COMP 3005 Assignment 4

P1.1

MovieTitle, Director -> Producer
MovieTitle, Director ->MovieOf_SPTitle, MovieOf_Author
SPTitle, Author, SceneNo -> StoryLoc, FilmLoc
SPTitle, Author, SceneNo, TakeNo -> Date, Time
ActorName -> Phone, Address, AgentName
SPTitle, Author, SceneNo, ActorName -> temp1
SPTitle, Author -> temp2

P1.2
MovieTitle, Director -> Producer
MovieTitle, Director -> MovieOf_SPTitle, MovieOf_Author
SPTitle, Author, SceneNo -> StoryLoc, FilmLoc
SPTitle, Author, SceneNo, TakeNo -> Date, Time
ActorName -> Phone, Address, AgentName
SPTitle, Author, SceneNo, ActorName -> temp1
SPTitle, Author -> temp2

P1.3

Movie			
MovieTitle	Director	Producer	

MovieTitle, Director -> Producer

ScreenPlay	
SPTitle	Author

SPTitle, Author -> temp2

MovieOf			
MovieTitle	Director	SPTitle	Author

MovieTitle, Director ->MovieOf_SPTitle, MovieOf_Author

Actor			
ActorName	Phone	Address	AgentName

ActorName -> Phone, Address, AgentName

Scenes				
SPTitle	Author	SceneNo	StoryLoc	FilmLoc

SPTitle, Author, SceneNo -> StoryLoc, FilmLoc

Takes					
SPTitle	Author	SceneNo	TakeNo	Date	Time

SPTitle, Author, SceneNo, TakeNo -> Date, Time

Appears_In			
SPTitle	Author	SceneNo	ActorName

SPTitle, Author, SceneNo, ActorName -> temp

P2.1
BookCode -> BookTitle, Publisher, Date
SongID -> SongTitle, Composer
BookCode, SongID -> AppearsIn_length, AppearsIn_page
UserID -> Password, Name, EmailAddress
BookCode, UserID -> Owns_pdfFileName, Owns_pdfOffset

P2.2
BookCode -> BookTitle, Publisher, Date
SongID -> SongTitle, Composer
BookCode, SongID -> AppearsIn_length, AppearsIn_page
UserID -> Password, Name, EmailAddress
BookCode, UserID -> Owns_pdfFileName, Owns_pdfOffset

P2.3

BookCodes			
BookCode	BookTitle	Publisher	Date

BookCode -> BookTitle, Publisher, Date

Songs		
SongID	SongTitle	Composer

SongID -> SongTitle, Composer

Users			
UserID	Password	Name	EmailAddress

UserID -> Password, Name, EmailAddress

Appears_In			
BookCode	SongID	Length	Page

BookCode, SongID -> AppearsIn_length, AppearsIn_page

Owns			
BookCode	UserID	pdfFileName	pdfOffset

BookCode, UserID -> Owns_pdfFileName, Owns_pdfOffset

P3.1
stdnum -> stdname, email, city, postcode, strnum, street, area_code, office_code, station_code
postcode -> city
area_code, office_code -> city
period, room_num, term -> course_section
course_section -> term
course_num -> department_name, course_name
room_num -> building
course_num, course_section, stdnum -> temp
course_num, stdnum -> grade

P3.2
stdnum -> stdname, email, postcode, strnum, street, area_code, office_code, station_code
postcode -> city
area_code, office_code -> city
period, room_num, term -> course_section
course_section -> term
course_num -> department_name, course_name
room_num -> building
course_num, course_section, stdnum- > temp
course_num, stdnum -> grade

P3.3

Students									
stdnum	stdname	email	postcode	strnum	street	area_code	office_code	station_code	city

stdnum -> stdname, email, postcode, street, area_code, office_code, station_code
postcode -> city
area_code, office_code -> city

Offerings					
period	room_num	term	building	course_section	

period, room_num, term -> course_section
room_num -> building

Section Terms	
course_section	term

course_section -> term

Courses			
course_num	course_name	department_name	

course_num -> department_name, course_name

Enrollment			
course_num	course_section	stdnum	

course_num, course_section, stdnum- > temp

Grades			
course_num	stdnum	grade	

course_num, stdnum -> grade

P4.1

Attributes	Comments
Uid	Unique identifier for a user
Password	User password
UserName	Name of user
AuthorName	Name of an author
GenreName	Name of an genre
ISBN	Unique identifier for a book
BookTitle	Title of a book
Description	Description (summary) of a book
Rating	Rating for a book review
ReviewText	Content of the review
RecommendationStrength	The strength rating (1-5) of a book recommendation to a particular user

Note: if you are using my assignment 2 schema as reference notice that I made a few minor changes, such as removing gid, as genre names should be unique in themselves, so the tables will be slightly different.

P4.2
Assumptions:

- Users cannot write multiple reviews of the same book
- Not all books will have been read by a user, and users will not necessarily have read any books
- Not all books will appear in recommendations and users won't get recommendations until they enter some data
- Not all authors will be somebody's favourite and not all users will have favourites
- Not all users will specify genre interests and a genre will not necessarily be interesting to any users
- Books will not have multiple authors

Scenario:

1. Store data (title, cover, description, author) for many books in a variety of genres.
2. Allow the user to create a list of books they have read as well as an optional rating based on how much they enjoyed each book (ratings will be editable).
3. Allow users to create a list of their favourite authors.
4. Provide a list of reviews for any particular book and allow users to write their own reviews of books that they choose.
5. The database will allow users to search for books in many ways including: by topic or genre, by title, by author, or by ISBN.
6. Users will be able to easily add to and edit their "reading list" as well as maintain a list of topics they are interested in.
7. The application should provide users a list of recommendations that will be automatically generated and updated based on their personal data.
8. Recommendations will be based off many factors including: books related to the user's topics of interest, books by similar authors, books read by others with similar interests etc.
9. The target size of the database would be upwards of 10, 000 books.
10. The application hosting should be web based and give users a to simple interface to browse and easily view and edit their data using any popular browser (chrome, safari, firefox etc.).

P4.3
Uid -> UserName, Password
ISBN -> BookTitle, Description
ISBN -> AuthorName
ISBN -> GenreName
Uid, ISBN -> Rating, ReviewText
Uid, GenreName -> temp1
Uid, AuthorName -> temp2

Uid, ISBN -> RecommendationStrength
Uid, ISBN -> temp3

P4.4
Uid -> UserName, Password
ISBN -> BookTitle, Description, AuthorName, GenreName
Uid, ISBN -> Rating, ReviewText
Uid, GenreName -> temp1
Uid, AuthorName -> temp2
Uid, ISBN -> RecommendationStrength
Uid, ISBN -> temp3

P4.5

Books				
ISBN	BookTitle	AuthorName	Description	Genre

ISBN -> BookTitle, Description, AuthorName, GenreName

Users		
Uid	Password	UserName

Uid -> UserName, Password

Reviews			
ISBN	Uid	Rating	ReviewText

Uid, ISBN -> Rating, ReviewText

FavAuthors	
Uid	AuthorName

Uid, AuthorName -> temp2

Recommended		
Uid	ISBN	Strength

Uid, ISBN -> RecommendationStrength

Read	
Uid	ISBN

Uid, ISBN -> temp3

Interests	
Uid	GenreName

Uid, GenreName -> temp1