

American International University-Bangladesh Department of Computer Science and Engineering Dhaka, Bangladesh

$00892~{\rm ADVANCE~DATABASE~MANAGEMENT~SYSTEM}$

PROJECT Report

[B][SUMMER 22-23]

E-Sports Management System

Submitted by

Names of Students	ID
AJRAN HOSSAIN	19-39334-1
SHARIF HADI MAHATAB	20-43625-2
MD. SARAFAT ALI ADIR	20-41926-1
MEDHA CHOWDHURY	20-41930-1

Date of Submission

July 14, 2023

Submitted to JUENA AHMED NOSHIN

Assistant Professor, Faculty
Department of Computer Science and Engineering
American International University-Bangladesh

Contribution

	AJRAN HOSSAIN	SHARIF HADI MAHATAB	MD. SARAFAT ALI ADIR	MEDHA CHOWDHURY	Contribution (%)
	19-39334-1	20-43625-2	20-41926-1	20-41930-1	
Diagram	40%	20%	0%	0%	60(%)
UI Design	100%	0%	0%	0%	100(%)
Normalization	50%	50%	0%	0%	100(%)
SQL Query	70%	15%	15%	15%	100(%)
Relational Algebra	15%	0%	85%	0%	100(%)
Report Writing	50%	30%	10%	10%	100(%)

Contents

1	Intr	oduction 1				
	1.1	Project Proposal				
		1.1.1 Purposes				
		1.1.2 Methodology				
	1.2	Project Scenario				
2 Diagrams						
	2.1	ER Diagram				
	2.2	Class Diagram				
	2.3	Use Case Diagram				
	2.4	Activity Diagram				
3	Use	r Interface 8				
	3.1	Technologies Used				
		3.1.1 Home Page				
		3.1.2 About Page				
		3.1.3 Login Page				
		3.1.4 Profile Page				
		3.1.5 Tournament Page				
		3.1.6 Team Page				
		3.1.7 Game Page				
4	NF	and Schema 11				
	4.1	Manage branch				
		4.1.1 (Admin \rightarrow Manager)				
		4.1.2 (Manager \rightarrow Finance)				
		$4.1.3$ (Manager \rightarrow Teams)				
		$4.1.4$ (SocialMedia \rightarrow ContentCreator)				
	4.2	Pay branch				
		4.2.1 (Finance \rightarrow SocialMedia)				
		4.2.2 (Finance \rightarrow organization)				
	4.3	Formed branch				
		$4.3.1$ (Teams \rightarrow Player)				
	4.4	Has branch				
		$4.4.1$ (Record \rightarrow Tournament)				
		$4.4.2$ (Tournament \rightarrow Game)				
	4.5	Participate branch				
		$4.5.1 (\text{Teams} \rightarrow \text{Game}) \dots \dots \dots \dots \dots \dots \dots \dots \dots $				
	4.6	Host branch				
	1.0	$4.6.1$ (Organizer \rightarrow Tournament)				
	4.7	Sponsor branch				
	1.1	4.7.1 (Companies \rightarrow Organization)				
		4.7.2 (Companies \rightarrow Teams)				
	4.8	Temporary Tables				
	4.9	Final Tables				
		Schema Diagram				
	1.10	~~::::::::::::::::::::::::::::::::::::				

5	SQI	L Queries	31
	5.1	Table Creation	31
	5.2	Sequence Creation	44
	5.3	Index for Table	
	5.4	Alter Table	46
	5.5	Data Insertion	47
	5.6	Single Row Functions	58
	5.7	Group Functions	59
	5.8	SubQuery	60
	5.9	Join Queries	62
	5.10	Creating View	64
	5.11	Synonyms	65
6	Rela	ational Algebra	66
7	Con	nclusion	67

Introduction

Esports Management System is an innovative platform that will revolutionize the management and organization of esports teams, participants, tournaments, and sponsors. This system seeks to provide users with an efficient and user-friendly way to search for their preferred professional esports players.

A user-friendly interface is at the core of the Esports Management System, allowing users to seamlessly navigate and explore the realm of professional esports. With only a few clicks, users can search for potential professional athletes and teams, as well as access valuable information such as their winning records and accomplishments. This enables fans and enthusiasts to remain up-to-date on their preferred players and teams, nurturing a stronger connection within the esports community.

The Esports Management System's ability to facilitate sponsorships is a crucial feature. Numerous organizations and businesses can engage in sponsorship activities, whether for the purpose of supporting tournaments or individual athletes. The system serves as a centralized repository where the information and details of these sponsors can be efficiently stored and managed. This facilitates the sponsorship process and ensures that sponsors and the esports industry collaborate effectively.

There are specialized administrators within the Esports Management System who play crucial roles in managing and enhancing the overall experience. The social media manager is among these supervisors; he or she supervises the organization's online presence and engagement on various social media platforms. In addition, the content creator/VFX/GFX team assures the creation of visually stunning and captivating content that enhances the overall esports experience.

Dynamic features and functionalities make the Esports Management System an indispensable instrument for the esports industry. It makes it easier for fans to discover and connect with professional esports players, allowing them to remain informed and engaged. It enhances collaboration between organizations and the esports community by providing a centralized platform for sponsorship management. In addition, the system enables administrators to enhance the organization's online presence and develop visually appealing content, ensuring that all stakeholders have an engaging experience.

In the following sections, we will delve deeper into the features, functionalities, and innovative aspects of the Esports Management System, demonstrating its potential to revolutionize the management and celebration of esports teams, players, tournaments, and sponsors.

1.1 Project Proposal

This proposal for the development and implementation of an Esports Management System is presented with pleasure. This revolutionary platform seeks to transform the management and organization of esports teams, players, tournaments, and sponsors. The Esports Management System will improve the user experience, encourage community engagement, and expedite operations within the esports industry by leveraging advanced technology and comprehensive functionalities.

1.1.1 Purposes

- Create an intuitive web-based platform that serves as the central hub for esports administration, catering to the requirements of teams, players, tournament organizers, and sponsors.
- Implement a sophisticated matching algorithm to facilitate the search and discovery of favored professional esports players, thereby enhancing the fan experience and fostering esports community connections.
- Provide efficient sponsorship administration capabilities, enabling organizations and businesses to support tournaments or individual athletes through sponsorship activities.
- Enhance the organization's online presence by supervising social media platforms and having the content creator/VFX/GFX team produce visually spectacular and engaging content.

1.1.2 Methodology

System Development:

- Conduct exhaustive investigation on the necessary requirements and features of an effective Esports Management System.
- Utilize industry-standard programming languages and technologies to create a scalable and secure web-based platform.
- Implement a user-friendly interface with intuitive navigation in order to provide a seamless and enjoyable user experience.

Matching Algorithm

- Collaboration with data scientists and psychologists to create a matching algorithm based on personality traits, values, and beliefs.
- Integrate the matching algorithm into the system to recommend professional esports players compatible with the user's preferences.

Sponsorship Management

- Create an all-encompassing sponsorship management module to facilitate collaborations between organizations and the esports industry.
- Provide a centralized repository for sponsor information to facilitate communication and sponsorship efficiency.

Online Presence Enhancement

- Appoint a social media manager to supervise the organization's online presence and interact with the esports community.
- Appoint a social media manager to supervise the organization's online presence and interact with the esports community.

1.2 Project Scenario

Imagine an eSports organization called "eSports FTW" that manages various teams and tournaments in the gaming industry. The organization is led by an Admin who oversees the operations. The Admin entity contains attributes such as email, picture, password, name, and a unique ID.

Under the Admin, there are multiple Managers responsible for different departments. Each Manager has attributes including hire date, picture, salary, email, name, department ID, and a unique ID. One Manager specifically manages the Finance department, ensuring financial stability and handling the accounts for the organization. The Finance entity consists of attributes such as a unique ID, account number, and balance.

In addition to managing the finances, each Manager is in charge of a specific team. The Teams entity includes details such as the team's established date, country, name, team ID, team icon, winning numbers, and total prize money. Each team has a unique Manager assigned to it, ensuring proper coordination and organization. The Manager entity is linked to the Teams entity through the Manager ID attribute.

Within each team, there are multiple Players who represent the organization in various games. The Players entity contains attributes such as name, player ID, picture, salary, winning prize money, total hours played, phone number, and address (including country, city, zip code, and road number). Additionally, players have links to their social media profiles on platforms like Facebook, Instagram, Twitter, and YouTube.

The organization hosts tournaments, bringing together teams from different games. The Tournament entity consists of attributes such as tournament ID, name, prize pool, starting date, and ending date. Each tournament features various games such as Valorant, Mobile Legends: Bang Bang (MLBB), and Rainbow Six Siege. The Game entity contains attributes like name, release date, genre, game picture, publisher, platform, game ID, and prize pool. Each tournament may have different games associated with it, creating diverse competitive environments.

To support the teams and tournaments financially, eSports FTW seeks sponsorships from different companies. The Companies entity includes attributes such as name, company ID, location, sponsor date, and phone number. Multiple companies can sponsor both teams and tournaments, forming a many-to-many relationship between the Companies, Teams, and Tournament entities.

Additionally, eSports FTW employs a Social Media Manager responsible for managing the organization's online presence. The Social Media Manager entity contains attributes such as name, picture, email, manager ID, hire date, salary, phone number, and social media links (Facebook, Instagram, Twitter, YouTube). The Social Media Manager oversees the VFX/GFX and Content Creator teams, ensuring engaging content creation and visual effects. The VFX/GFX and Content Creator entities include attributes like name, picture, VFX/GFX ID, email, phone number, salary, and address (country, city, zip code, and road number).

Diagrams

2.1 ER Diagram



ER Diagram

2.2 Class Diagram



Class Diagram

2.3 Use Case Diagram



Use Case Diagram

2.4 Activity Diagram



Activity Diagram

User Interface

3.1 Technologies Used

- SvelteKit
- Tailwind CSS

3.1.1 Home Page



3.1.2 About Page



3.1.3 Login Page



3.1.4 Profile Page



3.1.5 Tournament Page



3.1.6 Team Page



3.1.7 Game Page



Normalization and Schema Design

4.1 Manage branch

$4.1.1 \quad (Admin \rightarrow Manager)$

UNF

(<u>Admin_ID</u>, Admin_Name, Admin_Email, Admin_Password, Admin_Picture, <u>Manager_ID</u>, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate, Manager_Phone)

1NF

Phone is multi-valued attribute.

• (<u>Admin_ID</u>, Admin_Name, Admin_Email, Admin_Password, Admin_Picture, <u>Manager_ID</u>, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, <u>Manager_Hiredate</u>, Manager_Phone)

2NF

- Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture
- Manager_ID, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate, Manager_Phone

3NF

No transitive dependencies found. Same as 2NF

- Manager (Manager_ID, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate, Admin_ID)
- Admin (Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture)
- Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)

4.1.2 (Manager \rightarrow Finance)

UNF

(<u>Manager_ID</u>, Manager_Name, Manager_Email, Manager_Password,Manager_Picture, Manager_Hiredate, Manager_Phone, Finance_ID, Finance_Account_Number, Finance_Balance)

1NF

Phone is multi-valued attribute.

• (Manager_ID, Manager_Name, Manager_Email, Manager_Password,Manager_Picture, Manager_Hiredate, Manager_Phone, Finance_ID, Finance_Account_Number, Finance_Balance)

2NF

- Manager_ID, Manager_Name, Manager_Email, Manager_Password,Manager_Picture, Manager_Hiredate, Manager_Phone)
- Finance(Finance_ID, Finance_Account_Number, Finance_Balance)

3NF

No transitive dependency found. Same as 2NF.

- Manager_Manager_ID, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate)
- Finance(Finance_ID, Finance_Account_Number, Finance_Balance, Manager_ID)
- Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)

4.1.3 (Manager \rightarrow Teams)

UNF

(<u>Manager_ID</u>, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate, Manager_Phone, <u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Price_Money, Team_Winnig)

1NF

Winning & Phone are multi-valued attribute.

• (Manager_ID, Manager_Name, Manager_Email, Manager_Password,Manager_Picture, Manager_Hiredate, Manager_Phone, <u>Team_ID</u>, Team_Name,Team_Icon, Team_established_date, Team_country,Team_Total_Price_Money,Team_Winnig)

2NF

- Manager (Manager ID, Manager Name, Manager Email, Manager Password, Manager Picture, Manager Hiredate, Manager Phone)
- Team(Team_ID, Team_Name, Team_Icon, Team_established_date, Total_Price_Money, Team_country

3NF

No transitive dependency found. Same as 2NF.

- Manager (Manager ID, Manager Name, Manager Email, Manager Password, Manager Picture, Manager Hiredate)
- **Team**(<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Price_Money, **Manager_ID**
- Team_Winnig(Tw_ID, Team_ID, Team_Winnig)
- Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)

4.1.4 (SocialMedia \rightarrow ContentCreator)

UNF

(<u>SocialMedia_ID</u>, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Phone, SocialMedia_Hiredate, SocialMedia_Salary, ContentCreator_ID, ContentCreator_Name, ContentCreator_Email, ContentCreator_Password, ContentCreator_Picture, ContentCreator_Phone, ContentCreator_Hiredate, ContentCreator_Salary, ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_Instagram_Link, ContentCreator_Youtube_Link, ContentCreator_Country, ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)

1NF

Phone is multi-valued attribute.

• (<u>SocialMedia_ID</u>, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Phone, SocialMedia_Hiredate, SocialMedia_Salary, <u>ContentCreator_ID</u>, ContentCreator_Name, ContentCreator_Email, ContentCreator_Password, ContentCreator_Picture, ContentCreator_Phone, ContentCreator_Hiredate, ContentCreator_Salary, ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_ator_

 $Instagram_Link, ContentCreator_Youtube_Link \,,\, ContentCreator_Country, ContentCreator_City, \, ContentCreator_Street, \, ContentCreator_Zip_Code)$

2NF

- SocialMedia (SocialMedia_ID, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Phone, SocialMedia_Hiredate, SocialMedia_Salary)
- ContentCreator (ContentCreator_ID, ContentCreator_Name, ContentCreator_Email, ContentCreator_Password, ContentCreator_Picture, ContentCreator_Phone, ContentCreator_Hiredate, ContentCreator_Salary, ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_Unstagram_Link_ContentCreator_Voutube_Link_ContentCreator_Country_Country_ContentCreator_Country_Countr

Instagram_Link, ContentCreator_Youtube_Link, ContentCreator_Country, ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)

3NF

- SocialMedia <u>SocialMedia_ID</u>, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary)
- ContentCreator <u>ContentCreator_ID</u>, ContentCreator_Name, ContentCreator_Email, ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate, ContentCreator_Salary)
- ContentCreator_SocialMedia (ContentCreator_ID, SocialMedia_ID, ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_Instagram_Link, ContentCreator_Youtube_Link)
- ContentCreator_Address (ContentCreator_ID, ContentCreator_Country, ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)

- SocialMedia (SocialMedia_ID, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary)
- ContentCreator (<u>ContentCreator_ID</u>, ContentCreator_Name, ContentCreator_Email, ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate, ContentCreator_Salary)
- ContentCreator_SocialMedia (<u>Ccs_ID</u>,ContentCreator_ID, ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_ Instagram_Link, ContentCreator_Youtube_Link)
- ContentCreator_Address (<u>Cca_ID</u>, ContentCreator_ID, ContentCreator_Country, ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)
- ContentCreator_Phone (Ccp_ID, ContentCreator_ID, ContentCreator_Phone)
- SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)
- ContentCreator_SocialMedia_Phone (<u>Ccp_ID</u>, ContentCreator_ID, ContentCreator_Phone)

4.2 Pay branch

4.2.1 (Finance \rightarrow SocialMedia)

\mathbf{UNF}

(<u>Finance_ID</u>, Finance_Account_Number, Finance_Balance, <u>SocialMedia_ID</u>, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Phone, SocialMedia_Hiredate, SocialMedia_Salary)

1NF

Phone is multi-valued attribute.

• (<u>Finance_ID</u>, Finance_Account_Number, Finance_Balance, <u>SocialMedia_ID</u>, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Phone, SocialMedia_Hiredate, SocialMedia_Salary)

2NF

- Finance(Finance_ID, Finance_Account_Number, Finance_Balance)
- SocialMedia(SocialMedia_ID, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Phone, SocialMedia_Hiredate, SocialMedia_Salary)

3NF

No transitive dependency found. Same as 2NF.

- Finance(Finance_ID, Finance_Account_Number, Finance_Balance)
- SocialMedia(SocialMedia_ID, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary, Finance_ID)
- SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)

4.2.2 (Finance \rightarrow organization)

UNF

(<u>Finance_ID</u>, Finance_Account_Number, Finance_Balance, Organization_ID, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone)

1NF

Phone is multi-valued attribute.

• (<u>Finance_ID</u>, Finance_Account_Number, Finance_Balance, <u>Organization_ID</u>, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone)

2NF

- Finance(Finance_ID, Finance_Account_Number, Finance_Balance)
- Organization(Organization_ID, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone)

3NF

No transitive dependency found. Same as 2NF.

- Finance(Finance_ID, Finance_Account_Number, Finance_Balance)
- Organization(Organization_ID, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone, Finance_ID)
- Organization_Phone (Op_ID, Organization_ID,Organization_Phone)

4.3 Formed branch

$4.3.1 \quad ({ m Teams} ightarrow { m Player})$

\mathbf{UNF}

(<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Earning,

Team_Winnig, <u>Player_ID</u>, <u>Player_Name</u>, <u>Player_Email</u>, <u>Player_Password</u>, <u>Player_Picture</u>, <u>Player_JoinDate</u> <u>Player_Phone</u>, <u>Player_Salary</u>, <u>Player_Winnig_Money</u>, <u>Player_Play_Hours</u>, <u>Player_country</u>, <u>Player_City</u>, <u>Player_Street</u>, <u>Player_Zip_Code</u>, <u>Player_DOB</u>, <u>Player_Facebook_Link</u>, <u>Player_Instagram_Link</u>, <u>Player_Twitter_Link</u>, <u>Player_Youtube_Link</u>)

1NF

Phone & Wining number are multi-valued attribute.

• (<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Earning, Team_Winnig, <u>Player_ID</u>, Player_Name, Player_Email, Player_Password, Player_Picture, Player_JoinDate, <u>Player_Phone</u>, Player_Salary, Player_Winnig_Money, Player_Play_Hours, Player_country, Player_City, Player_Street, Player_Zip_Code, Player_DOB, Player_Facebook_Link, Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)

2NF

- Team(<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Earning, Team_Winnig)
- Player_Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture, Player_JoinDate, Player_Salary, Player_Winnig_Money, Player_Play_Hours, Player_country, Player_City, Player_Street, Player_Zip_Code, Player_DOB, Player_Facebook_Link, Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)

3NF

- **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>)
- Player Player Name, Player Email, Player Password, Player Picture, Player JoinDate, Player Salary, Player Winnig Money, Player Play Hours, Player DOB)
- $\bullet \ \ Player_Address \ (\underline{Pa_ID}, Player_ID, Player_country, Player_City, Player_Street, Player_Zip_Code)$
- Player_Social_Link (<u>Psl_ID</u>, Player_ID, Player_Facebook_Link, Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)

- **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>)
- Player(Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture, Player_JoinDate, Player_Salary, Player_Play_Hours, Player_DOB)

- Player_Address (Pa_ID, Player_ID, Player_country, Player_City, Player_Street, Player_Zip_Code)
- Player_Social_Link (<u>Psl_ID</u>, Player_ID, Player_Facebook_Link, Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)
- Player_Phone (Pp_ID, Player_ID, Player_Phone)
- Player_Wining (Pw_ID, Player_ID, Player_Winnig)
- $\bullet \ \, \mathbf{Player_Team} \ (\underline{\mathrm{Pt_ID}}, \, \mathbf{Player_ID}, \, \mathbf{Team_ID})$

4.4 Has branch

$\textbf{4.4.1} \quad (\textbf{Record} \rightarrow \textbf{Tournament})$

\mathbf{UNF}

(<u>Record_ID</u>, Record_Date, Record_Price_Pool, <u>Tournament_ID</u>, Tournament_Name, Tournament_StartingDate, Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool)

1NF

• (<u>Record_ID</u>, Record_Date, Record_Price_Pool, <u>Tournament_ID</u>, Tournament_Name, Tournament_StartingDate, Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool)

2NF

- Record(Record_ID, Record_Date, Record_Price_Pool, Tournament_ID)
- Tournament_ID, Tournament_Name, Tournament_StartingDate,Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool)

3NF

No transitive dependency found. Same as 2NF.

- Record(Record_ID, Record_Date, Record_Price_Pool, Tournament_ID)
- Tournament_ID, Tournament_Name, Tournament_StartingDate,Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool)

4.4.2 (Tournament \rightarrow Game)

UNF

(<u>Tournament_ID</u>, Tournament_Name, Tournament_StartingDate, Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool, <u>Game_ID</u>, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)

1NF

• (<u>Tournament_ID</u>, Tournament_Name, Tournament_StartingDate, Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool, <u>Game_ID</u>, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)

2NF

- Tournament_ID, Tournament_Name, Tournament_StartingDate,Tournament_Ending Tournament_Location, Tournament_Prize_Pool)
- Game(<u>Game_ID</u>, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)

3NF

No transitive dependency found. Same as 2NF.

- **Tournament**(<u>Tournament_ID</u>, Tournament_Name, Tournament_StartingDate,Tournament_Ending Tournament_Location, Tournament_Prize_Pool)
- Game(Game_ID, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)
- Tournament_Game(Tournament_ID, Game_ID)

4.5 Participate branch

$4.5.1 \quad (\text{Teams} \rightarrow \text{Game})$

$\overline{\text{UNF}}$

(<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Earning, Team_Winnig, <u>Game_ID</u>, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool Game_Genre, Game_Publisher))

1NF

• (<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Earning, Team_Winnig, <u>Game_ID</u>, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)

2NF

- **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>)
- Game(Game_ID, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)

3NF

No transitive dependency found. Same as 2NF.

- **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>)
- Game(<u>Game_ID</u>, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)
- Team_Game(Team_ID, Game_ID)

4.6 Host branch

4.6.1 (Organizer \rightarrow Tournament)

UNF

(<u>Tournament_ID</u>, Tournament_Name, Tournament_StartingDate, Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool, <u>Organization_ID</u>, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone)

1NF

Phone is a multi-vale attribute.

• (<u>Tournament_ID</u>, Tournament_Name, Tournament_StartingDate,Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool, <u>Organization_ID</u>, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone)

2NF

- Tournament_ID, Tournament_Name, Tournament_StartingDate,Tournament_Ending Tournament_Location, Tournament_Prize_Pool, Organization_ID)
- Organization (Organization_ID, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone)

3NF

No transitive dependency found. Same as 2NF.

- Tournament_ID, Tournament_Name, Tournament_StartingDate,Tournament_Ending Tournament_Location, Tournament_Prize_Pool, Organization_ID)
- Organization (Organization_ID, Organization_Name, Organization_Email, Organization_Password, Organization_Picture)
- Organization_Phone(Organization_ID, Organization_Phone)
- Organization_Tournament(Organization_ID, Tournament_ID)

4.7 Sponsor branch

4.7.1 (Companies \rightarrow Organization)

UNF

(<u>Organization_ID</u>, Organization_Name, Organization_Email, Organization_Picture, Organization_Phone, <u>Company_ID</u>, Company_Name, Company_Email, Company_Picture, Company_Phone, location)

1NF

phone is a multi-value attribute.

• (Organization_ID, Organization_Name, Organization_Email, Organization_Picture, Organization_Phone, Company_ID, Company_Name, Company_Email, Company_Picture, Company_Phone, location)

2NF

- Organization(Organization_ID, Organization_Name, Organization_Email, Organization_Picture, Organization_Phone, Company_ID)
- Company (Company_ID, Company_Name, Company_Email, Company_Picture, Company_Phone, location)

3NF

No transitive dependency found. Same as 2NF.

- Organization(Organization_ID, Organization_Name, Organization_Email, Organization_Picture, Company_ID)
- Organization_Phone(Organization_ID, Organization_Phone)
- Company (Company_ID, Company_Name, Company_Email, Company_Picture, Company_Phone, location)
- Company_Phone(Company_ID, Company_Phone)
- Organization_Company(Organization_ID, Company_ID)

4.7.2 (Companies \rightarrow Teams)

UNF

(<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Earning, Team_Winnig, <u>Company_ID</u>, Company_Name, Company_Email, Company_Password, Company_Picture, <u>Company_Phone</u>, Company_location)

1NF

Phone is multi-value attribute.

• (<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Earning, Team_Winnig, <u>Company_ID</u>, Company_Name, Company_Email, Company_Password, Company_Picture, Company_Phone, Company_location)

2NF

- **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>
- Company (Company ID, Company Name, Company Email, Company Password, Company Picture, Company Phone, Company Location)

3NF

No transitive dependency found. Same as 2NF.

- **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>
- Company (Company ID, Company Name, Company Email, Company Password, Company Picture, Company Phone, Company location)
- Team_Company(<u>Team_ID</u>, Company_ID)
- Company_Phone(Company_ID, Company_Phone)

4.8 Temporary Tables

- 1. **Manager** (Manager_ID, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate, **Admin_ID**)
- 2. Admin (Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture)
- 3. Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
- 4. Manager_Manager_ID, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate)
- 5. Finance(Finance_ID, Finance_Account_Number, Finance_Balance, Manager_ID)
- 6. Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
- 7. Manager_Manager_D, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate)
- 8. **Team**(<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Price_Money, **Manager_ID**
- 9. **Team_Winnig**(<u>Tw_ID</u>, <u>Team_ID</u>, Team_Winnig)
- 10. Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
- 11. **SocialMedia** (<u>SocialMedia_ID</u>, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary)
- 12. **ContentCreator** (<u>ContentCreator_ID</u>, ContentCreator_Name, ContentCreator_Email, ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate, ContentCreator_Salary)
- 13. ContentCreator_SocialMedia (<u>Ccs_ID</u>,ContentCreator_ID, ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_ Instagram_Link, ContentCreator_Youtube_Link)
- 14. ContentCreator_Address (<u>Cca_ID</u>, ContentCreator_ID, ContentCreator_Country, ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)
- 15. ContentCreator_Phone (Ccp_ID, ContentCreator_ID, ContentCreator_Phone)
- 16. SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)
- 17. Finance (Finance ID, Finance Account Number, Finance Balance)
- 18. **Organization**(Organization_ID, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone, **Finance_ID**)
- 19. **Organization_Phone** (Op_ID, **Organization_ID**, Organization_Phone)
- 20. Finance Finance ID, Finance Account Number, Finance Balance)
- 21. SocialMedia_SocialMedia_ID, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary, Finance_ID)
- 22. SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)

- 23. **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>)
- 24. **Player**(Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture, Player_JoinDate, Player_Salary, Player_Play-Hours, Player_DOB)
- 25. Player_Address (Pa_ID, Player_ID, Player_country, Player_City, Player_Street, Player_Zip_Code)
- 26. **Player_Social_Link** (<u>Psl_ID</u>, **Player_ID**, Player_Facebook_Link, Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)
- 27. Player_Phone (Pp_ID, Player_ID, Player_Phone)
- 28. Player_Wining (Pw_ID, Player_ID, Player_Winnig)
- 29. Player_Team (Pt_ID, Player_ID, Team_ID)
- 30. **Record**(Record_ID, Record_Date, Record_Price_Pool, **Tournament_ID**)
- 31. **Tournament**_ID, Tournament_Name, Tournament_StartingDate,Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool)
- 32. **Tournament**_ID, Tournament_Name, Tournament_StartingDate,Tournament_Ending Tournament_Location, Tournament_Prize_Pool)
- 33. **Game**(<u>Game_ID</u>, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)
- 34. Tournament_Game(<u>Tournament_ID</u>, <u>Game_ID</u>)
- 35. **Tournament**_ID, Tournament_Name, Tournament_StartingDate,Tournament_Ending Tournament_Location, Tournament_Prize_Pool, Organization_ID)
- 36. Organization (Organization ID, Organization Name, Organization Email, Organization Password Organization Picture)
- 37. Organization_Phone(Organization_ID, Organization_Phone)
- 38. **Organization_Tournament**(Organization_ID, Tournament_ID)
- 39. **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>)
- 40. Game_Came_ID, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)
- 41. Team_Game(Team_ID, Game_ID)
- 42. Organization (Organization ID, Organization Name, Organization Email, Organization Picture, Company ID)
- 43. Organization_Phone(Organization_ID, Organization_Phone)
- 44. Company (Company ID, Company Name, Company Email, Company Picture, Company Phone, location)

- 45. Company_Phone(Company_ID, Company_Phone)
- 46. **Organization_Company**(Organization_ID, Company_ID)
- 47. **Team**(<u>Team_ID</u>, <u>Team_Name</u>, <u>Team_Icon</u>, <u>Team_established_date</u>, <u>Team_country</u>, <u>Total_Earning</u>, <u>Team_Winnig</u>
- 48. Company_Company_ID, Company_Name, Company_Email, Company_Password, Company_Pietur Company_Phone, Company_location)
- 49. **Team_Company**(<u>Team_ID</u>, Company_ID)
- 50. Company_Phone(Company_ID, Company_Phone)

4.9 Final Tables

- 1. **Manager** (Manager_ID, Manager_Name, Manager_Email, Manager_Password, Manager_Picture, Manager_Hiredate, **Admin_ID**)
- 2. Admin (Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture)
- 3. Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
- 4. Finance(Finance_ID, Finance_Account_Number, Finance_Balance, Manager_ID)
- 5. **Team**(<u>Team_ID</u>, Team_Name, Team_Icon, Team_established_date, Team_country, Total_Price_Money, **Manager_ID**
- 6. **Team_Winnig**(<u>Tw_ID</u>, <u>Team_ID</u>, Team_Winnig)
- 7. **SocialMedia** (<u>SocialMedia_ID</u>, SocialMedia_Name, SocialMedia_Email, SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary)
- 8. **ContentCreator** (<u>ContentCreator_ID</u>, ContentCreator_Name, ContentCreator_Email, ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate, ContentCreator_Salary)
- 9. ContentCreator_SocialMedia (<u>Ccs_ID</u>,ContentCreator_ID, ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_ Instagram_Link, ContentCreator_Youtube_Link)
- 10. ContentCreator_Address (<u>Cca_ID</u>, ContentCreator_ID, ContentCreator_Country, ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)
- 11. ContentCreator_Phone (Ccp_ID, ContentCreator_ID, ContentCreator_Phone)
- 12. SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)

- 13. **Organization**(Organization_ID, Organization_Name, Organization_Email, Organization_Password, Organization_Picture, Organization_Phone, **Finance_ID**)
- 14. **Organization_Phone** (Op_ID, **Organization_ID**, Organization_Phone)
- 15. **Player**(Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture, Player_JoinDate, Player_Salary, Player_Play-Hours, Player_DOB)
- 16. Player_Address (Pa_ID, Player_ID, Player_country, Player_City, Player_Street, Player_Zip_Code)
- 17. **Player_Social_Link** (<u>Psl_ID</u>, **Player_ID**, Player_Facebook_Link, Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)
- 18. Player_Phone (Pp_ID, Player_ID, Player_Phone)
- 19. Player_Wining (Pw_ID, Player_ID, Player_Winnig)
- 20. Player_Team (Pt_ID, Player_ID, Team_ID)
- 21. Record(Record_ID, Record_Date, Record_Price_Pool, Tournament_ID)
- 22. **Tournament**_ID, Tournament_Name, Tournament_StartingDate,Tournament_EndingDate, Tournament_Location, Tournament_Prize_Pool)
- 23. **Game**(<u>Game_ID</u>, Game_Name, Game_Icon, Game_ReleaseDate, Game_Platform, Game_PricePool, Game_Genre, Game_Publisher)
- 24. Tournament_Game(Tournament_ID, Game_ID)
- 25. **Organization_Tournament**(Organization_ID, Tournament_ID)
- 26. **Team_Game**(Team_ID, Game_ID)
- 27. Company_Company_ID, Company_Name, Company_Email, Company_Picture, Company_Phone, location)
- 28. Company_Phone(Company_ID, Company_Phone)
- 29. **Organization_Company**(Organization_ID, Company_ID)
- 30. **Team_Company**(<u>Team_ID</u>, Company_ID)

4.10 Schema Diagram



Schema Diagram

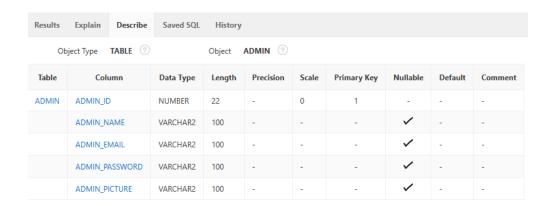
SQL Queries

5.1 Table Creation

```
CREATE TABLE Admin (
Admin_ID INT PRIMARY KEY,
Admin_Name VARCHAR(100),
Admin_Email VARCHAR(100),
Admin_Password VARCHAR(100),
Admin_Picture VARCHAR(100)

);
```

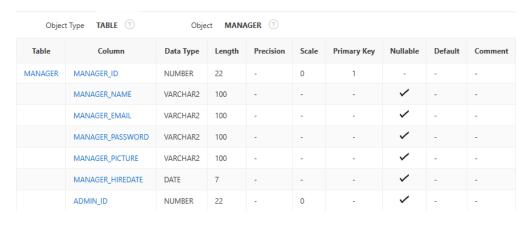
Create Admin table



Admin table description

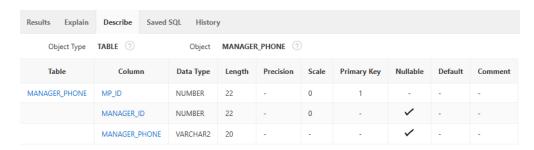
```
CREATE TABLE Manager (
1
2
            Manager_ID INT PRIMARY KEY,
            Manager_Name VARCHAR (100),
3
            Manager_Email VARCHAR(100),
5
            Manager_Password VARCHAR(100),
6
            Manager_Picture VARCHAR (100),
            Manager_Hiredate DATE,
8
            Admin_ID INT,
            FOREIGN KEY (Admin_id) REFERENCES Manager (Admin_ID)
9
       );
10
```

Create Manager table



Manager table description

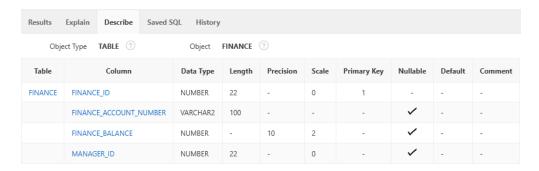
Create Manager Phone table



Manager Phone table description

```
CREATE TABLE Finance (
Finance_ID INT PRIMARY KEY,
Finance_Account_Number VARCHAR(100),
Finance_Balance DECIMAL(10, 2),
Manager_ID INT,
FOREIGN KEY (Manager_ID) REFERENCES Manager (Manager_ID)
);
```

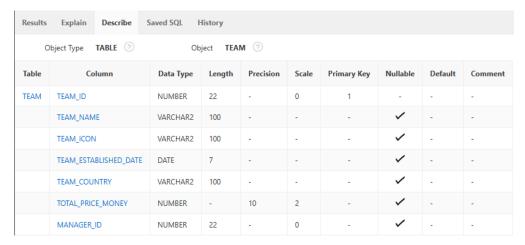
Create Finance table



Finance table description

```
CREATE TABLE Team (
1
2
            Team_ID INT PRIMARY KEY,
3
            Team_Name VARCHAR (100),
4
            Team_Icon VARCHAR (100),
5
            Team_Established_Date DATE,
6
            Team_Country VARCHAR(100),
7
            Total_Price_Money DECIMAL(10, 2),
            Manager_ID INT,
8
9
            FOREIGN KEY (Manager_ID) REFERENCES Manager (Manager_ID)
10
        );
```

Create Team table



Team table description

```
CREATE TABLE Team_Winning (
Tw_ID INT PRIMARY KEY,

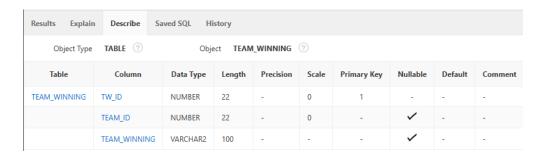
Team_ID INT,

Team_Winning VARCHAR(100),

FOREIGN KEY (Team_ID) REFERENCES Team (Team_ID)

);
```

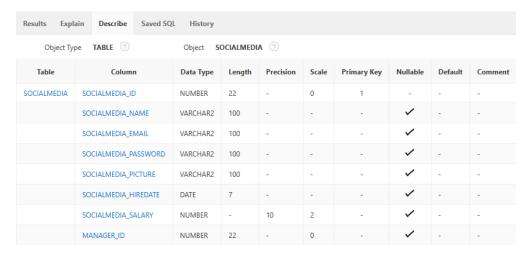
Create Team Winning table



Team Winning table description

```
1
        CREATE TABLE SocialMedia (
2
            SocialMedia_ID INT PRIMARY KEY,
3
            SocialMedia_Name VARCHAR(100),
            SocialMedia_Email VARCHAR(100),
4
5
            {\tt SocialMedia\_Password\ VARCHAR\,(100)\,,}
6
            SocialMedia_Picture VARCHAR(100),
            SocialMedia_Hiredate DATE,
7
8
            SocialMedia_Salary DECIMAL(10, 2),
9
            Manager_ID INT,
10
            FOREIGN KEY (Manager_ID) REFERENCES Manager (Manager_ID)
       );
```

Create SocialMedia table



SocialMedia table description

```
1
        CREATE TABLE ContentCreator (
2
            ContentCreator_ID INT PRIMARY KEY,
3
            ContentCreator_Name VARCHAR(100),
4
            ContentCreator_Email VARCHAR(100),
            ContentCreator_Password VARCHAR (100),
5
6
            ContentCreator_Picture VARCHAR(100),
7
            ContentCreator_Hiredate DATE,
8
            ContentCreator_Salary DECIMAL(10, 2),
9
            SocialMedia_ID INT,
10
            FOREIGN KEY (SocialMedia_ID) REFERENCES SocialMedia (SocialMedia_ID)
       );
11
```

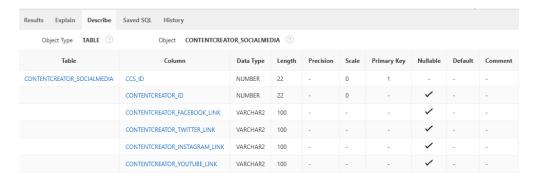
Create ContentCreator table



ContentCreator table description

```
1
       CREATE TABLE ContentCreator_SocialMedia (
2
            Ccs_ID INT PRIMARY KEY,
            ContentCreator_ID INT,
            {\tt ContentCreator\_Facebook\_Link~VARCHAR(100),}
4
            ContentCreator_Twitter_Link VARCHAR(100)
5
6
            ContentCreator_Instagram_Link VARCHAR(100),
7
            {\tt ContentCreator\_Youtube\_Link\ VARCHAR}\ (100)\ ,
            FOREIGN KEY (ContentCreator_ID) REFERENCES ContentCreator (ContentCreator_ID)
8
9
       );
```

Create ContentCreator SocialMedia table



ContentCreator SocialMedia table description

```
1
       CREATE TABLE ContentCreator_Address (
2
           Cca_ID INT PRIMARY KEY,
3
           ContentCreator_ID INT,
4
           ContentCreator_Country VARCHAR (100),
5
           ContentCreator_City VARCHAR(100),
6
           ContentCreator_Street VARCHAR(100),
           ContentCreator_Zip_Code VARCHAR(20),
7
           FOREIGN KEY (ContentCreator_ID) REFERENCES ContentCreator (ContentCreator_ID)
       );
```

Create ContentCreator Address table



ContentCreator Address table description

Create ContentCreator Phone table

```
CREATE TABLE SocialMedia_Phone (
Smp_ID INT PRIMARY KEY,
SocialMedia_ID INT,
SocialMedia_Phone VARCHAR(20),
FOREIGN KEY (SocialMedia_ID) REFERENCES SocialMedia (SocialMedia_ID)
);
```

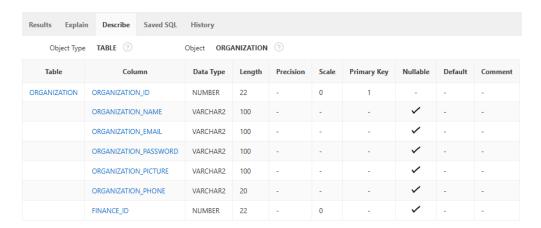
Create SocialMedia Phone table



SocialMedia Phone table description

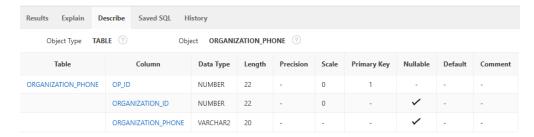
```
1
        CREATE TABLE Organization (
2
            Organization_ID INT PRIMARY KEY,
3
            Organization_Name VARCHAR(100),
            Organization_Email VARCHAR(100),
4
5
            {\tt Organization\_Password\ VARCHAR}\,({\tt 100})\,,
6
            Organization_Picture VARCHAR(100),
7
            Organization_Phone VARCHAR(20),
8
            Finance_ID INT,
9
            FOREIGN KEY (Finance_ID) REFERENCES Finance (Finance_ID)
10
        );
```

Create Organization table



Organization table description

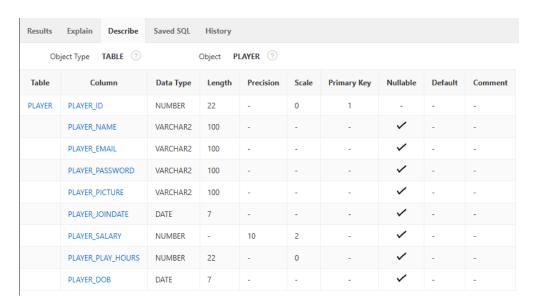
Create Organization Phone table



Organization Phone table description

```
CREATE TABLE Player (
2
             Player_ID INT PRIMARY KEY,
             Player_Name VARCHAR(100),
Player_Email VARCHAR(100),
3
4
5
             Player_Password VARCHAR (100),
6
             Player_Picture VARCHAR(100),
7
             Player_JoinDate DATE,
8
             Player_Salary DECIMAL(10, 2),
9
             Player_Play_Hours INT,
10
             Player_DOB DATE
        );
11
```

Create Player table



Player table description

```
CREATE TABLE Player_Address (
1
2
           Pa_ID INT PRIMARY KEY,
3
           Player_ID INT,
           Player_Country VARCHAR(100),
4
5
           Player_City VARCHAR (100),
           Player_Street VARCHAR (100),
6
7
           Player_Zip_Code VARCHAR(20),
           FOREIGN KEY (Player_ID) REFERENCES Player (Player_ID)
8
       );
```

Create Player Address table



Player Address table description

```
CREATE TABLE Player_Social_Link (
1
           Psl_ID INT PRIMARY KEY,
2
3
           Player_ID INT,
           Player_Facebook_Link VARCHAR(100),
5
           {\tt Player\_Instagram\_Link~VARCHAR(100),}
6
7
           Player_Twitter_Link VARCHAR(100),
            Player_Youtube_Link VARCHAR(100),
8
           FOREIGN KEY (Player_ID) REFERENCES Player (Player_ID)
9
       );
```

Create Player Social Link table



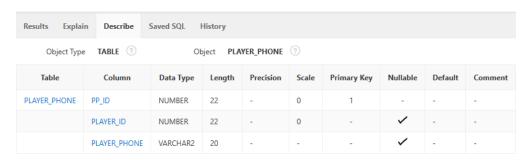
Player Social Link table description

```
CREATE TABLE Player_Phone (
Pp_ID INT PRIMARY KEY,

Player_ID INT,
Player_Phone VARCHAR(20),
FOREIGN KEY (Player_ID) REFERENCES Player (Player_ID)

);
```

Create Player Phone table



Player Phone table description

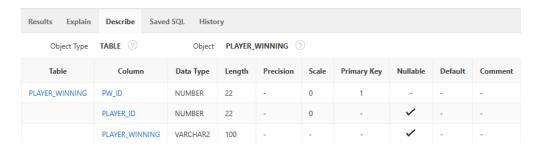
```
CREATE TABLE Player_Winning (
Pw_ID INT PRIMARY KEY,

Player_ID INT,

Player_Winning VARCHAR(100),
FOREIGN KEY (Player_ID) REFERENCES Player (Player_ID)

);
```

Create Player Winning table



Player Winning table description

```
CREATE TABLE Player_Team (

Pt_ID INT PRIMARY KEY,

Player_ID INT,

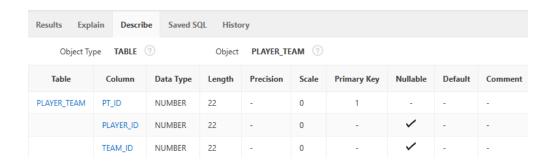
Team_ID INT,

FOREIGN KEY (Player_ID) REFERENCES Player (Player_ID),

FOREIGN KEY (Team_ID) REFERENCES Team (Team_ID)

7);
```

Create Player Team table



Player Team table description

```
1
        CREATE TABLE Tournament (
2
            Tournament_ID INT PRIMARY KEY,
3
            Tournament_Name VARCHAR (100),
4
            Tournament_StartingDate DATE,
5
            Tournament_EndingDate DATE,
6
            Tournament_Location VARCHAR(100),
7
            Tournament_Prize_Pool DECIMAL(10, 2),
            Organization_ID INT,
8
9
            FOREIGN KEY (Organization_ID) REFERENCES Organization (Organization_ID)
10
       );
```

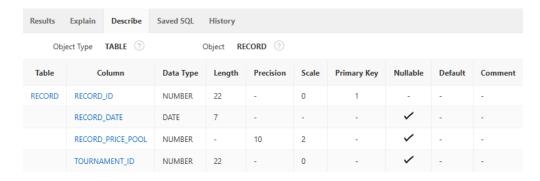
Create Tournament table



Tournament table description

```
CREATE TABLE Record (
Record_ID INT PRIMARY KEY,
Record_Date DATE,
Record_Price_Pool DECIMAL(10, 2),
Tournament_ID INT,
FOREIGN KEY (Tournament_ID) REFERENCES Tournament (Tournament_ID)
);
```

Create Record table



Record table description

```
CREATE TABLE Game (
2
            Game_ID INT PRIMARY KEY,
3
            Game_Name VARCHAR(100),
            Game_Icon VARCHAR(100),
5
            Game_ReleaseDate DATE,
6
            Game_Platform VARCHAR(100),
7
            Game_Price_Pool DECIMAL(10, 2),
8
            Game_Genre VARCHAR(100),
9
            Game_Publisher VARCHAR(100)
10
       );
```

Create Game table



Game table description

```
CREATE TABLE Tournament_Game (
Tournament_ID INT,
Game_ID INT,
PRIMARY KEY (Tournament_ID, Game_ID),
FOREIGN KEY (Tournament_ID) REFERENCES Tournament (Tournament_ID),
FOREIGN KEY (Game_ID) REFERENCES Game (Game_ID)
);
```

Create Tournament Game table



Tournament Game table description

```
CREATE TABLE Organization_Tournament (
Organization_ID INT,
Tournament_ID INT,
PRIMARY KEY (Organization_ID, Tournament_ID),
FOREIGN KEY (Organization_ID) REFERENCES Organization (Organization_ID),
FOREIGN KEY (Tournament_ID) REFERENCES Tournament (Tournament_ID)

7 );
```

Create Organization Tournament table

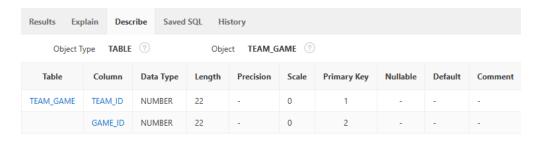


Organization Tournament table description

```
CREATE TABLE Team_Game (
Team_ID INT,
Game_ID INT,
PRIMARY KEY (Team_ID, Game_ID),
FOREIGN KEY (Team_ID) REFERENCES Team (Team_ID),
FOREIGN KEY (Game_ID) REFERENCES Game (Game_ID)

7 );
```

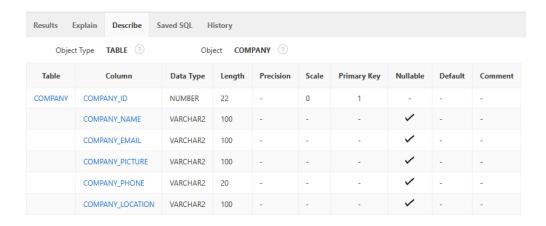
Create Team Game table



Team Game table description

```
CREATE TABLE Company (
Company_ID INT PRIMARY KEY,
Company_Name VARCHAR(100),
Company_Email VARCHAR(100),
Company_Picture VARCHAR(100),
Company_Phone VARCHAR(20),
Company_Location VARCHAR(100)
);
```

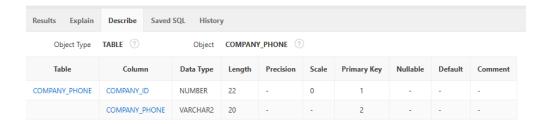
Create Company table



Company table description

```
CREATE TABLE Company_Phone (
Company_ID INT,
Company_Phone VARCHAR(20),
PRIMARY KEY (Company_ID, Company_Phone),
FOREIGN KEY (Company_ID) REFERENCES Company (Company_ID)
);
```

Create Company Phone table



Company Phone table description

Create Organization Company table



Organization Company table description

```
CREATE TABLE Team_Company (
Team_ID INT,
Company_ID INT,
PRIMARY KEY (Team_ID, Company_ID),
FOREIGN KEY (Team_ID) REFERENCES Team (Team_ID),
FOREIGN KEY (Company_ID) REFERENCES Company (Company_ID)
);
```

Create Team Company table



Team Company table description

5.2 Sequence Creation

```
-- Create sequence for Manager table
   CREATE SEQUENCE seq_manager_id START WITH 1 INCREMENT BY 1;
   -- Create sequence for Admin table
   CREATE SEQUENCE seq_admin_id START WITH 1 INCREMENT BY 1;
    -- Create sequence for Manager_Phone table
6
   CREATE SEQUENCE seq_mp_id START WITH 1 INCREMENT BY 1;
   -- Create sequence for Finance table
   CREATE SEQUENCE seq_finance_id START WITH 1 INCREMENT BY 1;
8
9
10
   -- Create sequence for Team table
   CREATE SEQUENCE seq_team_id START WITH 1 INCREMENT BY 1;
11
12
   -- Create sequence for Team_Winning table
   CREATE SEQUENCE seq_tw_id START WITH 1 INCREMENT BY 1;
13
14
    -- Create sequence for SocialMedia table
15
   CREATE SEQUENCE seq_socialmedia_id START WITH 1 INCREMENT BY 1;
16
    -- Create sequence for ContentCreator table
   CREATE SEQUENCE seq_contentcreator_id START WITH 1 INCREMENT BY 1;
17
   -- Create sequence for ContentCreator_SocialMedia table
18
19
   CREATE SEQUENCE seq_ccs_id START WITH 1 INCREMENT BY 1;
20
   -- Create sequence for ContentCreator_Address table
21
   CREATE SEQUENCE seq_cca_id START WITH 1 INCREMENT BY 1;
22
   -- Create sequence for ContentCreator_Phone table
23
   CREATE SEQUENCE seq_ccp_id START WITH 1 INCREMENT BY 1;
24
   -- Create sequence for SocialMedia_Phone table
25
26
   CREATE SEQUENCE seq_smp_id START WITH 1 INCREMENT BY 1;
27
    -- Create sequence for Organization table
   CREATE SEQUENCE seq_organization_id START WITH 1 INCREMENT BY 1;
28
   -- Create sequence for Organization_Phone table
30
   CREATE SEQUENCE seq_op_id START WITH 1 INCREMENT BY 1;
31
   -- Create sequence for Player table
   CREATE SEQUENCE seq_player_id START WITH 1 INCREMENT BY 1;
33
    -- Create sequence for Player_Address table
34
   CREATE SEQUENCE seq_pa_id START WITH 1 INCREMENT BY 1;
   -- Create sequence for Player_Social_Link table
35
36
   CREATE SEQUENCE seq_psl_id START WITH 1 INCREMENT BY 1;
37
38
   -- Create sequence for Player_Phone table
   CREATE SEQUENCE seq_pp_id START WITH 1 INCREMENT BY 1;
39
40

    Create sequence for Player_Winning table

   CREATE SEQUENCE seq_pw_id START WITH 1 INCREMENT BY 1;
41
42
   -- Create sequence for Record table
43
   CREATE SEQUENCE seq_record_id START WITH 1 INCREMENT BY 1;
44
   -- Create sequence for Tournament table
   CREATE SEQUENCE seq_tournament_id START WITH 1 INCREMENT BY 1;
    -- Create sequence for Game table
46
   CREATE SEQUENCE seq_game_id START WITH 1 INCREMENT BY 1;
47
    -- Create sequence for Company table
   CREATE SEQUENCE seq_company_id START WITH 1 INCREMENT BY 1;
49
50
   -- Create sequence for Company_Phone table
   CREATE SEQUENCE seq_cp_id START WITH 1 INCREMENT BY 1;
51
52
   -- Create sequence for Organization_Company table
   CREATE SEQUENCE seq_oc_id START WITH 1 INCREMENT BY 1;
53
54
```

Sequence Creation

5.3 Index for Table

```
1
        -- Create index for Manager table
2
       CREATE INDEX idx_manager_email ON Manager (Manager_Email);
3
        -- Create index for Admin table
4
       CREATE INDEX idx_admin_email ON Admin (Admin_Email);
        -- Create index for Manager_Phone table
       CREATE INDEX idx_manager_phone_manager_id ON Manager_Phone (Manager_ID);
6
7
        -- Create index for Finance table
8
       CREATE INDEX idx_finance_manager_id ON Finance (Manager_ID);
9
        -- Create index for Team table
10
       CREATE INDEX idx_team_manager_id ON Team (Manager_ID);
11
12
       -- Create index for Team_Winning table
       CREATE INDEX idx_team_winning_team_id ON Team_Winning (Team_ID);
13
14
        -- Create index for SocialMedia table
15
       CREATE INDEX idx_socialmedia_name ON SocialMedia (SocialMedia_Name);
16
         - Create index for ContentCreator table
       CREATE INDEX idx_contentcreator_name ON ContentCreator (ContentCreator_Name);
17
        -- Create index for ContentCreator_SocialMedia table
18
       CREATE INDEX idx_ccs_contentcreator_id ON ContentCreator_SocialMedia (ContentCreator_ID)
19
20
        -- Create index for ContentCreator_Address table
       CREATE INDEX idx_cca_contentcreator_id ON ContentCreator_Address (ContentCreator_ID);
21
22
        -- Create index for ContentCreator_Phone table
23
       CREATE INDEX idx_ccp_contentcreator_id ON ContentCreator_Phone (ContentCreator_ID);
24
        -- Create index for SocialMedia_Phone table
25
       CREATE INDEX idx_smp_socialmedia_id ON SocialMedia_Phone (SocialMedia_ID);
26
        - Create index for Organization table
27
       CREATE INDEX idx_organization_name ON Organization (Organization_Name);
        -- Create index for Organization_Phone table
29
       CREATE INDEX idx_op_organization_id ON Organization_Phone (Organization_ID);
30
        -- Create index for Player table
31
       CREATE INDEX idx_player_name ON Player (Player_Name);
32
33
        -- Create index for Player_Address table
       CREATE INDEX idx_pa_player_id ON Player_Address (Player_ID);
34
35
        -- Create index for Player_Social_Link table
36
       CREATE INDEX idx_psl_player_id ON Player_Social_Link (Player_ID);
37
        -- Create index for Player_Phone table
38
       CREATE INDEX idx_pp_player_id ON Player_Phone (Player_ID);
39
          Create index for Player_Winning table
       CREATE INDEX idx_pw_player_id ON Player_Winning (Player_ID);
40
41
        -- Create index for Record table
42
       CREATE INDEX idx_record_date ON Record (Record_Date);
43
        -- Create index for Tournament table
       CREATE INDEX idx_tournament_name ON Tournament (Tournament_Name);
45
46
        -- Create index for Game table
47
       CREATE INDEX idx_game_name ON Game (Game_Name);
        -- Create index for Company table
48
49
       CREATE INDEX idx_company_name ON Company (Company_Name);
        -- Create index for Company_Phone table
50
       CREATE INDEX idx_cp_company_id ON Company_Phone (Company_ID);
51
        -- Create index for Organization_Company table
        CREATE INDEX idx_oc_organization_id ON Organization_Company (Organization_ID);
53
54
```

Index for Table

5.4 Alter Table for effective indexing

```
1
        -- Alter Manager table to add index
2
       ALTER TABLE Manager ADD CONSTRAINT idx_manager_email UNIQUE (Manager_Email);
3
        -- Alter Admin table to add index
4
       ALTER TABLE Admin ADD CONSTRAINT idx_admin_email UNIQUE (Admin_Email);
5
        -- Alter Manager_Phone table to add index
       ALTER TABLE Manager_Phone ADD CONSTRAINT idx_manager_phone_manager_id UNIQUE (Manager_ID
6
           );
7
          Alter Finance table to add index
       ALTER TABLE Finance ADD CONSTRAINT idx_finance_manager_id UNIQUE (Manager_ID);
8
9
        -- Alter Team table to add index
       ALTER TABLE Team ADD CONSTRAINT idx_team_manager_id UNIQUE (Manager_ID);
10
11
        -- Alter Team_Winning table to add index
       ALTER TABLE Team_Winning ADD CONSTRAINT idx_team_winning_team_id UNIQUE (Team_ID);
12
13
14
        -- Alter SocialMedia table to add index
15
       ALTER TABLE SocialMedia ADD CONSTRAINT idx_socialmedia_name UNIQUE (SocialMedia_Name);
16
        -- Alter ContentCreator table to add index
       ALTER TABLE ContentCreator ADD CONSTRAINT idx_contentcreator_name UNIQUE (
17
           ContentCreator_Name);
18
        -- Alter ContentCreator_SocialMedia table to add index
19
       ALTER TABLE ContentCreator_SocialMedia ADD CONSTRAINT idx_ccs_contentcreator_id UNIQUE (
           ContentCreator_ID);
20
       -- Alter ContentCreator_Address table to add index
21
       ALTER TABLE ContentCreator_Address ADD CONSTRAINT idx_cca_contentcreator_id UNIQUE (
           ContentCreator_ID);
22
        -- Alter ContentCreator_Phone table to add index
23
       ALTER TABLE ContentCreator_Phone ADD CONSTRAINT idx_ccp_contentcreator_id UNIQUE (
           ContentCreator_ID);
        -- Alter SocialMedia_Phone table to add index
25
       ALTER TABLE SocialMedia_Phone ADD CONSTRAINT idx_smp_socialmedia_id UNIQUE (
           SocialMedia_ID);
        -- Alter Organization table to add index
       ALTER TABLE Organization ADD CONSTRAINT idx_organization_name UNIQUE (Organization_Name)
27
        -- Alter Organization_Phone table to add index
28
29
       ALTER TABLE Organization_Phone ADD CONSTRAINT idx_op_organization_id UNIQUE (
           Organization_ID);
30
31
        -- Alter Player table to add index
32
       ALTER TABLE Player ADD CONSTRAINT idx_player_name UNIQUE (Player_Name);
33
        -- Alter Player_Address table to add index
34
        ALTER TABLE Player_Address ADD CONSTRAINT idx_pa_player_id UNIQUE (Player_ID);
35
        -- Alter Player_Social_Link table to add index
       ALTER TABLE Player_Social_Link ADD CONSTRAINT idx_psl_player_id UNIQUE (Player_ID);
36
        -- Alter Player_Phone table to add index
       ALTER TABLE Player_Phone ADD CONSTRAINT idx_pp_player_id UNIQUE (Player_ID);
38
39
        -- Alter Player_Winning table to add index
       ALTER TABLE Player_Winning ADD CONSTRAINT idx_pw_player_id UNIQUE (Player_ID);
41
        -- Alter Record table to add index
42
       ALTER TABLE Record ADD CONSTRAINT idx_record_date UNIQUE (Record_Date);
43
44
        -- Alter Tournament table to add index
       ALTER TABLE Tournament ADD CONSTRAINT idx_tournament_name UNIQUE (Tournament_Name);
        -- Alter Game table to add index
46
47
       ALTER TABLE Game ADD CONSTRAINT idx_game_name UNIQUE (Game_Name);
        -- Alter Company table to add index
       ALTER TABLE Company ADD CONSTRAINT idx_company_name UNIQUE (Company_Name);
49
50
        -- Alter Company_Phone table to add index
51
       ALTER TABLE Company_Phone ADD CONSTRAINT idx_cp_company_id UNIQUE (Company_ID);
52
        -- Alter Organization_Company table to add index
        ALTER TABLE Organization_Company ADD CONSTRAINT idx_oc_organization_id UNIQUE (
           Organization_ID);
54
```

Alter Table

5.5 Data Insertion

```
INSERT INTO Admin (Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture)
1
   VALUES (seq_admin_id.NEXTVAL, 'Admin 1', 'admin1@example.com', 'adminpass1', 'admin1.jpg');
3
   INSERT INTO Admin (Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture)
   VALUES (seq_admin_id.NEXTVAL, 'Admin 2', 'admin2@example.com', 'adminpass2', 'admin2.jpg');
6
   INSERT INTO Admin (Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture)
8
   VALUES (seq_admin_id.NEXTVAL, 'Admin 3', 'admin3@example.com', 'adminpass3', 'admin3.jpg');
9
   INSERT INTO Admin (Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture)
11
   VALUES (seq_admin_id.NEXTVAL, 'Admin 4', 'admin4@example.com', 'adminpass4', 'admin4.jpg');
12
13
   INSERT INTO Admin (Admin_ID, Admin_Name, Admin_Email, Admin_Password, Admin_Picture)
   VALUES (seq_admin_id.NEXTVAL, 'Admin 5', 'admin5@example.com', 'adminpass5', 'admin5.jpg');
14
```

Inserting data into Admin table

Results	Explain	Describe Saved SQL	History		
			•		
ADMI	IN_ID	ADMIN_NAME	ADMIN_EMAIL	ADMIN_PASSWORD	ADMIN_PICTURE
1		Admin 1	admin1@example.com	adminpass1	admin1.jpg
2		Admin 2	admin2@example.com	adminpass2	admin2.jpg
3		Admin 3	admin3@example.com	adminpass3	admin3.jpg
4		Admin 4	admin4@example.com	adminpass4	admin4.jpg
5		Admin 5	admin5@example.com	adminpass5	admin5.jpg

5 rows returned in 0.00 seconds

Inserted data of Admin table

```
INSERT INTO Manager (Manager_ID, Manager_Name, Manager_Email, Manager_Password,
1
        Manager_Picture, Manager_Hiredate, Admin_ID)
    VALUES (seq_manager_id.NEXTVAL, 'John Doe', 'john.doe@example.com', 'password123', 'profile.
    jpg', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 1);
2
3
    INSERT INTO Manager (Manager_ID, Manager_Name, Manager_Email, Manager_Password,
4
        Manager_Picture, Manager_Hiredate, Admin_ID)
    VALUES (seq_manager_id.NEXTVAL, 'Jane Smith', 'jane.smith@example.com', 'password456', '
5
       profile2.jpg', TO_DATE('2022-02-01', 'YYYY-MM-DD'), 2);
6
7
    INSERT INTO Manager (Manager_ID, Manager_Name, Manager_Email, Manager_Password,
       Manager_Picture, Manager_Hiredate, Admin_ID)
8
    VALUES (seq_manager_id.NEXTVAL, 'Mike Johnson', 'mike.johnson@example.com', 'password789', '
       profile3.jpg', TO_DATE('2022-03-01', 'YYYY-MM-DD'), 1);
9
10
    INSERT INTO Manager (Manager_ID, Manager_Name, Manager_Email, Manager_Password,
       Manager_Picture, Manager_Hiredate, Admin_ID)
11
    VALUES (seq_manager_id.NEXTVAL, 'Sarah Williams', 'sarah.williams@example.com', 'password123
        ', 'profile4.jpg', TO_DATE('2022-04-01', 'YYYY-MM-DD'), 2);
12
13
    INSERT INTO Manager (Manager_ID, Manager_Name, Manager_Email, Manager_Password,
        Manager_Picture, Manager_Hiredate, Admin_ID)
    VALUES (seq_manager_id.NEXTVAL, 'Robert Davis', 'robert.davis@example.com', 'password456', '
14
        profile5.jpg', TO_DATE('2022-05-01', 'YYYY-MM-DD'), 1);
```

Inserting data into manager tables



Inserted data of manager table

```
1
   INSERT INTO Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
2
   VALUES (seq_mp_id.NEXTVAL, 1, '1234567890');
3
   INSERT INTO Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
5
   VALUES (seq_mp_id.NEXTVAL, 2, '0987654321');
6
   INSERT INTO Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
7
8
   VALUES (seq_mp_id.NEXTVAL, 3, '1112223333');
9
10
   INSERT INTO Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
11
   VALUES (seq_mp_id.NEXTVAL, 4, '4445556666');
12
13
   INSERT INTO Manager_Phone (Mp_ID, Manager_ID, Manager_Phone)
14
   VALUES (seq_mp_id.NEXTVAL, 5, '7778889999');
```

Inserting data into manager phone tables

Results Explain Describe Saved	SQL History	
MP_ID	MANAGER_ID	MANAGER_PHONE
1	1	1234567890
2	2	0987654321
3	3	1112223333
4	4	4445556666
5	5	7778889999
5 rows returned in 0.00 seconds		

Inserted data of manager phone table

```
INSERT INTO Finance (Finance_ID, Finance_Account_Number, Finance_Balance, Manager_ID)
    VALUES (seq_finance_id.NEXTVAL, 'ABC123456', 10000, 1);
3
    INSERT INTO Finance (Finance_ID, Finance_Account_Number, Finance_Balance, Manager_ID)
    VALUES (seq_finance_id.NEXTVAL, 'DEF789012', 20000, 2);
5
6
7
    INSERT INTO Finance (Finance_ID, Finance_Account_Number, Finance_Balance, Manager_ID)
    VALUES (seq_finance_id.NEXTVAL, 'GHI345678', 15000, 3);
8
9
    INSERT INTO Finance (Finance_ID, Finance_Account_Number, Finance_Balance, Manager_ID)
VALUES (seq_finance_id.NEXTVAL, 'JKL901234', 18000, 4);
10
11
12
    INSERT INTO Finance (Finance_ID, Finance_Account_Number, Finance_Balance, Manager_ID)
VALUES (seq_finance_id.NEXTVAL, 'MN0567890', 22000, 5);
13
14
```

Inserting data into Finance tables

Results Explain Describe Saved SQL History					
FINANCE_ID	FINANCE_ACCOUNT_NUMBER	FINANCE_BALANCE	MANAGER_ID		
1	ABC123456	10000	1		
2	DEF789012	20000	2		
3	GHI345678	15000	3		
4	JKL901234	18000	4		
5	MNO567890	22000	5		
5 rows returned in 0.00 seconds					

Inserted data of Finance table

```
1
2
   INSERT INTO Team (Team_ID, Team_Name, Team_Icon, Team_Established_Date, Team_Country,
       Total_Price_Money , Manager_ID)
3
   VALUES (seq_team_id.NEXTVAL, 'Team A', 'teamA.png', TO_DATE('2022-01-01', 'YYYY-MM-DD'), '
       USA', 50000, 1);
4
5
   INSERT INTO Team (Team_ID, Team_Name, Team_Icon, Team_Established_Date, Team_Country,
       Total_Price_Money , Manager_ID)
6
   VALUES (seq_team_id.NEXTVAL, 'Team B', 'teamB.png', TO_DATE('2022-02-01', 'YYYY-MM-DD'), 'UK
       <sup>'</sup>, 60000, 2);
7
8
   INSERT INTO Team (Team_ID, Team_Name, Team_Icon, Team_Established_Date, Team_Country,
       Total_Price_Money , Manager_ID)
   VALUES (seq_team_id.NEXTVAL, 'Team C', 'teamC.png', TO_DATE('2022-03-01', 'YYYY-MM-DD'), '
9
       Australia', 45000, 3);
10
11
   INSERT INTO Team (Team_ID, Team_Name, Team_Icon, Team_Established_Date, Team_Country,
       Total_Price_Money, Manager_ID)
   VALUES (seq_team_id.NEXTVAL, 'Team D', 'teamD.png', TO_DATE('2022-04-01', 'YYYY-MM-DD'), '
12
       Canada', 55000, 4);
13
   INSERT INTO Team (Team_ID, Team_Name, Team_Icon, Team_Established_Date, Team_Country,
14
       Total_Price_Money, Manager_ID)
   VALUES (seq_team_id.NEXTVAL, 'Team E', 'teamE.png', TO_DATE('2022-05-01', 'YYYY-MM-DD'), '
15
       Germany', 70000, 5);
```

Inserting data into Team tables

Results Explain Describe Saved SQL History						
TEAM_NAME	TEAM_ICON	TEAM_ESTABLISHED_DATE	TEAM_COUNTRY	TOTAL_PRICE_MONEY	MANAGER_ID	
Team A	teamA.png	01-JAN-22	USA	50000	1	
Team B	teamB.png	01-FEB-22	UK	60000	2	
Team C	teamC.png	01-MAR-22	Australia	45000	3	
Team D	teamD.png	01-APR-22	Canada	55000	4	
Team E	teamE.png	01-MAY-22	Germany	70000	5	
	Team A Team B Team C Team D	Team A teamA.png Team B teamB.png Team C teamC.png Team D teamD.png Team E teamE.png	Team A teamA.png 01-JAN-22 Team B teamB.png 01-FEB-22 Team C teamC.png 01-MAR-22 Team D teamD.png 01-APR-22 Team E teamE.png 01-MAY-22	Team A teamA.png 01-JAN-22 USA Team B teamB.png 01-FEB-22 UK Team C teamC.png 01-MAR-22 Australia Team D teamD.png 01-APR-22 Canada Team E teamE.png 01-MAY-22 Germany	Team A teamA.png 01-JAN-22 USA 50000 Team B teamB.png 01-FEB-22 UK 60000 Team C teamC.png 01-MAR-22 Australia 45000 Team D teamD.png 01-APR-22 Canada 55000 Team E teamE.png 01-MAY-22 Germany 70000	

Inserted data of Team table

```
2
   INSERT INTO Team_Winning (Tw_ID, Team_ID, Team_Winning)
3
   VALUES (seq_tw_id.NEXTVAL, 1, 'Championship 2022');
4
   INSERT INTO Team_Winning (Tw_ID, Team_ID, Team_Winning)
5
6
   VALUES (seq_tw_id.NEXTVAL, 2, 'Tournament 2023');
7
   INSERT INTO Team_Winning (Tw_ID, Team_ID, Team_Winning)
8
9
   VALUES (seq_tw_id.NEXTVAL, 3, 'Cup 2022');
10
11
   INSERT INTO Team_Winning (Tw_ID, Team_ID, Team_Winning)
12
   VALUES (seq_tw_id.NEXTVAL, 4, 'League 2022');
13
14
   INSERT INTO Team_Winning (Tw_ID, Team_ID, Team_Winning)
15
   VALUES (seq_tw_id.NEXTVAL, 5, 'Championship 2023');
```

Inserting data into Team_Winning tables

Results	Explain Describe Saved SQL	History	
	TW_ID	TEAM_ID	TEAM_WINNING
1		1	Championship 2022
2		2	Tournament 2023
3		3	Cup 2022
4		4	League 2022
5		5	Championship 2023
5 rows ref	turned in 0.00 seconds		

Inserted data of Team Winning table

```
INSERT INTO SocialMedia (SocialMedia_ID, SocialMedia_Name, SocialMedia_Email,
2
       SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary,
       MANAGER ID)
3
   VALUES (seq_socialmedia_id.NEXTVAL, 'SocialMediaUser221', 'social.user1@example.com', '
       socialpass1', 'social1.jpg', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 5000, 1);
5
   INSERT INTO SocialMedia (SocialMedia_ID, SocialMedia_Name, SocialMedia_Email,
       SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary,
       MANAGER_ID)
6
   VALUES (seq_socialmedia_id.NEXTVAL, 'SocialMediaUser442', 'social.user2@example.com', '
       socialpass2', 'social2.jpg', TO_DATE('2022-02-01', 'YYYY-MM-DD'), 6000, 2);
   INSERT INTO SocialMedia (SocialMedia_ID, SocialMedia_Name, SocialMedia_Email,
8
       SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary,
       MANAGER_ID)
9
   VALUES (seq_socialmedia_id.NEXTVAL, 'SocialMediaUser423', 'social.user3@example.com', '
       socialpass3', 'social3.jpg', TO_DATE('2022-03-01', 'YYYY-MM-DD'), 7000, 3);
10
11
   INSERT INTO SocialMedia (SocialMedia_ID, SocialMedia_Name, SocialMedia_Email,
       SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary,
       MANAGER ID)
   VALUES (seq_socialmedia_id.NEXTVAL, 'SocialMediaUser4234', 'social.user4@example.com', '
12
       socialpass4', 'social4.jpg', TO_DATE('2022-04-01', 'YYYY-MM-DD'), 8000, 4);
13
   INSERT INTO SocialMedia (SocialMedia_ID, SocialMedia_Name, SocialMedia_Email,
14
       SocialMedia_Password, SocialMedia_Picture, SocialMedia_Hiredate, SocialMedia_Salary,
       MANAGER_ID)
   VALUES (seq_socialmedia_id.NEXTVAL, 'SocialMediaUser5523', 'social.user5@example.com', '
15
       socialpass5', 'social5.jpg', TO_DATE('2022-05-01', 'YYYY-MM-DD'), 9000, 5);
```

Inserting data into SocialMedia tables



Inserted data of Social Media table

```
INSERT INTO ContentCreator (ContentCreator_ID, ContentCreator_Name, ContentCreator_Email,
       ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate,
       ContentCreator_Salary, SOCIALMEDIA_ID)
2
   VALUES (seq_contentcreator_id.NEXTVAL, 'ContentCreator 1', 'cc1@example.com', 'ccpass1', '
       cc1.jpg', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 3000, 1)
3
4
   INSERT INTO ContentCreator (ContentCreator_ID, ContentCreator_Name, ContentCreator_Email,
       ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate,
        ContentCreator_Salary, SOCIALMEDIA_ID)
   VALUES (seq_contentcreator_id.NEXTVAL, 'ContentCreator 2', 'cc2@example.com', 'ccpass2', '
5
       cc2.jpg', TO_DATE('2022-02-01', 'YYYY-MM-DD'), 3500, 2)
6
   INSERT INTO ContentCreator (ContentCreator_ID, ContentCreator_Name, ContentCreator_Email,
7
        ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate,
       ContentCreator_Salary, SOCIALMEDIA_ID)
   VALUES (seq_contentcreator_id.NEXTVAL, 'ContentCreator 3', 'cc3@example.com', 'ccpass3', '
8
       cc3.jpg', TO_DATE('2022-03-01', 'YYYY-MM-DD'), 4000, 3)
9
10
   INSERT INTO ContentCreator (ContentCreator_ID, ContentCreator_Name, ContentCreator_Email,
       ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate,
       {\tt ContentCreator\_Salary}\;,\;\; {\tt SOCIALMEDIA\_ID})
11
   VALUES (seq_contentcreator_id.NEXTVAL, 'ContentCreator 4', 'cc4@example.com', 'ccpass4', '
       cc4.jpg', TO_DATE('2022-04-01', 'YYYY-MM-DD'), 4500, 4)
12
13
   INSERT INTO ContentCreator (ContentCreator_ID, ContentCreator_Name, ContentCreator_Email,
       ContentCreator_Password, ContentCreator_Picture, ContentCreator_Hiredate,
        ContentCreator_Salary, SOCIALMEDIA_ID)
14
   VALUES (seq_contentcreator_id.NEXTVAL, 'ContentCreator 5', 'cc5@example.com', 'ccpass5', '
       cc5.jpg', TO_DATE('2022-05-01', 'YYYY-MM-DD'), 5000, 5)
```

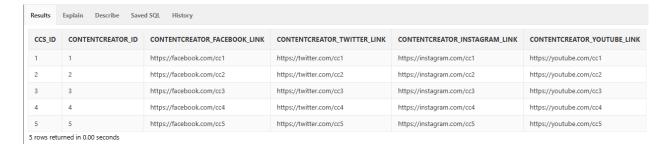
Inserting data into ContentCreator tables



Inserted data of Content Creator table

```
1
    INSERT INTO ContentCreator_SocialMedia (Ccs_ID, ContentCreator_ID,
        ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_Instagram_Link
          ContentCreator_Youtube_Link)
 2
    VALUES (seq_ccs_id.NEXTVAL, 1, 'https://facebook.com/cc1', 'https://twitter.com/cc1', 'https
        ://instagram.com/cc1', 'https://youtube.com/cc1');
 3
    INSERT INTO ContentCreator_SocialMedia (Ccs_ID, ContentCreator_ID,
        {\tt ContentCreator\_Facebook\_Link}\;,\;\; {\tt ContentCreator\_Twitter\_Link}\;,\;\; {\tt ContentCreator\_Instagram\_Link}
          ContentCreator_Youtube_Link)
    VALUES (seq_ccs_id.NEXTVAL, 2, 'https://facebook.com/cc2', 'https://twitter.com/cc2', 'https://instagram.com/cc2', 'https://youtube.com/cc2');
 5
 6
 7
    INSERT INTO ContentCreator_SocialMedia (Ccs_ID, ContentCreator_ID,
        {\tt ContentCreator\_Facebook\_Link}\;,\;\; {\tt ContentCreator\_Twitter\_Link}\;,\;\; {\tt ContentCreator\_Instagram\_Link}\;
          ContentCreator_Youtube_Link)
    VALUES (seq_ccs_id.NEXTVAL, 3, 'https://facebook.com/cc3', 'https://twitter.com/cc3', 'https://instagram.com/cc3', 'https://youtube.com/cc3');
 8
10
    INSERT INTO ContentCreator_SocialMedia (Ccs_ID, ContentCreator_ID,
        {\tt ContentCreator\_Facebook\_Link}\;,\;\; {\tt ContentCreator\_Twitter\_Link}\;,\;\; {\tt ContentCreator\_Instagram\_Link}\;
         , ContentCreator_Youtube_Link)
11
    VALUES (seq_ccs_id.NEXTVAL, 4, 'https://facebook.com/cc4', 'https://twitter.com/cc4', 'https
        ://instagram.com/cc4', 'https://youtube.com/cc4');
12
13
    INSERT INTO ContentCreator_SocialMedia (Ccs_ID, ContentCreator_ID,
        ContentCreator_Facebook_Link, ContentCreator_Twitter_Link, ContentCreator_Instagram_Link
          ContentCreator_Youtube_Link)
14
    VALUES (seq_ccs_id.NEXTVAL, 5, 'https://facebook.com/cc5', 'https://twitter.com/cc5', 'https
        ://instagram.com/cc5', 'https://youtube.com/cc5');
```

Inserting data into ContentCreator_SocialMedia tables



Inserted data of Content Creator Social Media table

```
1
   INSERT INTO ContentCreator_Address (Cca_ID, ContentCreator_ID, ContentCreator_Country,
        ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)
2
   VALUES (seq_cca_id.NEXTVAL, 1, 'USA', 'New York', '123 Street', '10001');
3
   INSERT INTO ContentCreator_Address (Cca_ID, ContentCreator_ID, ContentCreator_Country,
4
        ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)
   VALUES (seq_cca_id.NEXTVAL, 2, 'USA', 'Los Angeles', '456 Avenue', '90001');
5
6
7
   INSERT INTO ContentCreator_Address (Cca_ID, ContentCreator_ID, ContentCreator_Country,
       {\tt ContentCreator\_City}\;,\;\; {\tt ContentCreator\_Street}\;,\;\; {\tt ContentCreator\_Zip\_Code})
8
   VALUES (seq_cca_id.NEXTVAL, 3, 'UK', 'London', '789 Road', 'SW1A 1AA');
10
   INSERT INTO ContentCreator_Address (Cca_ID, ContentCreator_ID, ContentCreator_Country,
        ContentCreator_City, ContentCreator_Street, ContentCreator_Zip_Code)
11
   VALUES (seq_cca_id.NEXTVAL, 4, 'Canada', 'Toronto', '321 Boulevard', 'M5V 2T3');
12
13
   INSERT INTO ContentCreator_Address (Cca_ID, ContentCreator_ID, ContentCreator_Country,
        {\tt ContentCreator\_City}\;,\;\; {\tt ContentCreator\_Street}\;,\;\; {\tt ContentCreator\_Zip\_Code})
14
   VALUES (seq_cca_id.NEXTVAL, 5, 'Germany', 'Berlin', '987 Strasse', '12345');
```

Inserting data into ContentCreator_Address tables

Results Explain Describe Saved SQL History						
CCA_ID	CONTENTCREATOR_ID	CONTENTCREATOR_COUNTRY	CONTENTCREATOR_CITY	CONTENTCREATOR_STREET	CONTENTCREATOR_ZIP_CODE	
1	1	USA	New York	123 Street	10001	
2	2	USA	Los Angeles	456 Avenue	90001	
3	3	UK	London	789 Road	SW1A 1AA	
4	4	Canada	Toronto	321 Boulevard	M5V 2T3	
5	5	Germany	Berlin	987 Strasse	12345	
5 rows returne	d in 0.00 seconds					

Inserted data of Content Creator Address table

```
INSERT INTO ContentCreator_Phone (Ccp_ID, ContentCreator_ID, ContentCreator_Phone)
   VALUES (seq_ccp_id.NEXTVAL, 1, '9876543210');
3
   INSERT INTO ContentCreator_Phone (Ccp_ID, ContentCreator_ID, ContentCreator_Phone)
   VALUES (seq_ccp_id.NEXTVAL, 2, '1234567890');
5
6
7
   INSERT INTO ContentCreator_Phone (Ccp_ID, ContentCreator_ID, ContentCreator_Phone)
8
   VALUES (seq_ccp_id.NEXTVAL, 3, '5551234567');
9
10
   INSERT INTO ContentCreator_Phone (Ccp_ID, ContentCreator_ID, ContentCreator_Phone)
   VALUES (seq_ccp_id.NEXTVAL, 4, '7775558888');
11
12
13
   INSERT INTO ContentCreator_Phone (Ccp_ID, ContentCreator_ID, ContentCreator_Phone)
   VALUES (seq_ccp_id.NEXTVAL, 5, '9990001111');
14
```

Inserting data into ContentCreator_Phone tables



Inserted data of Content Creator Phone table

```
INSERT INTO SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)
2
   VALUES (seq_smp_id.NEXTVAL, 1, '5551234567');
3
   INSERT INTO SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)
   VALUES (seq_smp_id.NEXTVAL, 2, '6669876543');
5
6
7
   INSERT INTO SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)
   VALUES (seq_smp_id.NEXTVAL, 3, '7774561230');
9
10
   INSERT INTO SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)
11
   VALUES (seq_smp_id.NEXTVAL, 4, '8887890123');
12
13
   INSERT INTO SocialMedia_Phone (Smp_ID, SocialMedia_ID, SocialMedia_Phone)
14
   VALUES (seq_smp_id.NEXTVAL, 5, '9996547890');
```

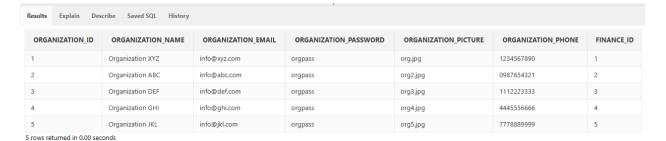
Inserting data into SocialMedia_Phone tables

Results Explain Describe Saved	SQL History	
SMP_ID	SOCIALMEDIA_ID	SOCIALMEDIA_PHONE
1	1	5551234567
2	2	6669876543
3	3	7774561230
4	4	8887890123
5	5	9996547890
5 rows returned in 0.00 seconds		

Inserted data of Social Media Phone table

```
INSERT INTO Organization (Organization_ID, Organization_Name, Organization_Email,
1
       Organization_Password, Organization_Picture, Organization_Phone, Finance_ID)
2
   VALUES (seq_organization_id.NEXTVAL, 'Organization XYZ', 'info@xyz.com', 'orgpass', 'org.jpg
3
   INSERT INTO Organization (Organization_ID, Organization_Name,Organization_Email,
4
       Organization_Password, Organization_Picture, Organization_Phone, Finance_ID)
   VALUES (seq_organization_id.NEXTVAL, 'Organization ABC', 'info@abc.com', 'orgpass', 'org2.
5
       jpg',
             '0987654321', 2);
6
   INSERT INTO Organization (Organization_ID, Organization_Name, Organization_Email,
7
       Organization_Password, Organization_Picture, Organization_Phone, Finance_ID)
8
   VALUES (seq_organization_id.NEXTVAL, 'Organization DEF', 'info@def.com', 'orgpass', 'org3.
              '1112223333', 3);
9
10
   INSERT INTO Organization (Organization_ID, Organization_Name, Organization_Email,
       Organization_Password, Organization_Picture, Organization_Phone, Finance_ID)
11
   VALUES (seq_organization_id.NEXTVAL, 'Organization GHI', 'info@ghi.com', 'orgpass', 'org4.
       jpg', '4445556666', 4);
12
   INSERT INTO Organization (Organization_ID, Organization_Name, Organization_Email,
13
       Organization_Password, Organization_Picture, Organization_Phone, Finance_ID)
   VALUES (seq_organization_id.NEXTVAL, 'Organization JKL', 'info@jkl.com', 'orgpass', 'org5.
       jpg', '7778889999', 5);
```

Inserting data into Organization tables



Inserted data of Organization table

```
INSERT INTO Organization_Phone (Op_ID, Organization_ID, Organization_Phone)
2
   VALUES (seq_op_id.NEXTVAL, 1, '9998887777');
3
   INSERT INTO Organization_Phone (Op_ID, Organization_ID, Organization_Phone)
   VALUES (seq_op_id.NEXTVAL, 2, '8887776666');
5
6
7
   INSERT INTO Organization_Phone (Op_ID, Organization_ID, Organization_Phone)
8
   VALUES (seq_op_id.NEXTVAL, 3, '7776665555');
9
10
   INSERT INTO Organization_Phone (Op_ID, Organization_ID, Organization_Phone)
   VALUES (seq_op_id.NEXTVAL, 4, '6665554444');
11
12
13
   INSERT INTO Organization_Phone (Op_ID, Organization_ID, Organization_Phone)
   VALUES (seq_op_id.NEXTVAL, 5, '5554443333');
14
```

Inserting data into Organization_Phone tables

Results	Explain Describe	Saved SQL History	
	OP_ID	ORGANIZATION_ID	ORGANIZATION_PHONE
1		1	9998887777
2		2	8887776666
3		3	7776665555
4		4	6665554444
5		5	5554443333
5 rows ret	turned in 0.00 seconds		

Inserted data of Organization Phone table

```
1
     INSERT INTO Player (Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture,
          {\tt Player\_JoinDate}\;,\;\; {\tt Player\_Salary}\;,\;\; {\tt Player\_Play\_Hours}\;,\;\; {\tt Player\_DOB})
     VALUES (seq_player_id.NEXTVAL, 'Player 1', 'player1@example.com', 'playerpass', 'player1.jpg
', TO_DATE('2022-01-01', 'YYYY-MM-DD')), 5000, 100, TO_DATE('1990-01-01', 'YYYY-MM-DD'));
 2
 3
     INSERT INTO Player (Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture,
 4
          Player_JoinDate, Player_Salary, Player_Play_Hours, Player_DOB)
     VALUES (seq_player_id.NEXTVAL, 'Player 2', 'player2@example.com', 'playerpass', 'player2.jpg
', TO_DATE('2022-02-01', 'YYYY-MM-DD')), 6000, 200, TO_DATE('1992-05-10', 'YYYY-MM-DD'));
 5
 6
 7
     INSERT INTO Player (Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture,
          Player_JoinDate, Player_Salary, Player_Play_Hours, Player_DOB)
     VALUES (seq_player_id.NEXTVAL, 'Player 3', 'player3@example.com', 'playerpass', 'player3.jpg
', TO_DATE('2022-03-01', 'YYYY-MM-DD')), 7000, 150, TO_DATE('1994-09-20', 'YYYY-MM-DD'));
 8
 9
10
     INSERT INTO Player (Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture,
          Player_JoinDate, Player_Salary, Player_Play_Hours, Player_DOB)
     VALUES (seq_player_id.NEXTVAL, 'Player 4', 'player4@example.com', 'playerpass', 'player4.jpg
', TO_DATE('2022-04-01', 'YYYY-MM-DD'), 8000, 120, TO_DATE('1996-12-05', 'YYYY-MM-DD'));
11
12
     INSERT INTO Player (Player_ID, Player_Name, Player_Email, Player_Password, Player_Picture,
13
          Player_JoinDate, Player_Salary, Player_Play_Hours, Player_DOB)
     VALUES (seq_player_id.NEXTVAL, 'Player 5', 'player5@example.com', 'playerpass', 'player5.jpg
', TO_DATE('2022-05-01', 'YYYY-MM-DD'), 9000, 180, TO_DATE('1998-03-15', 'YYYY-MM-DD'));
14
```

Inserting data into Player tables



Inserted data of Player table

```
1
   INSERT INTO Player_Address (Pa_ID, Player_ID, Player_Country, Player_City, Player_Street,
       Player_Zip_Code)
2
   VALUES (seq_pa_id.NEXTVAL, 1, 'USA', 'New York', '123 Street', '10001');
3
   INSERT INTO Player_Address (Pa_ID, Player_ID, Player_Country, Player_City, Player_Street,
       Player_Zip_Code)
   VALUES (seq_pa_id.NEXTVAL, 2, 'USA', 'Los Angeles', '456 Avenue', '90001');
4
   INSERT INTO Player_Address (Pa_ID, Player_ID, Player_Country, Player_City, Player_Street,
5
       Player_Zip_Code)
6
   VALUES (seq_pa_id.NEXTVAL, 3, 'UK', 'London', '789 Road', 'SW1A 1AA');
   INSERT INTO Player_Address (Pa_ID, Player_ID, Player_Country, Player_City, Player_Street,
7
       Player_Zip_Code)
8
   VALUES (seq_pa_id.NEXTVAL, 4, 'Canada', 'Toronto', '321 Boulevard', 'M5V 2T3');
   INSERT INTO Player_Address (Pa_ID, Player_ID, Player_Country, Player_City, Player_Street,
9
       Player_Zip_Code)
   VALUES (seq_pa_id.NEXTVAL, 5, 'Germany', 'Berlin', '987 Strasse', '12345');
10
```

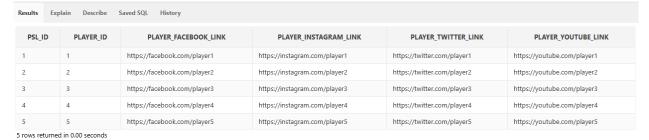
Inserting data into Player_Address tables

PLAYER_ID	PLAYER_COUNTRY	PLAYER_CITY	PLAYER_STREET	PLAYER_ZIP_CODE
	USA	New York	123 Street	10001
2	USA	Los Angeles	456 Avenue	90001
3	UK	London	789 Road	SW1A 1AA
ı	Canada	Toronto	321 Boulevard	M5V 2T3
5	Germany	Berlin	987 Strasse	12345
1	econds	UK Canada Germany	UK London Canada Toronto Germany Berlin	UK London 789 Road Canada Toronto 321 Boulevard Germany Berlin 987 Strasse

Inserted data of Player Address table

```
INSERT INTO PLAYER_SOCIAL_LINK (Psl_ID, Player_ID, Player_Facebook_Link,
       Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)
2
   VALUES (seq_psl_id.NEXTVAL, 1, 'https://facebook.com/player1', 'https://instagram.com/
       player1', 'https://twitter.com/player1', 'https://youtube.com/player1');
3
   INSERT INTO PLAYER_SOCIAL_LINK (Psl_ID, Player_ID, Player_Facebook_Link,
4
       Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)
5
   VALUES (seq_psl_id.NEXTVAL, 2, 'https://facebook.com/player2', 'https://instagram.com/
       player2', 'https://twitter.com/player2', 'https://youtube.com/player2');
6
7
   INSERT INTO PLAYER_SOCIAL_LINK (Psl_ID, Player_ID, Player_Facebook_Link,
       Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)
8
   VALUES (seq_psl_id.NEXTVAL, 3, 'https://facebook.com/player3', 'https://instagram.com/
       player3', 'https://twitter.com/player3', 'https://youtube.com/player3');
9
   INSERT INTO PLAYER_SOCIAL_LINK (Psl_ID, Player_ID, Player_Facebook_Link,
10
       Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)
   VALUES (seq_psl_id.NEXTVAL, 4, 'https://facebook.com/player4', 'https://instagram.com/
11
       player4', 'https://twitter.com/player4', 'https://youtube.com/player4');
12
   INSERT INTO PLAYER_SOCIAL_LINK (Psl_ID, Player_ID, Player_Facebook_Link,
13
       Player_Instagram_Link, Player_Twitter_Link, Player_Youtube_Link)
14
   VALUES (seq_psl_id.NEXTVAL, 5, 'https://facebook.com/player5', 'https://instagram.com/
       player5', 'https://twitter.com/player5', 'https://youtube.com/player5');
```

Inserting data into Player_SocialLink tables



Inserted data of Player Social Link table

```
1
   INSERT INTO Player_Phone (Pp_ID, Player_ID, Player_Phone)
2
   VALUES (seq_pp_id.NEXTVAL, 1, '1112223333');
4
   INSERT INTO Player_Phone (Pp_ID, Player_ID, Player_Phone)
   VALUES (seq_pp_id.NEXTVAL, 2, '2223334444');
5
7
   INSERT INTO Player_Phone (Pp_ID, Player_ID, Player_Phone)
8
   VALUES (seq_pp_id.NEXTVAL, 3, '3334445555');
9
10
   INSERT INTO Player_Phone (Pp_ID, Player_ID, Player_Phone)
11
   VALUES (seq_pp_id.NEXTVAL, 4, '4445556666');
12
   INSERT INTO Player_Phone (Pp_ID, Player_ID, Player_Phone)
13
14
   VALUES (seq_pp_id.NEXTVAL, 5, '5556667777');
```

Inserting data into Player_Phone tables

Results Explain Describe Saved SQL History				
PP_ID	PLAYER_ID	PLAYER_PHONE		
1	1	1112223333		
2	2	2223334444		
3	3	3334445555		
4	4	4445556666		
5	5	5556667777		
5 rows returned in 0.00 seconds				

Inserted data of Player Phone table

```
INSERT INTO Player_Winning (Pw_ID, Player_ID, Player_Winning)
   VALUES (seq_pw_id.NEXTVAL, 1, 'Tournament 2022');
4
   INSERT INTO Player_Winning (Pw_ID, Player_ID, Player_Winning)
   VALUES (seq_pw_id.NEXTVAL, 2, 'Championship 2023');
5
   INSERT INTO Player_Winning (Pw_ID, Player_ID, Player_Winning)
7
8
   VALUES (seq_pw_id.NEXTVAL, 3, 'Cup 2022');
10
   INSERT INTO Player_Winning (Pw_ID, Player_ID, Player_Winning)
11
   VALUES (seq_pw_id.NEXTVAL, 4, 'Tournament 2023');
12
13
   INSERT INTO Player_Winning (Pw_ID, Player_ID, Player_Winning)
   VALUES (seq_pw_id.NEXTVAL, 5, 'League 2022');
```

Inserting data into Player_Winning tables



5 rows returned in 0.00 seconds

Inserted data of Player Winning table

```
INSERT INTO Player_Team (Pt_ID, Player_ID, Team_ID)
1
2
   VALUES (seq_pt_id.NEXTVAL, 1, 1);
3
   INSERT INTO Player_Team (Pt_ID, Player_ID, Team_ID)
5
   VALUES (seq_pt_id.NEXTVAL, 2, 2);
6
7
   INSERT INTO Player_Team (Pt_ID, Player_ID, Team_ID)
8
   VALUES (seq_pt_id.NEXTVAL, 3, 3);
9
10
   INSERT INTO Player_Team (Pt_ID, Player_ID, Team_ID)
11
   VALUES (seq_pt_id.NEXTVAL, 4, 4);
12
13
   INSERT INTO Player_Team (Pt_ID, Player_ID, Team_ID)
   VALUES (seq_pt_id.NEXTVAL, 5, 5);
14
```

Inserting data into Player_Team tables



5 rows returned in 0.00 seconds

Inserted data of Player Team table

5.6 Single Row Functions

• Retrieve the email domain for the admin with Admin_ID = 1

```
SELECT Admin_Email, SUBSTR(Admin_Email, INSTR(Admin_Email, '@') + 1) AS Email_Domain
FROM Admin
WHERE Admin_ID = 1;
```

Query 1



Result of Query 1

• Get the hire date of the manager named 'John Doe' formatted in a specific way

```
SELECT Manager_Name, TO_CHAR(Manager_Hiredate, 'DD-Mon-YYYY') AS Hire_Date
FROM Manager
WHERE Manager_Name = 'John Doe';
```

Query 2



Result of Query 2

• Concatenate the first and last name of the content creator with ContentCreator_ID = 1

Query 3



Result of Query 3

5.7 Group Functions

• Calculate the average balance for all finance records

```
SELECT AVG(Finance_Balance) AS Average_Balance
FROM Finance;
```

Query 1



Result of Query 1

• Count the number of teams established in each country

```
SELECT Team_Country, COUNT(*) AS Team_Count
FROM Team
GROUP BY Team_Country;
```

Query 2



Result of Query 2

• Calculate the total salary expense for content creators

```
SELECT SUM(ContentCreator_Salary) AS Total_Salary_Expense FROM ContentCreator;
```

Query 3



Result of Query 3

5.8 SubQuery

• Retrieve the managers associated with teams established before a specific date

```
SELECT Manager_Name
FROM Manager
WHERE Manager_ID IN (
SELECT Manager_ID
FROM Team
WHERE Team_Established_Date < TO_DATE('2022-03-01', 'YYYY-MM-DD')

7 );
```

Query 1



Result of Query 1

• Get the content creators who have a higher salary than the average salary of all content creators

```
SELECT ContentCreator_Name
FROM ContentCreator

WHERE ContentCreator_Salary > (
SELECT AVG(ContentCreator_Salary)
FROM ContentCreator

);
```

Query 2



Result of Query 2

• Retrieve the teams managed by managers who have won a championship

```
SELECT Team_Name
1
2
         FROM Team
3
         WHERE Manager_ID IN (
4
         SELECT Manager_ID
         FROM Manager
WHERE Manager_ID IN (
5
6
7
              SELECT DISTINCT Manager_ID
             FROM Team_Winning
WHERE Team_Winning LIKE '%Championship%'
8
9
10
11
    );
```

Query 3



Result of Query 3

5.9 Join Queries

• Get the team name and manager name for each team

```
SELECT t.Team_Name, m.Manager_Name
FROM Team t
JOIN Manager m ON t.Manager_ID = m.Manager_ID;
```

Query 1



Result of Query 1

• Retrieve the player name, team name, and country for each player

```
SELECT p.Player_Name, t.Team_Name, t.Team_Country
FROM Player p
JOIN Player_Team pt ON p.Player_ID = pt.Player_ID
JOIN Team t ON pt.Team_ID = t.Team_ID;
```

Query 2



Result of Query 2

• Get the content creator name, social media name, and email for each content creator

```
SELECT cc.ContentCreator_Name, sm.SocialMedia_Name, sm.SocialMedia_Email
FROM ContentCreator cc
JOIN SocialMedia sm ON cc.SOCIALMEDIA_ID = sm.SocialMedia_ID;
```

Query 3

Results Explain Describe Saved SQL History		
CONTENTCREATOR_NAME	SOCIALMEDIA_NAME	SOCIALMEDIA_EMAIL
ContentCreator 1	SocialMediaUser221	social.user1@example.com
ContentCreator 2	SocialMediaUser442	social.user2@example.com
ContentCreator 3	SocialMediaUser423	social.user3@example.com
ContentCreator 4	SocialMediaUser4234	social.user4@example.com
ContentCreator 5	SocialMediaUser5523	social.user5@example.com
5 rows returned in 0.01 seconds Download		

Result of Query 3

5.10 Creating View

• Create a view to display the details of managers and their associated teams

```
CREATE VIEW ManagerTeamView AS
SELECT m.Manager_Name, t.Team_Name, t.Team_Country
FROM Manager m
JOIN Team t ON m.Manager_ID = t.Manager_ID;
```

ManagerTeamView



Result of ManagerTeamView

• Create a view to show the average salary of content creators

```
CREATE VIEW AverageSalaryView AS
SELECT AVG(ContentCreator_Salary) AS Average_Salary
FROM ContentCreator;
```

AvgSalaryView



Result of AverageSalaryView

• Create a view to list the players and their corresponding teams

```
CREATE VIEW PlayerTeamView AS

JOIN Team t ON pt.Team_ID = t.Team_ID;

SELECT p.Player_Name, t.Team_Name

FROM Player p

JOIN Player_Team pt ON p.Player_ID = pt.Player_ID
```

PlayerTeamView

Results	Explain Describe	Saved SQL	History	
		PLAYER_NA	ME	TEAM_NAME
Player 1				Team A
Player 2				Team B
Player 3				Team C
Player 4				Team D
Player 5				Team E
5 rows ret	urned in 0.00 seconds	Download		

Result of PlayerTeamView

5.11 Synonyms

```
-- Create synonym for the ORGANIZATION_TOURNAMENT table
CREATE SYNONYM org_tour FOR Organization_Tournament;

-- Create synonym for the CONTENTCREATOR_SOCIALMEDIA table
CREATE SYNONYM cc_sm FOR ContentCreator_SocialMedia;

-- Create synonym for the CONTENTCREATOR_PHONE table
CREATE SYNONYM cc_ph FOR ContentCreator_Phone;
```

Synonyms

Results Explain Describe Saved SQL History				
OWNER	SYNONYM_NAME	TABLE_OWNER	TABLE_NAME	DB_LINK
ESPORTFTW	CC_PH	ESPORTFTW	CONTENTCREATOR_PHONE	-
ESPORTFTW	CC_SM	ESPORTFTW	CONTENTCREATOR_SOCIALMEDIA	-
ESPORTFTW	ORG_TOUR	ESPORTFTW	ORGANIZATION_TOURNAMENT	-
3 rows returned in 1.38 seconds Download				

List of Synonyms

Relational Algebra

1. Find the name of the manager whose manager id is 5.

$$\Pi_{\text{Manager_Name}}(\sigma_{\text{Manager_ID}=5(\text{Manager})})$$

2. Find the Salary of 'Player 1'.

$$\Pi_{\text{Player_Salary}}(\sigma_{\text{name}=\text{"Player 1"(Player)}})$$

3. Find Player id whose birthday is on 1992-05-10.

$$\Pi_{\text{Player_ID}}(\sigma_{\text{Player_DOB}="1992-05-10"}(\text{Player}))$$

4. Find the Country, City, Street, and Zip code where Content Creator ID is equal to 4.

$$\Pi_{ContentCreator_Country,\ ContentCreator_City,ContentCreator_Street,\ ContentCreator_Zip_Code}$$

$$\left(\sigma_{ContentCreator_ID=4(ContentCreator_Address)}\right)$$

5. Find the Team ID that won the Championship 2023.

$$\Pi_{\text{Team_ID}}(\sigma_{\text{Team_Winning}} = \text{"Championship 2023"}(\text{Team_Winning}))$$

Conclusion

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.