENCES DiscussionGroup(GroupID)
```

```
);
```
Part 7 Populate

```sql
INSERT INTO Ranking VALUES ('Latest', 1);
INSERT INTO Ranking VALUES ('Highest Rating', 2);
INSERT INTO Ranking VALUES ('Most Watching', 3);
INSERT INTO Ranking VALUES ('Most Upvotes', 4);
INSERT INTO Ranking VALUES ('Most Comments', 5);
```

Ranking

RankingName	RankingID
Latest	1
Highest Rating	2
Most Watching	3
Most Upvotes	4
Most Comments	5

```sql
INSERT INTO MovieCountryInfo VALUES ('English', 'Canada');
INSERT INTO MovieCountryInfo VALUES ('English', 'USA');
INSERT INTO MovieCountryInfo VALUES ('Mandarin', 'China');
INSERT INTO MovieCountryInfo VALUES ('Japanese', 'Japan');
INSERT INTO MovieCountryInfo VALUES ('English', 'Australia');
```

MovieCountryInfo

Language	Country
English	Canada
English	USA
Mandarin	China
Japanese	Japan
English	Australia

```sql
INSERT INTO MovieCompanyInfo VALUES ('Monster Hunter', 2020, 10000);
INSERT INTO MovieCompanyInfo VALUES ('Greenland', 2020, 20000);
INSERT INTO MovieCompanyInfo VALUES ('The Little Things', 2021, 30000);
INSERT INTO MovieCompanyInfo VALUES ('Endgame', 2021, 40000);
INSERT INTO MovieCompanyInfo VALUES ('The Yinyang Master', 2021 50000);
```

MovieCompanyInfo

Title	Year	CompanyID
Monster Hunter	2020	10000
Greenland	2020	20000
The Little Things	2021	30000
Endgame	2021	40000
The Yinyang Master	2021	50000

```sql
INSERT INTO MovieBasicInfo
VALUES ('Monster Hunter', 2020, 1, 104, 'Fantasy', 'Canada', 0, 0, 0, 0);
INSERT INTO MovieBasicInfo
VALUES ('Greenland', 2020, 2, 119 'Disaster', 'USA', 0, 0, 0, 0);
INSERT INTO MovieBasicInfo
VALUES ('The Little Things', 2021, 3, 128, 'Crime', 'USA', 0, 0, 0, 0);
INSERT INTO MovieBasicInfo
VALUES ('Endgame', 2021, 4, 119, 'Crime', 'China', 0, 0, 0, 0);
INSERT INTO MovieBasicInfo
VALUES ('The Yinyang Master', 2021, 5, 120 'Fantasy', 'China', 0, 0, 0, 0);
```

MovieBasicInfo

Title	Year	MovieID	Length	Category	Country	Rating	#ofUpvotes	#ofWatchings	#ofComments
Monster Hunter	2020	1	104	Fantasy	Canada	0	0	0	0
Greenland	2020	2	119	Disaster	USA	0	0	0	0
The Little Things	2021	3	128	Crime	USA	0	0	0	0
Endgame	2021	4	119	Crime	China	0	0	0	0
The Yinyang Master	2021	5	120	Fantasy	China	0	0	0	0

```
INSERT INTO RMContain VALUES (1, 1);
INSERT INTO RMContain VALUES (1, 3);
INSERT INTO RMContain VALUES (1, 4);
INSERT INTO RMContain VALUES (2, 2);
INSERT INTO RMContain VALUES (2, 4);
INSERT INTO RMContain VALUES (2, 5);
INSERT INTO RMContain VALUES (3, 1);
INSERT INTO RMContain VALUES (3, 2);
INSERT INTO RMContain VALUES (3, 3);
INSERT INTO RMContain VALUES (4, 3);
INSERT INTO RMContain VALUES (4, 4);
INSERT INTO RMContain VALUES (4, 5);
INSERT INTO RMContain VALUES (5, 1);
INSERT INTO RMContain VALUES (5, 3);
INSERT INTO RMContain VALUES (5, 5);
```

RMContain

RankingID	MovieID
1	1
1	3
1	4
2	2
2	4
2	5
3	1
3	2
3	3
4	3
4	4
4	5
5	1
5	3
5	5

```sql
INSERT INTO MovieCompany VALUES ('Looking Glass Studios', 10000);
INSERT INTO MovieCompany VALUES ('Continuum Films', 20000);
INSERT INTO MovieCompany VALUES ('Off the Spectrum Productions', 30000);
INSERT INTO MovieCompany VALUES ('Pendulum Film Company', 40000);
INSERT INTO MovieCompany VALUES ('Sandstone Films', 50000);
```

MovieCompany

Name	CompanyID
Looking Glass Studios	10000
Continuum Films	20000
Off the Spectrum Productions	30000
Pendulum Film Company	40000
Sandstone Films	50000

```sql
INSERT INTO User VALUES (692630, 'ERburETE', 'Kc0I8iYM&2*xgw^s14Mn');
INSERT INTO User VALUES (469738, 'ndRIzErT', 'sb&YvvY*NvoeaBH70Ew1');
INSERT INTO User VALUES (279183, 'terThisk', 'QYXwgaT*NL&QYGmor@$@');
INSERT INTO User VALUES (628591, 'OMeragMa', 'gnEvZg^VBRrgupsfiF*L');
INSERT INTO User VALUES (904166, 'OMEISMIc', '*XD*7E@Hx6ZJ#B28dZEt');
```

User

AccountNumber	Name	Password
692630	ERburETE	Kc0I8iYM&2*xgw^s14Mn
469738	ndRIzErT	sb&YvvY*NvoeaBH70Ew1
279183	terThisk	QYXwgaT*NL&QYGmor@$@
628591	OMeragMa	gnEvZg^VBRrgupsfiF*L
904166	OMEISMIc	*XD*7E@Hx6ZJ#B28dZEt

```sql
INSERT INTO Watch VALUES (1, 904166, '2021-4-14');
INSERT INTO Watch VALUES (1, 469738, '2021-4-16');
INSERT INTO Watch VALUES (1, 628591, '2021-4-28');
INSERT INTO Watch VALUES (1, 279183, '2021-5-11');
INSERT INTO Watch VALUES (2, 692630, '2021-6-10');
INSERT INTO Watch VALUES (2, 279183, '2021-6-20');
INSERT INTO Watch VALUES (2, 469738, '2021-7-19');
INSERT INTO Watch VALUES (3, 628591, '2021-7-21');
INSERT INTO Watch VALUES (4, 628591, '2021-7-22');
INSERT INTO Watch VALUES (4, 692630, '2021-10-1');
INSERT INTO Watch VALUES (5, 692630, '2021-10-3');
INSERT INTO Watch VALUES (5, 469738, '2021-10-27');
INSERT INTO Watch VALUES (5, 279183, '2021-10-30');
INSERT INTO Watch VALUES (5, 628591, '2021-11-8');
```

Watch

MovieID	AccountNumber	Date
1	904166	'2021-4-14'
1	469738	'2021-4-16'
1	628591	'2021-4-28'
1	279183	'2021-5-11'
2	692630	'2021-6-10'
2	279183	'2021-6-20'
2	469738	'2021-7-19'
3	628591	'2021-7-21'
4	628591	'2021-7-22'
4	692630	'2021-10-1'
5	692630	'2021-10-3'
5	469738	'2021-10-27'
5	279183	'2021-10-30'
5	628591	'2021-11-8'

```sql
INSERT INTO MovieList VALUES (11, 692630);
INSERT INTO MovieList VALUES (22, 469738);
INSERT INTO MovieList VALUES (33, 279183);
INSERT INTO MovieList VALUES (44, 628591);
INSERT INTO MovieList VALUES (55, 904166);
INSERT INTO MovieList VALUES (111, 692630);
INSERT INTO MovieList VALUES (222, 469738);
INSERT INTO MovieList VALUES (333, 279183);
INSERT INTO MovieList VALUES (444, 628591);
INSERT INTO MovieList VALUES (555, 904166);
```

MovieList

ListID	AccountNumber
11	692630
22	469738
33	279183
44	628591
55	904166
111	692630
222	469738
333	279183
444	628591
555	904166

```sql
INSERT INTO HistoryList VALUES (11);
INSERT INTO HistoryList VALUES (22);
INSERT INTO HistoryList VALUES (33);
INSERT INTO HistoryList VALUES (44);
INSERT INTO HistoryList VALUES (55);
```

HistoryList

ListID
11
22
33
44
55

```sql
INSERT INTO FavouriteList VALUES (111, 'My favourite action movies');
INSERT INTO FavouriteList VALUES (222, 'My favourite dramas');
INSERT INTO FavouriteList VALUES (333, 'Starring Jackie Chen');
INSERT INTO FavouriteList VALUES (444, 'My favourite fiction movies');
INSERT INTO FavouriteList VALUES (555, 'Oscars');
```

FavouriteList

ListID	Description
111	'My favourite 10 action movies'
222	'My favourite dramas'
333	'Starring Jackie Chen'
444	'My favourite fiction movies'
555	'Oscars'

```sql
INSERT INTO MMcontain VALUES (1, 11, 904166);
INSERT INTO MMcontain VALUES (3, 11, 904166);
INSERT INTO MMcontain VALUES (3, 111, 904166);
INSERT INTO MMcontain VALUES (4, 111, 904166);
INSERT INTO MMcontain VALUES (1, 22, 469738);
INSERT INTO MMcontain VALUES (2, 22, 469738);
```

```sql
INSERT INTO MMcontain VALUES (3, 22, 469738);
INSERT INTO MMcontain VALUES (4, 22, 469738);
INSERT INTO MMcontain VALUES (5, 22, 469738);
INSERT INTO MMcontain VALUES (4, 33, 279183);
INSERT INTO MMcontain VALUES (4, 333, 279183);
INSERT INTO MMcontain VALUES (1, 44, 628591);
```

MMContain

MovieID	ListID	AccountNumber
1	11	904166
3	11	904166
3	111	904166
4	111	904166
1	22	469738
2	22	469738
3	22	469738
4	22	469738
5	22	469738
4	33	279183
4	333	279183
1	44	628591

```sql
INSERT INTO RReview VALUES (0, 0, 'Good 0 ...', '2021-1-1', 8, 100, 1, 904166);
INSERT INTO RReview VALUES (0, 0, 'Good 1 ...', '2021-1-2', 9, 101, 1, 469738);
INSERT INTO RReview VALUES (0, 0, 'Bad 00 ...', '2021-1-1', 3, 102, 1, 628591);
INSERT INTO RReview VALUES (0, 0, 'Good 2 ...', '2021-1-4', 7, 103, 4, 279183);
INSERT INTO RReview VALUES (0, 0, 'Bad 01 ...', '2021-1-6', 4, 104, 3, 469738);
```

RReview

#Like	#DisLike	Content	Date	Rating	ReviewID	MovieID	AccountNumber
0	0	Good 0 ...	'2021-1-1'	8	100	1	904166
0	0	Good 1 ...	'2021-1-2'	9	101	1	469738
0	0	Bad 00 ...	'2021-1-1'	3	102	1	628591
0	0	Good 2 ...	'2021-1-4'	7	103	4	279183
0	0	Bad 01 ...	'2021-1-6'	4	104	3	469738

```
INSERT INTO CComment VALUES (0, 0, 'Agree 0', '2021-2-1', 1000, 100, 469738);
INSERT INTO CComment VALUES (0, 0, 'Agree 1', '2021-2-2', 1001, 100, 904166);
INSERT INTO CComment VALUES (0, 0, 'Noooooo', '2021-2-1', 1002, 102, 692630);
INSERT INTO CComment VALUES (0, 0, 'ABCDEFG', '2021-1-9', 1003, 103, 628591);
INSERT INTO CComment VALUES (0, 0, 'NOPEEEE', '2021-1-8', 1004, 103, 279183);
```

CComment

#Like	#DisLike	Content	Date	CommentID	ReviewID	AccountNumber
0	0	Agree 0	'2021-2-1'	1000	100	469738
0	0	Agree 1	'2021-2-2'	1001	100	904166
0	0	Noooooo	'2021-2-1'	1002	102	692630
0	0	ABCDEFG	'2021-1-9'	1003	103	628591
0	0	NOPEEEE	'2021-1-8'	1003	103	279183

```
INSERT INTO DiscussionGroup VALUES (1, 'Super Group 0');
INSERT INTO DiscussionGroup VALUES (2, 'Super Group 1');
INSERT INTO DiscussionGroup VALUES (3, 'Super Group 2');
INSERT INTO DiscussionGroup VALUES (4, 'Super Group 3');
INSERT INTO DiscussionGroup VALUES (5, 'Super Group 4');
```

DiscussionGroup

GroupID	GroupName
1	Super Group 0
2	Super Group 1
3	Super Group 2
4	Super Group 3
5	Super Group 4

```sql
INSERT INTO DiscussionContent
VALUES (5000, 1, 279183, 'I like this movie', '2021-2-2');
INSERT INTO DiscussionContent
VALUES (5001, 1, 904166, 'aaaaaaaa', '2021-2-2');
INSERT INTO DiscussionContent
VALUES (5002, 3, 279183, 'hhhhhhhhhhhh', '2021-1-30');
INSERT INTO DiscussionContent
VALUES (5003, 4, 692630, 'GOOD GOOD GOOD', '2021-2-5');
INSERT INTO DiscussionContent
VALUES (5004, 4, 692630, 'tgf34gt34wftgwr3t', '2021-2-8');
```

DiscussionContent

ContentID	GroupID	AccountNumber	Content	Date
5000	1	279183	I like this movie	'2021-2-2'
5001	1	904166	aaaaaaaa	'2021-2-2'
5002	3	279183	hhhhhhhhhhhh	'2021-1-30'
5003	4	692630	GOOD GOOD GOOD	'2021-2-5'
5004	4	692630	tgf34gt34wftgwr3t	'2021-2-8'

Part 8: Queries

Insertion: Add a movie to the HistoryList.

Deletion: Delete a comment from CComment.

Update: Update a user's password.

Selection: Select movies that are produced in 2021.

Projection: Project only the titles of the movies.

Join: Users want to check if a specific movie is in their favourite lists, need to Join FavouriteList and MovieList and MMcontain.

Division: Find all user's favourite lists that have all movies rating larger than 9.5.