SAVAGE

The next best MOBA developed by Q&A, Inc.

**Summary**

With the popularity Multiplayer Online Battle Arena (MOBA) games, Q&A, Inc. (Q&A) has been developing their first MOBA called Savage, to launch in the fourth quarter of this year. Q&A conducted a benchmark study and found that the most popular and successful MOBAs have very well-structured databases to handle the enormous amount of data. To meet the timeline of the game launch, a cross functional team will use entities and relationships to develop a conceptual and logical model for their database. By the end of the first quarter the cross functional team will at a minimum define and present the following to the stakeholders: the business rules, a glossary for the data, and a set of questions we expect the database to answer. At the conclusion of the presentation the stakeholders will need to approve the models in order for the team to begin the process of implementing the database.

**Stakeholders**

* CEO and Upper Management of Q&A, Inc.
* Online Gamers
* Software Engineers
* Marketing Firm
* Investors
* Cloud Provider
* Identity Protection Consultant
* Entertainment Software Rating Board

**Glossary**

A **gamer** is an individual that access and plays the MOBA.

A **game character** is one of many selectable characters in the MOBA that a Gamer can play as.

MOBA has three **game modes** that gamers can choose to play in with a team: Gladiator, Skirmish, or Domination.

* **Gladiator** is a five versus five Team battle to eliminate all the enemies of the opposing team with no time limit.
* **Skirmish** is a three versus three Team battle to capture the opposing team’s headquarters in a small defined map with no time limit.
* **Domination** is a four versus four Team battle to capture the opposing team’s headquarters in lager defined map with no time limit.

**Game equipment** may or may not be equipped on a game character during game play, and the game equipment is either offensive or defensive.

* **Offensive** is equipment that a gamer can equip on their game character that can improve any of their offensive stats: Power, Penetration, or Speed.
* **Defensive** is equipment that a gamer can equip on their game character that can improve any of their defensive stats: Physical Protection, Magical Protection, or CCR Protection.

A **sponsor** is an organization that chooses to make a monetary contribution to a gamer and/or a tournament in exchange for promoting the sponsors product or service.

The **sponsor address** is the address where the sponsor resides.

A **in game purchase** is a monetary transaction a gamer can make with a valid credit card to acquire new in game content released by Q&A, Inc. if they want to.

A **tournament** is a sponsored event where multiple gamers compete in the Domination mode for a cash prize.

**Business Rules**

* It is required that a gamer have a unique user ID and email address.
* The gamer must choose one game character from the many game characters before a match can start.
* The gamer must choose one of the three game modes to play a game.
* The gamer may or may not choose to equip their game character with one or many offensive equipment.
* The gamer may or may not choose to equip their game character with one or many defensive equipment.
* A gamer may or may not purchase additional in-game content.
* A valid credit card is required for the purchase of in-game content.
* A tournament can only have one team as a winner, no ties.
* A tournament must have a least one sponsor.
* The Domination game mode is the only mode used for a tournament.
* A sponsor must have a street address and not a Post Office box address.

**Conceptual Model**

Diagram, schematic

Description automatically generated

**Logical Model**

Diagram

Description automatically generated

To validate the MOBA database will meet the expectations of the stakeholders, the cross functional team constructed a set of questions out of the conversations they had with our stakeholders. Below are five questions that we expect to answer from the MOBA database:

1. What has been the most and least popular game character over the last 10 tournaments?
2. What are the top three grossing in-game purchases at the end of each quarter?
3. What is the percentage break down of a game character class played by a specific gamer?
4. What is the most selected offensive and defensive item for a particular game character?
5. What is the probability a specific gamer will win the tournament?

**Raw Data Sample**

The Q&A software team released the beta version of Savage to a small group of Gamers for testing in the second Quarter. Below is a sample of the raw data that will be collected by the database.

Graphical user interface

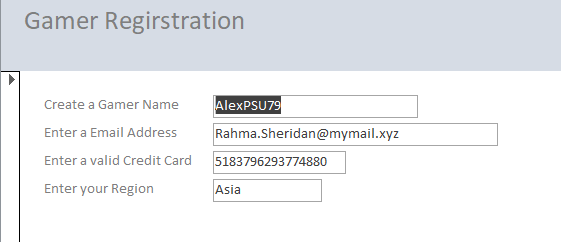
Description automatically generated

A picture containing text, indoor, vending machine

Description automatically generated

**Savage Physical Database**

The New Gamer Registration form created in Access will be used to log the Gamers information for the Savage Beta release.



-- Drop NewGamer procedure

DROP PROCEDURE IF EXISTS dbo.NewGamer

GO

-- Drop CreditCardUpdate procedure

DROP PROCEDURE IF EXISTS dbo.CreditCardUpdate

GO

--Drop Tables

DROP TABLE IF EXISTS GamerGameEquipment

GO

DROP TABLE IF EXISTS GamerSponsor

GO

DROP TABLE IF EXISTS InGamePurchaseGamer

GO

DROP TABLE IF EXISTS TournamentSponsor

GO

DROP TABLE IF EXISTS CharacterGameMode

GO

DROP TABLE IF EXISTS GamerTournament

GO

DROP TABLE IF EXISTS SponsorAddress

GO

DROP TABLE IF EXISTS Sponsor

GO

DROP TABLE IF EXISTS Defensive

GO

DROP TABLE IF EXISTS Offensive

GO

DROP TABLE IF EXISTS GameEquipment

GO

DROP TABLE IF EXISTS InGamePurchase

GO

DROP TABLE IF EXISTS GameCharacter

GO

DROP TABLE IF EXISTS Tournament

GO

DROP TABLE IF EXISTS GameMode

GO

DROP TABLE IF EXISTS Gamer

GO

--Drop all views

DROP VIEW IF EXISTS CharacterBreakDown

GO

DROP VIEW IF EXISTS QuarterSales

GO

--Drop all my function statements

DROP FUNCTION IF EXISTS dbo.Warrior

GO

DROP FUNCTION IF EXISTS dbo.Assassin

GO

DROP FUNCTION IF EXISTS dbo.Mage

GO

DROP FUNCTION IF EXISTS dbo.Hunter

GO

DROP FUNCTION IF EXISTS dbo.Guardian

GO

DROP FUNCTION IF EXISTS dbo.WinProbability

GO

-- Create Gamer table

CREATE TABLE Gamer(

-- Columns in Gamer table

GamerID int identity,

GamerName varchar (25) not null,

EmailAddress nchar(35) not null,

CreditCard varchar(16),

Region char(13) not null,

-- Constraints in Gamer table

CONSTRAINT PK1\_GamerID PRIMARY KEY (GamerID),

CONSTRAINT U1\_GamerName UNIQUE (GamerName),

CONSTRAINT U2\_EmailAddress UNIQUE (EmailAddress),

CONSTRAINT U3\_CreditCard UNIQUE (CreditCard)

)

GO

-- Create Game Mode table

CREATE TABLE GameMode(

-- Columns in Game Mode table

GameModeID int identity,

GameType char(10) not null,

StartTime smalldatetime not null,

EndTime smalldatetime not null,

Wins bit not null,

Losses bit not null,

KillCount int not null,

AssistCount int not null,

DeathCount int not null,

-- Constraint in Game Mode table

CONSTRAINT PK1\_GameModeID PRIMARY KEY (GameModeID)

)

GO

-- Create Tournament table

CREATE TABLE Tournament(

-- Column in Tournament table

TournamentID int identity,

CashPrize decimal (9,2) not null,

GameModeID int not null,

-- Constraint in Tournament table

CONSTRAINT PK1\_Tournament PRIMARY KEY (TournamentID),

CONSTRAINT FK1\_Tournament FOREIGN KEY (GameModeID) REFERENCES GameMode(GameModeID)

)

GO

-- Create Game Character table

CREATE TABLE GameCharacter(

-- Columns in Game Character table

CharacterID int identity,

CharacterName char(35) not null,

Class char(8)

-- Contraints in Game Character table

CONSTRAINT PK1\_CharacterID PRIMARY KEY (CharacterID),

CONSTRAINT U1\_CharacterName UNIQUE (CharacterName),

)

GO

-- Create In Game Purchase table

CREATE TABLE InGamePurchase (

-- Columns in In Game Purcahse table

PurchaseID int identity,

PurchaseDescription char(50) not null,

Cost decimal (9,2) not null,

PurchaseDate datetime not null,

-- Constraint in In Game Purchase table

CONSTRAINT PK1\_PurchaseID PRIMARY KEY (PurchaseID)

)

GO

-- Create Game Equipment table

CREATE TABLE GameEquipment (

-- Columns in Game Equipment table

EquipmentID int identity,

EquipmentName char(50) not null,

ItemType char (9) not null,

Passive varchar(255)

-- Constraint in Game Equipment table

CONSTRAINT PK1\_EquipmentID PRIMARY KEY (EquipmentID),

CONSTRAINT U1\_EquipmentName UNIQUE (EquipmentName)

)

GO

-- Create Offensive table

CREATE TABLE Offensive(

-- Columns in Offensive table

EquipmentID int,

EquipmentPower int,

Penetration int,

Speed int,

-- Constraint in Offensive table

CONSTRAINT PK2\_EquipmentID PRIMARY KEY (EquipmentID),

CONSTRAINT FK1\_EquipmentID FOREIGN KEY (EquipmentID) REFERENCES GameEquipment(EquipmentID)

)

GO

-- Create Defensive table

CREATE TABLE Defensive(

-- Columns in Defensive table

EquipmentID int,

PhysicalProtection int,

MagicalProtection int,

CCRProtection int,

-- Contraint in Defensive table

CONSTRAINT PK3\_EquipmentID PRIMARY KEY (EquipmentID),

CONSTRAINT FK1\_Defensive FOREIGN KEY (EquipmentID) REFERENCES GameEquipment(EquipmentID)

)

GO

-- Create Sponsor table

CREATE TABLE Sponsor(

--Columns in Sponsor table

SponsorID int identity,

SponsorName nchar(50) not null,

Contribution decimal (9,2) not null,

--Constraint in Sponsor table

CONSTRAINT PK1\_SponsorID PRIMARY KEY (SponsorID)

)

GO

-- Create Sponsor Address table

CREATE TABLE SponsorAddress(

-- Columns in Sponsor Address table

SponsorAddressID int identity,

SponsorID int,

StreetAddress varchar(35) not null,

City nchar(35) not null,

AddressState nchar(35) not null,

Country nchar(25) not null,

-- Constraints in Sponsor Address table

CONSTRAINT PK1\_SponsorAddressID PRIMARY KEY (SponsorAddressID),

CONSTRAINT FK1\_SponsorAddress FOREIGN KEY (SponsorID) REFERENCES Sponsor(SponsorID)

)

GO

-- Create Gamer Tournament table

CREATE TABLE GamerTournament(

-- Column in Game Tournament table

GamerTournamentID int identity,

GamerID int not null,

TournamentID int not null,

-- Constraint in Game Tournament table

CONSTRAINT PK1\_GamerTournamentID PRIMARY KEY (GamerTournamentID),

CONSTRAINT FK1\_GamerTournament FOREIGN KEY (GamerID) REFERENCES Gamer(GamerID),

CONSTRAINT FK2\_GamerTournament FOREIGN KEY (TournamentID) REFERENCES Tournament(TournamentID)

)

GO

-- Create Character Game Mode table

CREATE TABLE CharacterGameMode (

-- Columns in Character Game Mode table

CharacterGameModeID int identity,

CharacterID int not null,

GameModeID int not null,

-- Contraints in Character Game Mode table

CONSTRAINT PK1\_CharacterGameMode PRIMARY KEY (CharacterGameModeID),

CONSTRAINT FK1\_CharacterGameMode FOREIGN KEY (CharacterID) REFERENCES GameCharacter(CharacterID),

CONSTRAINT FK2\_CharacterGameMode FOREIGN KEY (GameModeID) REFERENCES GameMode(GamemodeID)

)

GO

-- Create Tournament Sponsor table

CREATE TABLE TournamentSponsor(

-- Columns in Tournament Sponsor table

TournamentSponsorID int identity,

TournamentID int not null,

SponsorID int not null,

-- Constraint in Tournament Sponsor table

CONSTRAINT PK1\_TournamentSponsorID PRIMARY KEY (TournamentSponsorID),

CONSTRAINT FK1\_TournamentSponsor FOREIGN KEY (TournamentID) REFERENCES Tournament(TournamentID),

CONSTRAINT FK2\_TournamentSponsor FOREIGN KEY (SponsorID) REFERENCES Sponsor(SponsorID)

)

GO

-- Create In Game Purchase Gamer table

CREATE TABLE InGamePurchaseGamer(

-- Columns in In Game Purchase Gamer table

InGamePurchaseGamerID int identity,

GamerID int not null,

PurchaseID int not null,

-- Constraint in In Game Purchase Gamer table

CONSTRAINT PK1\_InGamePurchaseGamerID PRIMARY KEY (InGamePurchaseGamerID),

CONSTRAINT FK1\_InGamePurchaseGamer FOREIGN KEY (GamerID) REFERENCES Gamer(GamerID),

CONSTRAINT FK2\_InGamePurchaseGamer FOREIGN KEY (PurchaseID) REFERENCES InGamePurchase(PurchaseID)

)

GO

-- Create Gamer Sponsor table

CREATE TABLE GamerSponsor(

--Columns in Gamer Sponsor table

GamerSponsorID int identity,

GamerID int not null,

SponsorID int not null,

-- Constraints in Gamer Sponsor table

CONSTRAINT PK1\_GamerSponsorID PRIMARY KEY (GamerSponsorID),

CONSTRAINT FK1\_GamerSponsor FOREIGN KEY (GamerID) REFERENCES Gamer(GamerID),

CONSTRAINT FK2\_GamerSponsor FOREIGN KEY (SponsorID) REFERENCES Sponsor(SponsorID)

)

GO

-- Create Gamer Game Equipment table

CREATE TABLE GamerGameEquipment (

-- Columns in Gamer Game Equipment table

GamerGameEquipmentID int identity,

EquipmentID int not null,

GamerID int not null,

-- Constraints in Gamer Game Equipment table

CONSTRAINT PK1\_GamerGameEquipmentID PRIMARY KEY (GamerGameEquipmentID),

CONSTRAINT FK1\_GamerGameEquipment FOREIGN KEY (EquipmentID) REFERENCES GameEquipment(EquipmentID),

CONSTRAINT FK2\_GamerGameEquipment FOREIGN KEY (GamerID) REFERENCES Gamer(GamerID)

)

GO

-- Insert data into Gamer table

INSERT INTO Gamer(GamerName, EmailAddress, CreditCard, Region)

VALUES ('LinesofChaos', 'Efe.Luna@mymail.xy', '5581436088576110', 'North America'),

('John2Tymes', 'Sol.Smyth@mymail.xyz', '5236609431366340', 'South America'),

('AlexPSU79', 'Rahma.Sheridan@mymail.xyz', '5183796293774880', 'Asia'),

('King\_Pain', 'Braydon.Wagner@mymail.xyz', '5200136813258580', 'North America'),

('VG Chewy', 'Luke.Cox@mymail.xyz', '5111825775276190', 'North America'),

('Iambigdamage', 'Yvette.Kline@mymail.xyz', '4556363084412060', 'Asia'),

('ItsMrJoker', 'Daniele.Frost@mymail.xyz', '4929245366652930', 'Asia'),

('BigRed748', 'Kathryn.Monaghan@mymail.xyz', '4024007172191520', 'Africa'),

('Don FunnyMan', 'Hollie.Bate@mymail.xyz', '4929616554375540', 'North America'),

('Drop\_Dead24', 'Jadine.Kaye@mymail.xyz', '4485727116748420', 'South America'),

('iamcoyson', 'Jake.Johns@mymail.xyz', '346776571462041', 'North America'),

('kid 4 show', 'Aadam.Mclellan@mymail.xyz', '346493544223415', 'Africa'),

('NightNija21', 'Mehdi.Blankenship@mymail.xyz', '344942544485178', 'Europe'),

('Platano4Life', 'Miguel.Carr@mymail.xyz', '376788231100732', 'Asia'),

('SnooBen21', 'Huw.Rosas@mymail.xyz', '344768498204061', 'Africa')

GO

-- Read rows from Gamer Table

SELECT \* FROM Gamer

GO

-- Insert data into Game Character table

INSERT INTO GameCharacter (CharacterName, Class)

VALUES ('Zeus', 'Guardian'), ('Wolverine', 'Hunter'), ('Cyclops','Mage'),

('Frodo', 'Guardian'), ('Loki', 'Mage'), ('Deadpool', 'Assassin'),

('Luffy', 'Assassin'), ('Naurato', 'Assassin'), ('Sanji', 'Warrior'),

('Hera', 'Warrior'), ('Nike', 'Assassin'), ('Zorro', 'Guardian'),

('Hulk','Hunter'), ('Sonic', 'Assassin'), ('Donkey Kong', 'Hunter')

GO

-- Read rows from Game Character table

SELECT \* FROM GameCharacter

GO

-- Insert data into Sponsor table

INSERT INTO Sponsor (SponsorName, Contribution)

VALUES ('Eco Focus', 2458183.00),('Innovation Arch', 952541.00), ('Strat Security', 713497.00),

('Inspire Fitness Co', 833449.00), ('Candor Corp', 1286274.00)

GO

-- Read rows in Sponsor table

SELECT \* FROM Sponsor

GO

-- Insert data into SponsorAddress (SELECT SponsorID FROM Sponsor WHERE SponsorName = 'Admire Arts')

INSERT INTO SponsorAddress (SponsorID, StreetAddress, City, AddressState, Country)

VALUES ((SELECT SponsorID FROM Sponsor WHERE SponsorName = 'Eco Focus'),'Morsbroicher Street. 76b', 'Nord Ecrinstadt', 'Mecklenburg-Vorpommern', 'Germany'),

((SELECT SponsorID FROM Sponsor WHERE SponsorName = 'Innovation Arch'),'Von-Knoeringen-Street. 71a', 'Pohlandstadt', 'Mecklenburg-Vorpommern', 'Germany'),

((SELECT SponsorID FROM Sponsor WHERE SponsorName = 'Strat Security'),'Contrada Nicodemo 33', 'Keyaish', 'Avellino', 'Italy'),

((SELECT SponsorID FROM Sponsor WHERE SponsorName = 'Inspire Fitness Co'),'76105 Olson Street', 'Kirstenshire', 'Pennsylvania', 'United States'),

((SELECT SponsorID FROM Sponsor WHERE SponsorName = 'Candor Corp'),'Joeykade 043b', 'Tjoenebroek', 'Overijssel', 'Netherlands')

GO

-- Read rows in SponsorAddress table

SELECT \* FROM SponsorAddress

GO

-- Insert data into In Game Purchase table

INSERT INTO InGamePurchase (PurchaseDescription, Cost, PurchaseDate)

VALUES('Gem Sale', 12.00, '4/16/2020 04:05:12'), ('Gem Sale', 18.00, '6/23/2020 14:29:21'),

('Character Skin', 8.00, '5/5/2020 02:45:22'), ('Gift Card', 11.00, '4/24/2020 17:02:34'),

('Gem Sale', 25.00, '4/5/2020 08:50:21'), ('Avatar Items', 7.00, '5/23/2020 19:49:35'),

('Expansion Pack Sale', 10.00, '5/24/2020 15:56:09'), ('Expansion Pack Sale', 11.00, '9/3/2020 03:46:35'),

('New Map', 12.00, '9/15/2020 22:38:38'), ('New Map', 14.00, '8/16/2020 01:06:34'),

('Expansion Pack Sale', 16.00, '8/8/2020 23:35:29'), ('Avatar Items', 16.00, '7/13/2020 17:45:53'),

('Avatar Items', 24.00, '8/9/2020 00:39:41'), ('New Map', 6.00, '8/24/2020 23:52:32'),

('Character Skin', 22.00, '12/10/2020 12:15:56'), ('New Game Character', 15.00, '12/17/2020 04:36:25'),

('Character Skin', 8.00, '12/28/2020 16:12:57'), ('Expansion Pack Sale', 6.00, '12/2/2020 20:47:40'),

('New Game Character', 12.00, '11/2/2020 10:11:37'),('Gift Card', 12.00, '12/13/2020 10:33:08'),

('Character Skin', 6.00, '11/28/2020 00:45:54')

GO

-- Read rows in In Game Purchase table

SELECT \* FROM InGamePurchase

GO

-- Insert data into GameMode

INSERT INTO GameMode(GameType, Wins, Losses, StartTime, EndTime, KillCount, AssistCount, DeathCount)

VALUES('Skirmish', 'TRUE', 'FALSE', '2/3/2021 6:19:10 PM', '2/3/2021 6:51:10 PM', 9, 22, 10),

('Domination', 'FALSE', 'TRUE','12/31/2020 2:43:02 AM', '12/31/2020 3:21:02 AM', 2, 8, 6),

('Gladiator', 'TRUE', 'FALSE', '11/28/2020 3:31:15 AM', '11/28/2020 3:47:15 AM', 10, 10, 24),

('Gladiator', 'TRUE', 'FALSE', '10/17/2020 4:25:50 AM', '10/17/2020 5:00:50 AM', 25, 3, 13),

('Gladiator', 'FALSE', 'TRUE', '2/13/2021 8:12:59 AM', '2/13/2021 8:34:59 AM', 17, 22, 25),

('Gladiator', 'TRUE', 'FALSE', '12/20/2020 1:56:48 PM', '12/20/2020 2:23:48 PM', 6, 3, 20),

('Domination', 'TRUE', 'FALSE', '11/9/2020 12:28:44 AM', '11/9/2020 1:11:44 AM', 13, 17, 1),

('Skirmish', 'TRUE', 'FALSE', '10/12/2020 3:26:04 PM', '10/12/2020 4:09:04 PM', 2, 13, 17),

('Skirmish', 'TRUE', 'FALSE', '9/1/2020 4:16:53 PM', '9/1/2020 4:40:53 PM', 12, 2, 10),

('Domination', 'FALSE', 'TRUE', '10/10/2020 2:34:56 PM', '10/10/2020 2:52:56 PM', 14, 12, 14),

('Skirmish', 'TRUE', 'FALSE', '11/10/2020 2:08:06 PM', '11/10/2020 2:50:06 PM', 6, 5, 16),

('Skirmish', 'TRUE', 'FALSE', '11/10/2020 9:11:24 PM', '11/10/2020 9:49:24 PM', 20, 5, 6),

('Domination', 'FALSE', 'TRUE', '1/11/2021 1:10:16 AM', '1/11/2021 1:31:16 AM', 18, 15, 0),

('Domination', 'TRUE', 'FALSE', '12/19/2020 5:59:35 PM', '12/19/2020 6:25:35 PM', 14, 17, 13),

('Domination', 'TRUE', 'FALSE', '12/3/2020 11:09:23 PM', '12/3/2020 11:40:23 PM', 2, 19, 15),

('Gladiator', 'FALSE', 'TRUE', '1/16/2021 11:26:06 AM', '1/16/2021 12:09:06 PM', 17, 6, 13),

('Domination', 'TRUE', 'FALSE','1/26/2021 4:15:18 PM', '1/26/2021 4:31:18 PM', 25, 7, 19),

('Skirmish', 'TRUE', 'FALSE', '2/9/2021 12:42:06 AM', '2/9/2021 1:08:06 AM', 10, 20, 21),

('Gladiator', 'FALSE', 'TRUE', '11/1/2020 9:07:11 AM', '11/1/2020 9:39:11 AM', 20, 25, 4),

('Skirmish', 'FALSE', 'TRUE', '12/26/2020 3:56:17 PM', '12/26/2020 4:35:17 PM', 1, 7, 7),

('Domination', 'TRUE', 'FALSE', '9/14/2020 4:41:33 AM', '9/14/2020 5:17:33 AM', 4, 2, 21),

('Domination', 'FALSE', 'TRUE', '10/17/2020 5:59:32 AM', '10/17/2020 6:15:32 AM', 21, 19, 12),

('Skirmish', 'TRUE', 'FALSE', '10/10/2020 11:08:00 PM', '10/10/2020 11:40:00 PM', 20, 6, 2),

('Domination', 'FALSE', 'TRUE', '2/27/2021 6:12:10 AM', '2/27/2021 6:35:10 AM', 20, 6, 7),

('Skirmish', 'TRUE', 'FALSE', '1/6/2021 3:56:45 AM', '1/6/2021 4:29:45 AM', 23, 15, 3),

('Gladiator', 'TRUE', 'FALSE', '10/31/2020 1:42:33 AM', '10/31/2020 2:14:33 AM', 5, 8, 8),

('Gladiator', 'FALSE', 'TRUE', '1/10/2021 11:08:41 PM', '1/10/2021 11:35:41 PM', 19, 11, 17),

('Gladiator', 'FALSE', 'TRUE', '2/26/2021 6:14:43 AM', '2/26/2021 6:42:43 AM', 5, 16, 5),

('Domination', 'FALSE', 'TRUE', '1/24/2021 7:46:53 PM', '1/24/2021 8:22:53 PM', 0, 20, 14),

('Domination', 'TRUE', 'FALSE', '10/31/2020 7:38:11 AM', '10/31/2020 8:16:11 AM', 5, 18, 2),

('Domination', 'TRUE', 'FALSE', '11/20/2020 4:47:18 AM', '11/20/2020 5:11:18 AM', 12, 24, 1),

('Domination', 'TRUE', 'FALSE', '12/17/2020 1:22:05 AM', '12/17/2020 2:05:05 AM', 7 ,13 ,6),

('Domination', 'FALSE', 'TRUE', '1/8/2021 10:58:38 PM', '1/8/2021 11:33:38 PM', 0, 7, 10),

('Domination', 'TRUE', 'FALSE', '1/29/2021 9:14:02 PM', '1/29/2021 9:53:02 PM', 9, 6, 8),

('Domination', 'TRUE', 'FALSE', '2/21/2021 9:41:31 PM', '2/21/2021 10:17:31 PM', 18 ,20, 18),

('Domination', 'TRUE', 'FALSE', '12/21/2020 7:41:17 PM', '12/21/2020 8:13:17 PM', 1,11,21),

('Domination', 'FALSE', 'TRUE', '11/1/2020 8:51:57 PM', '11/1/2020 9:26:57 PM', 11 ,11, 24),

('Domination', 'FALSE', 'TRUE', '11/28/2020 6:55:35 AM', '11/28/2020 7:40:35 AM', 25, 4, 19),

('Domination','FALSE', 'TRUE', '10/22/2020 7:26:10 AM', '10/22/2020 8:07:10 AM',17,12,6),

('Domination','FALSE', 'TRUE', '10/14/2020 11:24:28 PM', '10/14/2020 11:40:28 PM', 6, 17, 15)

GO

-- Read rows in Game Mode table

SELECT \* FROM GameMode

GO

-- Insert data into Tournament table

INSERT INTO Tournament (CashPrize,GameModeID)

VALUES(1600529.00,2), (131875.00,7), (1810528.00,10),(2678057.00,13),(354010.00,14),(2458552.00,15),

(724362.00,17),(1763711.00,21),(120400.00,22),(1469401.00,24),(290043.00,29),(2512511.00,30),(2070842.00,31),

(903150.00,32),(2844248.00,33),(173610.00,34),(195665.00,35),(170703.00,36),(2793399.00,37),(1840479.00,38)

GO

-- Read rows in Tournament table

SELECT \* FROM Tournament

GO

-- Insert data into Game Equipment table

INSERT INTO GameEquipment (EquipmentName, ItemType, Passive)

VALUES ('Rod of Smite', 'Offensive', 'Gives the power of smiting your opponent in one hit'),

('Book of Knowledge', 'Offensive', 'Gives you the knowledge to avoid Smite attacks'),

('Hammer of Dawn', 'Defensive', 'Bings down the hammer on the one you target'),

('Boots of Nike', 'Defensive', 'Dunk like Mike'),

('Torque Bow', 'Offensive', 'Improves your snipping ability'),

('Bloodforge', 'Offensive', 'When you need more health just kill a teammate'),

('Vail of Dispair', 'Defensive', 'You think you have it all together, but you do not'),

('Cursed Onk', 'Defensive', 'Oh no, you have cursed your best friend'),

('Shogun Blessing', 'Offensive', 'Gives you the blessing to do whatever you want for 30 seconds'),

('Epic Bow', 'Offensive', 'Slightly better then a Torque Bow')

GO

-- Read rows in Game Equipement table

SELECT \* FROM GameEquipment

GO

-- Insert data into Defensive table

INSERT INTO Defensive (EquipmentID, PhysicalProtection, MagicalProtection, CCRProtection)

VALUES(2,20,25,23),(4,8,null,8),(6,11,19,null ),(8,null,1,16),(10,6,21,10)

GO

-- Read rows in Defensive table

SELECT \* FROM Defensive

GO

-- Insert data into Offensive table

INSERT INTO Offensive(EquipmentID,EquipmentPower, Penetration,Speed)

VALUES(1,12,17,9),(3,10,17,null),(5,20,null,8 ),(7,13,12,null),(9,12,null,13)

GO

-- Read rows in Offensive table

SELECT \* FROM Offensive

GO

-- Insert data into GamerGamerEquipment table

INSERT INTO GamerGameEquipment (EquipmentID, GamerID)

VALUES ((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Rod of Smite'),(SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Rod of Smite'),(SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Rod of Smite'),(SELECT GamerID FROM Gamer WHERE GamerName = 'AlexPSU79')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Bloodforge'),(SELECT GamerID FROM Gamer WHERE GamerName = 'AlexPSU79')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Bloodforge'),(SELECT GamerID FROM Gamer WHERE GamerName = 'LinesofChaos')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Epic Bow'), (SELECT GamerID FROM Gamer WHERE GamerName = 'LinesofChaos')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Hammer of Dawn'),(SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Hammer of Dawn'),(SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Hammer of Dawn'),(SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Cursed Onk'),(SELECT GamerID FROM Gamer WHERE GamerName = 'AlexPSU79')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Cursed Onk'),(SELECT GamerID FROM Gamer WHERE GamerName = 'LinesofChaos')),

((SELECT EquipmentID FROM GameEquipment WHERE EquipmentName = 'Vail of Dispair'),(SELECT GamerID FROM Gamer WHERE GamerName = 'LinesofChaos'))

GO

-- Read rows in GamerGamerEquipment table

SELECT \* FROM GamerGameEquipment

GO

-- Insert data into GamerTournament table

INSERT INTO GamerTournament (GamerID, TournamentID)

VALUES ((SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes'),1),

((SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes'),2),

((SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes'),3),

((SELECT GamerID FROM Gamer WHERE GamerName = 'John2Tymes'),4),

((SELECT GamerID FROM Gamer WHERE GamerName = 'AlexPSU79'),5),

((SELECT GamerID FROM Gamer WHERE GamerName = 'AlexPSU79'),6),

((SELECT GamerID FROM Gamer WHERE GamerName = 'AlexPSU79'),7),

((SELECT GamerID FROM Gamer WHERE GamerName = 'AlexPSU79'),8),

((SELECT GamerID FROM Gamer WHERE GamerName = 'VG Chewy'),9),

((SELECT GamerID FROM Gamer WHERE GamerName = 'VG Chewy'),10),

((SELECT GamerID FROM Gamer WHERE GamerName = 'VG Chewy'),11),

((SELECT GamerID FROM Gamer WHERE GamerName = 'LinesofChaos'),12),

((SELECT GamerID FROM Gamer WHERE GamerName = 'LinesofChaos'),13),

((SELECT GamerID FROM Gamer WHERE GamerName = 'LinesofChaos'),14),

((SELECT GamerID FROM Gamer WHERE GamerName = 'King\_Pain'),15),

((SELECT GamerID FROM Gamer WHERE GamerName = 'King\_Pain'),16),

((SELECT GamerID FROM Gamer WHERE GamerName = 'ItsMrJoker'),17),

((SELECT GamerID FROM Gamer WHERE GamerName = 'ItsMrJoker'),18),

((SELECT GamerID FROM Gamer WHERE GamerName = 'iamcoyson'),19),

((SELECT GamerID FROM Gamer WHERE GamerName = 'kid 4 show'),20)

GO

-- Read rows in GamerTournament table

SELECT \* FROM GamerTournament

GO

-- Insert data into GamerSponsor table

INSERT INTO GamerSponsor(GamerID, SponsorID)

VALUES (2,4), (10,5), (3,1), (2,3), (9,2)

GO

-- Read rows in GamerSponsor table

SELECT \* FROM GamerSponsor

GO

-- Insert data into InGamePurchaseGamer table

INSERT INTO InGamePurchaseGamer(GamerID, PurchaseID)

VALUES (11,8), (2,15), (14,1), (1,12), (14,6)

GO

-- Read rows in InGamePurchaseGamer table

SELECT \* FROM InGamePurchaseGamer

GO

-- Insert data into TournamentSponsor table

INSERT INTO TournamentSponsor(TournamentID, SponsorID)

VALUES (1,2), (2,3), (3,2), (4,4), (5,5)

GO

-- Read rows in TournamentSponsor table

SELECT \* FROM TournamentSponsor

GO

-- Insert data into CharacterGameMode table

INSERT INTO CharacterGameMode(GameModeID, CharacterID)

VALUES (2,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Deadpool')),

(7,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Hera')),

(10,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Deadpool')),

(13,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Deadpool')),

(14,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Loki')),

(15,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Deadpool')),

(17,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Frodo')),

(21,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Frodo')),

(22,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Sanji')),

(24,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Sanji')),

(29,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'SanJi')),

(30,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Deadpool')),

(31,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Deadpool')),

(32,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Sanji')),

(33,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Loki')),

(34,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Loki')),

(35,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Sonic')),

(36,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Sonic')),

(37,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Hera')),

(38,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Sanji')),

(39,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Luffy')),

(40,(SELECT CharacterID FROM GameCharacter WHERE CharacterName = 'Deadpool'))

GO

-- Read rows in CharacterGameMode table

SELECT \* FROM CharacterGameMode

GO

-- Delete the GamerName Pnk Panther to allow for repeatable script using Pnk Panther name over and over

DELETE FROM Gamer WHERE GamerName = 'Pnk Panther'

GO

/\* This procedure is used when a new Gamer registers to play Savage and needs to be inserted into to the Gamer table

The first parameter, @gamerName,is the Gamer Name for the new Gamer

The second parameter, @emailAddress, is the New Gamer's Email Address

The third parameter, @creditCard, is the New Gamer's Credit Card

The fourth parameter, @region, is the Region the New Gamer lives

Returns:

SCOP\_IDENTITY() with the value inserted

\*/

CREATE PROCEDURE NewGamer(@gamerName varchar(25), @emailAddress nchar(35),

@creditCard varchar(16), @region char(13)) AS

BEGIN

INSERT INTO Gamer(GamerName, EmailAddress, CreditCard, Region)

VALUES (@gamerName, @emailAddress, @creditCard, @region)

RETURN SCOPE\_IDENTITY()

END

GO

DECLARE @newGamerID int

-- Gamer table before adding New Gamer

SELECT \* FROM Gamer

-- The new gamer name, emailaddress, card and region

EXEC @newGamerID = NewGamer 'Pnk Panther', 'janedoe@mymail.xyz','5517608342653934', 'North America';

-- Gamer Table after adding New Gamer

SELECT \* FROM Gamer

GO

/\* This procedure is used when the Gamer needs to update their credit card information

The first parameter, @gamerName, is the GamerName we are updating

The second parameter, @newCreditCard, is the Gamer's new Credit Card number

\*/

CREATE PROCEDURE CreditCardUpdate (@gamerName varchar(25), @newCreditCard varchar(16))

AS

BEGIN

UPDATE Gamer SET CreditCard = @newCreditCard

WHERE GamerName = @gamerName

END

GO

--Before the Credit Card update

SELECT

Gamer.GamerName,

Gamer.CreditCard

FROM Gamer

WHERE Gamer.GamerName = 'BigRed748'

ORDER BY Gamer.GamerName

GO

-- Updating the Credit Card

EXEC CreditCardUpdate 'BigRed748', '4109904278359505'

GO

-- After Credit Card update

SELECT

Gamer.GamerName,

Gamer.CreditCard

FROM Gamer

WHERE Gamer.GamerName = 'BigRed748'

ORDER BY Gamer.GamerName

GO

----Count the number times Warrior is used by a specific user to be used in the view GameCharacterBreakDown

CREATE FUNCTION dbo.Warrior (@gamerName varchar(25))

RETURNS int AS

BEGIN

DECLARE @warrior int

SELECT @warrior = COUNT(Class)FROM GamerTournament AS gt

JOIN Gamer AS g ON g.GamerID = gt.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

JOIN CharacterGameMode AS cgm ON cgm.GameModeID = gm.GameModeID

JOIN GameCharacter AS gc ON gc.CharacterID = cgm.CharacterID

WHERE gc.Class = 'Warrior'

AND g.GamerName = @gamerName

RETURN @warrior

END

GO

----Count the number times Assassin is used by a specific user to be used in the view GameCharacterBreakDown

CREATE FUNCTION dbo.Assassin (@gamerName varchar(25))

RETURNS int AS

BEGIN

DECLARE @assassin int

SELECT @assassin = COUNT(Class)FROM GamerTournament AS gt

JOIN Gamer AS g ON g.GamerID = gt.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

JOIN CharacterGameMode AS cgm ON cgm.GameModeID = gm.GameModeID

JOIN GameCharacter AS gc ON gc.CharacterID = cgm.CharacterID

WHERE gc.Class = 'Assassin'

AND g.GamerName = @gamerName

RETURN @assassin

END

GO

----Count the number times Mage is used by a specific user to be used in the view GameCharacterBreakDown

CREATE FUNCTION dbo.Mage (@gamerName varchar(25))

RETURNS int AS

BEGIN

DECLARE @mage int

SELECT @mage = COUNT(Class)FROM GamerTournament AS gt

JOIN Gamer AS g ON g.GamerID = gt.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

JOIN CharacterGameMode AS cgm ON cgm.GameModeID = gm.GameModeID

JOIN GameCharacter AS gc ON gc.CharacterID = cgm.CharacterID

WHERE gc.Class = 'Mage'

AND g.GamerName = @gamerName

RETURN @mage

END

GO

----Create a function to count the number times Hunter is used by a specific user to be used in the view GameCharacterBreakDown

CREATE FUNCTION dbo.Hunter (@gamerName varchar(25))

RETURNS int AS

BEGIN

DECLARE @hunter int

SELECT @hunter = COUNT(Class)FROM GamerTournament AS gt

JOIN Gamer AS g ON g.GamerID = gt.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

JOIN CharacterGameMode AS cgm ON cgm.GameModeID = gm.GameModeID

JOIN GameCharacter AS gc ON gc.CharacterID = cgm.CharacterID

WHERE gc.Class = 'Hunter'

AND g.GamerName = @gamerName

RETURN @hunter

END

GO

-- Count the number times Guardian is used by a specific user to be used in the view

-- GameCharacterBreakDown

CREATE FUNCTION dbo.Guardian (@gamerName varchar(25))

RETURNS int AS

BEGIN

DECLARE @guardian int

SELECT @guardian = COUNT(Class)FROM GamerTournament AS gt

JOIN Gamer AS g ON g.GamerID = gt.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

JOIN CharacterGameMode AS cgm on cgm.GameModeID = gm.GameModeID

JOIN GameCharacter AS gc ON gc.CharacterID = cgm.CharacterID

WHERE gc.Class = 'Guardian'

AND g.GamerName = @gamerName

RETURN @guardian

END

GO

--This view generates a table of the Gamers selection history of a Game Character Class that can be used in Access.

CREATE VIEW CharacterBreakDown AS

SELECT

Gamer.GamerName,

dbo.Warrior('AlexPSU79') AS Warrior,

dbo.Assassin('AlexPSU79') AS Assassin,

dbo.Mage('AlexPSU79') AS Mage,

dbo.Hunter('AlexPSU79') AS Hunter,

dbo.Guardian('AlexPSU79') AS Guardian

FROM Gamer

WHERE gamer.GamerName = 'AlexPSU79'

UNION

SELECT

Gamer.GamerName,

dbo.Warrior('John2Tymes') AS Warrior,

dbo.Assassin('John2Tymes') AS Assassin,

dbo.Mage('John2Tymes') AS Mage,

dbo.Hunter('John2Tymes') AS Hunter,

dbo.Guardian('John2Tymes') AS Guardian

FROM Gamer

WHERE gamer.GamerName = 'John2Tymes'

UNION

SELECT

Gamer.GamerName,

dbo.Warrior('King\_Pain') AS Warrior,

dbo.Assassin('King\_Pain') AS Assassin,

dbo.Mage('King\_Pain') AS Mage,

dbo.Hunter('King\_Pain') AS Hunter,

dbo.Guardian('King\_Pain') AS Guardian

FROM Gamer

WHERE gamer.GamerName = 'King\_Pain'

GO

-- This view creates a table of the top 3 In Game Purchases for each Quarter

CREATE VIEW QuarterSales AS

WITH SalesPerQuarter

AS

(

SELECT TOP 3 202002 AS SalesQuarter,

PurchaseDescription,

SUM(Cost) AS CostOfSales

FROM InGamePurchase

WHERE PurchaseDate BETWEEN '3/1/2020' AND '5/31/2020'

GROUP BY PurchaseDescription

ORDER BY CostOfSales DESC

UNION

SELECT TOP 3 202003, PurchaseDescription,

SUM(Cost) AS CostOfSales

FROM InGamePurchase

WHERE PurchaseDate BETWEEN '6/1/2020' AND '8/31/2020'

GROUP BY PurchaseDescription

ORDER BY CostOfSales DESC

UNION

SELECT TOP 3 202004, PurchaseDescription,

SUM(Cost) AS CostOfSales

FROM InGamePurchase

WHERE PurchaseDate BETWEEN '9/1/2020' AND '12/31/2020'

GROUP BY PurchaseDescription

ORDER BY CostOfSales DESC

)

SELECT \* FROM SalesPerQuarter

GO

--This function will caluclate the win probability of a gamer based on their total wins and losses

CREATE FUNCTION dbo.WinProbability (@gamerName varchar(25))

RETURNS decimal(4,1) AS

BEGIN

DECLARE @wins int, @losses int, @returnvalue decimal(4,1)

SELECT @wins = COUNT(Wins) FROM GamerTournament AS gt

JOIN Gamer AS g ON g.GamerID = gt.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

WHERE gm.GameType = 'Domination'

AND Wins = 1

AND GamerName = @gamerName

SELECT @losses = COUNT(Losses) FROM GamerTournament AS gt

JOIN Gamer AS g ON g.GamerID = gt.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

WHERE gm.GameType = 'Domination'

AND Losses = 1

AND GamerName = @gamerName

SET @returnvalue = 100 \* @wins / (@wins+@losses)

RETURN @returnvalue

END

GO

-- Question 1 What has been the most and least popular game character over the last 10 tournaments? (DONE)

SELECT

gc.CharacterName,

COUNT(CharacterName) AS CharacterNameCount

FROM Tournament AS t

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

JOIN CharacterGameMode AS cgm ON cgm.GameModeID = gm.GameModeID

JOIN GameCharacter AS gc ON gc.CharacterID = cgm.CharacterID

WHERE TournamentID-10>0

GROUP BY

gc.CharacterName

ORDER BY CharacterNameCount DESC

GO

-- Question 2: What are the top three grossing in-game purchases at the end of each quarter? (DONE)

SELECT \* from QuarterSales

ORDER BY SalesQuarter, CostOfSales DESC

GO

-- Question 3: What is the percentage break down for a game character played by a specific gamer?

-- In this example, we want to know the breakdown for Gamers: AlexPSU79, John2Tymes, and King\_Pain(DONE)

SELECT \* FROM CharacterBreakDown

-- Question 4: What is the most selected offensive and defensive item for a particular game character?

-- In this example, we are looking at the game character Deadpool. (DONE)

SELECT

ge.EquipmentName,

count(ge.EquipmentID) AS NumberTimesSelected

FROM GamerGameEquipment AS gge

JOIN GameEquipment AS ge ON ge.EquipmentID = gge.EquipmentID

JOIN Gamer AS g ON g.GamerID = gge.GamerID

JOIN GamerTournament AS gt ON gt.GamerID = g.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

JOIN CharacterGameMode AS cgm on cgm.GameModeID = gm.GameModeID

JOIN GameCharacter AS gc ON gc.CharacterID = cgm.CharacterID

WHERE ge.ItemType = 'Offensive'

AND gc.CharacterName = 'Deadpool'

GROUP BY ge.EquipmentName

ORDER BY NumberTimesSelected DESC

SELECT

ge.EquipmentName,

count(ge.EquipmentID) AS NumberTimesSelected

FROM GamerGameEquipment AS gge

JOIN GameEquipment AS ge ON ge.EquipmentID = gge.EquipmentID

JOIN Gamer AS g ON g.GamerID = gge.GamerID

JOIN GamerTournament AS gt ON gt.GamerID = g.GamerID

JOIN Tournament AS t ON t.TournamentID = gt.TournamentID

JOIN GameMode AS gm ON gm.GameModeID = t.GameModeID

JOIN CharacterGameMode AS cgm on cgm.GameModeID = gm.GameModeID

JOIN GameCharacter AS gc ON gc.CharacterID = cgm.CharacterID

WHERE ge.ItemType = 'Defensive'

AND gc.CharacterName = 'Deadpool'

GROUP BY ge.EquipmentName

ORDER BY NumberTimesSelected DESC

GO

-- Question 5 What is the probability a specific gamer will win the tournament?

-- In this example, we will review Gamers: AlexPSU79, John2Tymes, King\_Pain, and VG Chewy(DONE)

SELECT

Gamer.GamerName,

dbo.WinProbability('AlexPSU79') AS WinProbability

FROM Gamer

WHERE Gamer.GamerName = 'AlexPSU79'

UNION

SELECT

Gamer.GamerName,

dbo.WinProbability('John2Tymes') AS WinProbability

FROM Gamer

WHERE Gamer.GamerName = 'John2Tymes'

UNION

SELECT

Gamer.GamerName,

dbo.WinProbability('King\_Pain') AS WinProbability

FROM Gamer

WHERE Gamer.GamerName = 'King\_Pain'

UNION

SELECT

Gamer.GamerName,

dbo.WinProbability('VG Chewy') AS WinProbability

FROM Gamer

WHERE Gamer.GamerName = 'VG Chewy'

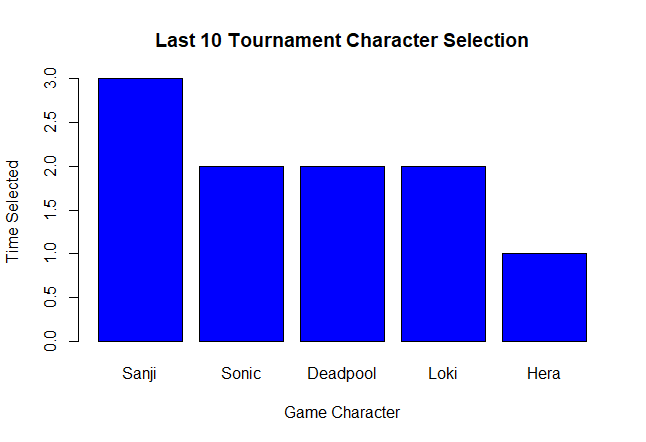
GO

**Validation**

To ensure the database is designed and performs to the requirements of Q&A Inc., the cross-functional team came up with five questions that we expect the database to be able to help us answer. Below are the questions and their answers:

1. What has been the most and least popular game character over the last 10 tournaments?

Using RStuido we can see the most popular game character in the last 10 games is Sanji, and the least popular is Hera.

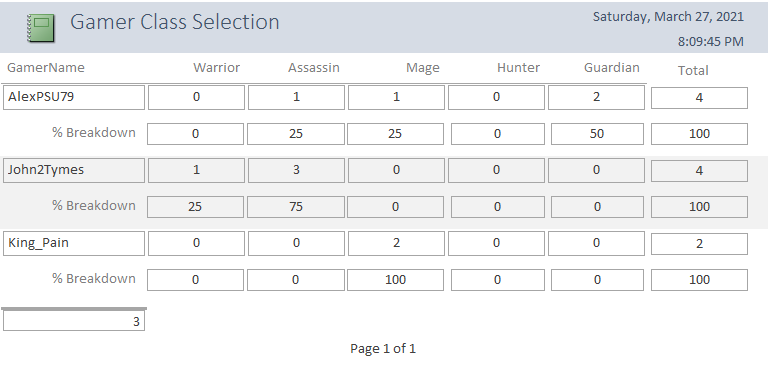


1. What are the top three grossing in-game purchases at the end of each quarter?

Linking with Excel we can see the top three purchases for the second, third and fourth quarter.

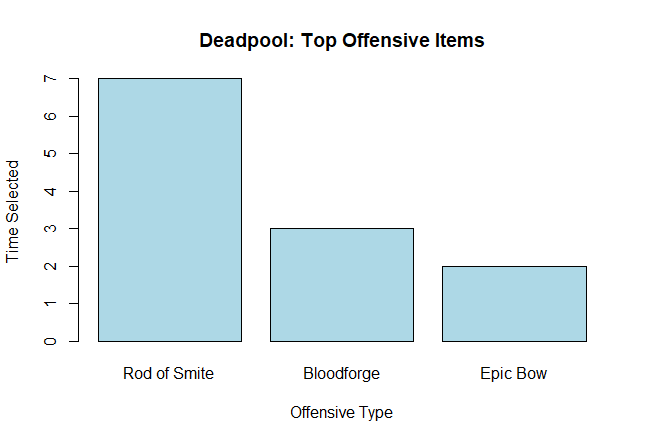
1. What is the percentage break down of a game character class played by a specific gamer?

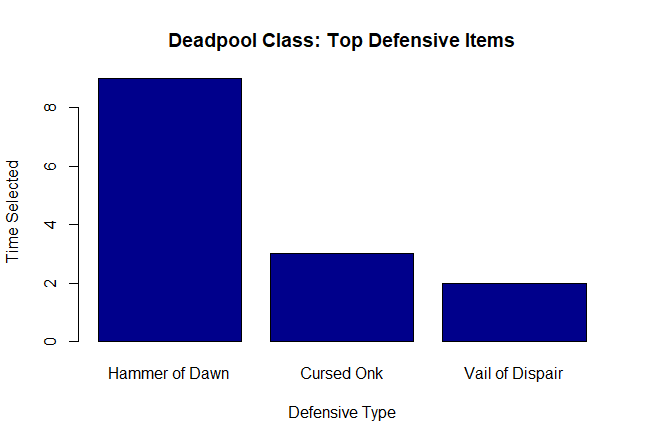
Linking with Access we can run a report that shows percentage break down of a game character class for these specific gamers: AlexPSU79, John2Tymes, and King\_Pain.



1. What is the most selected offensive and defensive item for a particular game character?

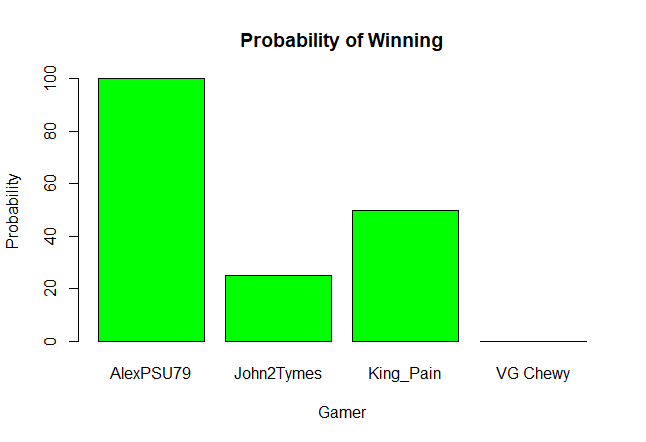
Using RStudio we can determine the most used offensive and defensive items used by the game character Deadpool.





1. What is the probability a specific gamer will win the tournament?

Using RStudio we can see the probability the following gamers will win the next tournament: AlexPSU79 (100%), John2Tymes (20%), King\_Pain (50%), and VG Chewy (0%).



**Reflection**

Going into this class all I knew about a database was the name Microsoft Access and it can store data. I did not understand that SAP and Oracle ERP’s were themselves databases until I was a few weeks into the class. I have a bigger appreciation for the database and all the planning that goes into the design, implementation, and maintenance. I would often ask our IT team “can we not just delete a part number” or “use the same customer code” when a vendor’s company name would change, and then get frustrated with the response. Constructing the conceptual and logical models help me understand the requirements for uniqueness and how our business rules drive the constraints we place on the database. I have not done programming in almost 20 years and found it to be my biggest hurdle, but it would be very gratifying when I would see my query execute correctly. I still struggle with finding the correct syntax for the primary and foreign key relationship between my Game Equipment, Offensive and Defensive tables. To a point that makes me rethink the logical model for those three tables. In parallel to this course, my work has been struggling with an Excel sheet that is storing and performing data analysis on approximately 8,000 row entries. I started asking some simple database design questions I learned though this course and now have been assigned to the team to create a database to handle the 8,000 and growing set of data. I would not have the confidence I have to take this on if it were not for this course and project.

**Summary**

A database is a powerful and a necessary tool for any business to succeed. However, the tool is only as good as it is designed, implemented, and maintained. Starting from the bottom with the conceptual model and building on top of that with the logical model give a strong foundation to the database. Progressing into using CRUD SQL to organize our data will allow us to then use SELECT, FUNCTION, and PROCEDURE script to ask questions about the business. SQL scripting will provide us with the data to answer the questions, but SQL is not the strongest tool to provide the analysis. Linking the database to Microsoft Access, Excel or RStudio will take the handoff from the database in order for the the business to analyze the trends and needs of the business.