SQL Assignment

In this assignment, you'll have to come up with SQL queries for the following database schema:

Artists (artistID: int, name: varchar(255))
SimilarArtists (artistID: int, simArtistID: int, weight: int)
Albums (albumID: int, artistID: int, name: varchar(255))
Tracks (trackID: int, artistID: int, name: varchar(255), length: int)
TrackLists (albumID: int, trackID: int, trackNum: int)

All primary keys are underlined. All foreign keys have the same name as the primary key that they are referencing. When asking about the similarity of one Artist to another, you can safely assume that the pair of Artists will only appear in one tuple in the SimilarArtists table. Please write SQL statements for the following ten queries:

1. Find the names of all Tracks that are more than 10 minutes (600,000 ms) long.
Result: (**name**: varchar(255))
2. Find the names of all Artists who have recorded a self-titled Album (the name of the Album is the same as the name of the Artist).
Result: (**name**: varchar(255))
3. Find the names of all Artists who have recorded an Album on which the first track is named "Intro".
Result: (**name**: varchar(255))
4. Find the names of all Artists who are more similar to Mogwai than to Nirvana (meaning the weight of their similarity to Mogwai is greater).
Result: (**name**: varchar(255))
5. Find the names of all Albums that have more than 30 tracks.
Result: (**name**: varchar(255))
6. Find the names of all Artists who do not have a similarity rating greater than 5 to any other Artist.

Result: (**name**: varchar(255))

7. For all Albums, list the Album's name and the name of its 15th Track. If the Album does not have a 15th Track, list the Track name as null.

Result: (**album_name**: varchar(255), **track_name**: varchar(255))

8. List the name of each Artist, along with the name and average Track length of their Album that has the highest average Track length.

Result: (**artist_name**: varchar(255), **album_name**: varchar(255), **avg_track_length**: float)

9. For all Artists that have released a Track with a name beginning with "The", give their name and the name of the Artist who is most similar to them that has released a Track with a name beginning with "Why". If there is a tie, choose the Artist with the lowest artistID. If there are no qualifying Artists, list the Artist name as null.

Result: (**artist_name_the**: varchar(255), **artist_name_why**: varchar(255))

10. If an Artist is within one degree of Hot Water Music, that means that their similarity to Hot Water Music is at least 5. If an Artist is within N degrees of Hot Water Music, then they have a similarity of at least 5 to an Artist that is within N-1 degrees of Hot Water Music. Find the percent of all Artists that are within 6 (or fewer) degrees of Hot Water Music.

(Your answer should also include Hot Water Music themselves! Also, note that the percentage should be an integer between 0 and 100, inclusive.)

Result: (**percentage**: int)