**APEX Legends: Database System Design and Implementation**

**1. Database Objectives And Scope**

The objective of the database for the game Apex Legends is to help gamers better understand the game on a statistical standpoint so they may be able to make better in game decisions that would better help their playstyle and maximize their in game positive output

The database will cover all aspects of the game: Characters, Special Abilities, Damage, Weapons, Attachments, etc.

**2. User Requirements**

**-** Each user must have an eligible gaming console(Personal Computer, Xbox One Console, Playstation 4 Console)

- PC Requirements

- Operating System: Windows 7 64-bit or Windows 10 64-bit

- Processor: Intel Core 13-6300 3.8 GHz / AMD FX-4350 4.2 GHz Quad Core

- RAM: 6GB

- Video Cord: 1 GB NVIDIA GeForce GT 640 / Radeon HD 7730

- Free Disk Space: 22 GB

- User must have a monitor compatible with console

- User must have a compatible controller with console

- User must have an internet connection with speeds of 1-3 Mb per second

**3. Business Rules**

*Every HeroID can have one to many ItemID​*

*Every ItemID can have zero to one HeroID​*

*Every HeroID can have zero to many WeaponID​*

*Every WeaponID can have zero to one HeroID​*

*Every HeroID can have one and only one AbilityID​*

*Every AbilityID can have zero to one HeroID​*

*Every WeaponID can have one to many ScopeID​*

*Every ScopeID can have zero to one WeaponID*

*Every WeaponnID can have zero to many GunID​*

*Every GunID can have zero to one WeaponID​*

*Every GunID can have one and only one FireModeID​*

*Every FireModeID can have one and only one GunID​*

*Every ItemID can have zero to many VestID​*

*Every VestID can have zero to one ItemID​*

*Every ItemID can have zero to many ShieldID​*

*Every ShieldId can have zero to one ItemID*

*Every ItemID can have zero to many HelmetID​*

*Every HelmetID can have zero to one ItemID​*

*Every ItemID can have zero to many ExtendedMagID​*

*Every ExtendedMagID can have zero to one ItemID​*

*Every ItemID can have zero to many StockID​*

*Every StockID can have zero to one ItemID​*

*Every ItemID can have zero to many BarrelStabilizerID​*

*Every BarrelStabilizerID can have zero to one ItemID*

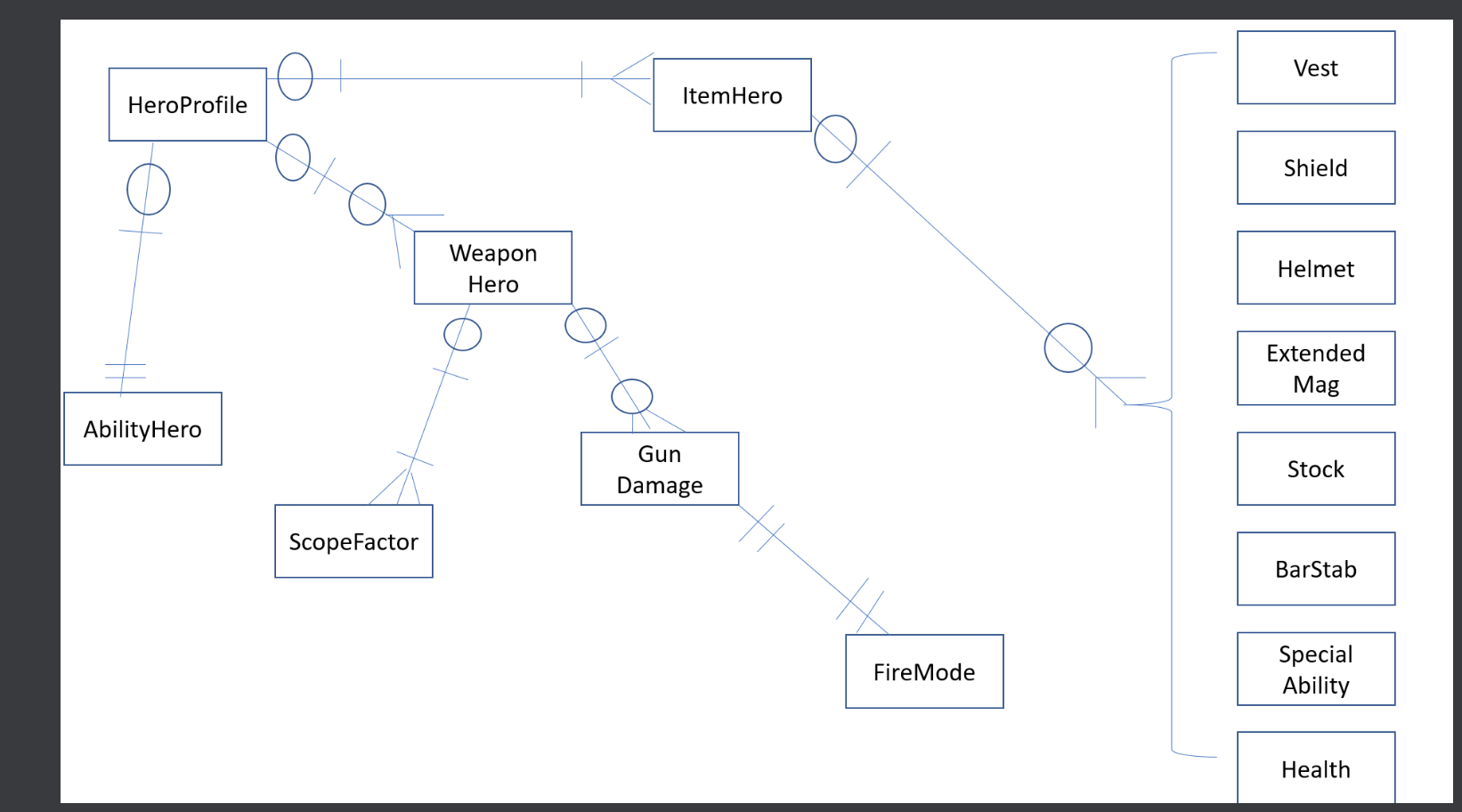
*Every ItemID can have zero to many SpecialAbilityID​*

*Every SpecialAbilityID can have zero to one ItemID​*

*Every ItemID can have zero to many HealthID​*

*Every HealthID can have zero to one ItemID*

**4. Entity Relationship Diagram**

****

Due to the scope of the ERD not being able to fit in one page, the properties of the entities and the identification of the FK and the PK will be listed below.

**HeroProfile Table:**

* Primary Key: HeroID(INT, NOT NULL)
* Hero Name(VarChar(50), NOT NULL)
* Finisher(VarChar(50), NOT NULL)
* Legend Type(VarChar(50), NOT NULL)

Foreign Keys:

* AbilityID(INT, NOT NULL) (REF AbilityHero)
* WeaponID(INT, NOT NULL) (REF WeaponHero)
* ItemID(INT, NOT NULL) (ItemHero)

**ItemHero Table**

PrimaryKey : ItemID(INT, NOT NULL)

GradeDesc(VarChar(50), NOT NULL)

Foreign Keys:

* HelmetID(INT, NOT NULL) (REF Helmet)
* ShieldID(INT, NOT NULL) (REF Shield)
* VestID(INT, NOT NULL) (REF Vest)
* ExtendedMagID(INT, NOT NULL) (REF ExtendedMag)
* StockID(INT, NOT NULL) (REF Stock)
* BarrelStabilizerID(INT, NOT NULL) (REF BarStab)
* SpecialAbilityID(INT, NOT NULL) (REF SpecialAbility)
* ShieldRechargeID(INT, NOT NULL) (REF ShieldRecharge)
* HealthID(INT, NOT NULL) (REF Health)

**Vest Table**

* Primary Key: VestID(INT, NOT NULL)
* VestGauges(VarChar(50), NOT NULL)
* VestGrade(VarChar(50), NOT NULL)

**Helmet Table**

* Primary Key: HelmetID(INT, NOT NULL)
* HelmetGrade(VarChar(50), NOT NULL)
* HelmetGauges(VarChar(50), NOT NULL)

**Shield Table**

* Primary Key: ShieldID(INT, NOT NULL)
* ShieldGauges(VarChar(50), NOT NULL)
* ShieldGrade(VarChar(50), NOT NULL

**ExtendedMag Table**

* Primary Key: ExtendedMagID(INT,NOT NULL)
* ExtendedMagGrade(VarChar(50), NOT NULL)

**Stock Table**

* Primary Key: StockID(INT, NOT NULL)
* StockGrade(VarChar(50), NOT NULL)

**BarStab Table**

* Primary Key: BarrelStabilizerID(INT, NOT NULL)
* BarStabGrade(VarChar(50), NOT NULL)

**SpecialAbility Table**

* Primary Key: SpecialAbilityID(INT, NOT NULL)
* SpecialAbilityName(VarChar(50), NOT NULL)
* SpecialAbilityProperty(VarChar(50), NOT NULL)
* SpecialAbilityGrade(VarChar(50), NOT NULL)

**ShieldRecharge Table**

* Primary Key: ShieldRechargeID(INT, NOT NULL)
* ShRechGrade(VarChar(50), NOT NULL)
* ShRechProp(VarChar(50), NOT NULL)

**Health Table**

* Primary Key: HealthID(INT, NOT NULL)
* HealthItemGrade(VarChar(50), NOT NULL)
* HealthItemProperty(VarChar(50), NOT NULL)
* HealthItem(VarChar(50), NOT NULL)

**AbilityHero Table**

* Primary Key: AbilityID(INT, NOT NULL)
* AbilityName(VarChar(50), NOT NULL)

**WeaponHero Table**

* Primary Key: WeaponHeroID(INT, NOT NULL)
* WeaponType(VarChar(50), NOT NULL)
* Foreign Key:
  + GunID(INT, NOT NULL) (REF GunDamage)
  + ScopeID(INT, NOT NULL) (REF ScopeFactor)

**GunDamage Table**

* Primary Key: GunID(INT, NOT NULL)
* AmmoType(VarChar(50), NOT NULL)
* GunName(VarChar(50), NOT NULL)
* MagSize(VarChar(50), NOT NULL)
* BodyShot(VarChar(50), NOT NULL)
* HeadShot(VarChar(50), NOT NULL)
* BodyShotPerMag(VarChar(50), NOT NULL)
* HeadShotPerMag(VarChar(50), NOT NULL)
* Pattern(VarChar(50), NULL)
* BodyPerHead(VarChar(50), NULL)
* Foreign Keys:
* FireModeID(INT, NOT NULL) (REF FireMode)
* WeaponID(INT, NOT NULL) (REF WeaponHero)

ScopeFactor Table

* Primary Key: ScopeID
* ScopeDesc
* Reticle
* Grade

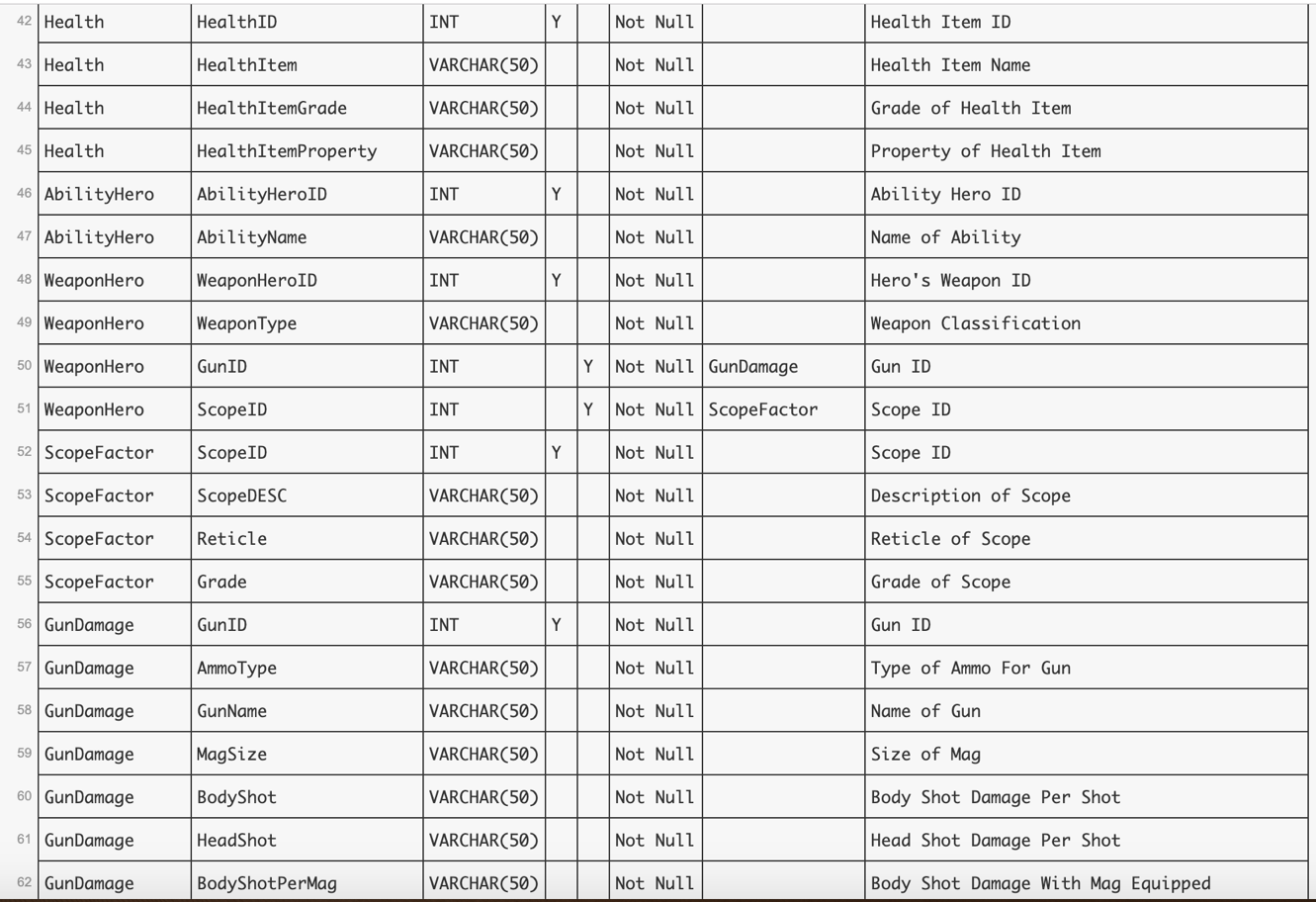
FireMode Table

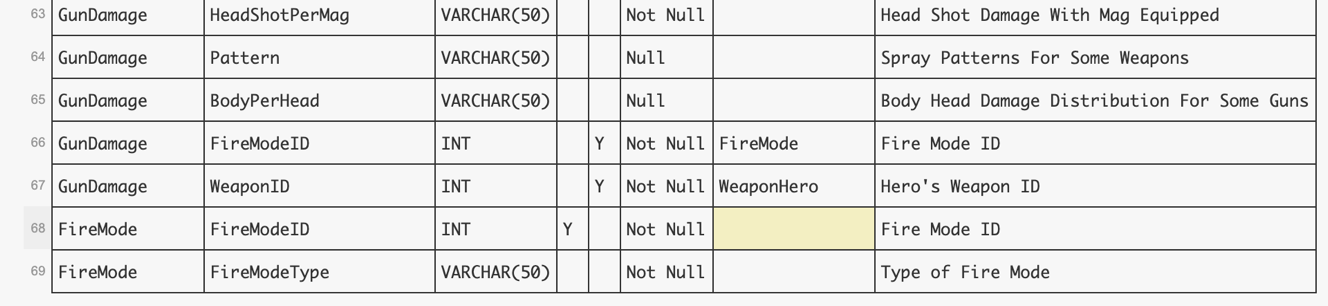
* Primary Key: FireModeID
* FireModeType

**5. Data Dictionary**









**6. SQL Created and Insert Values Tables Screenshots**

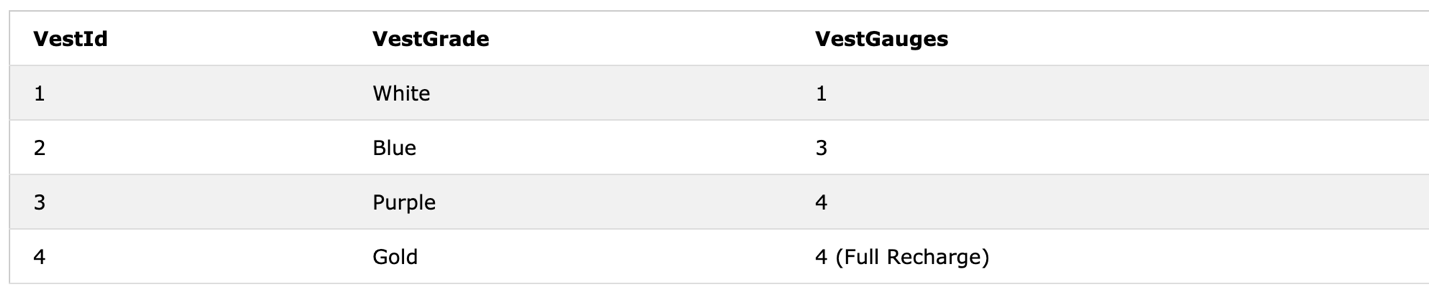
HeroProfile Table



ItemHero Table



Vest Table



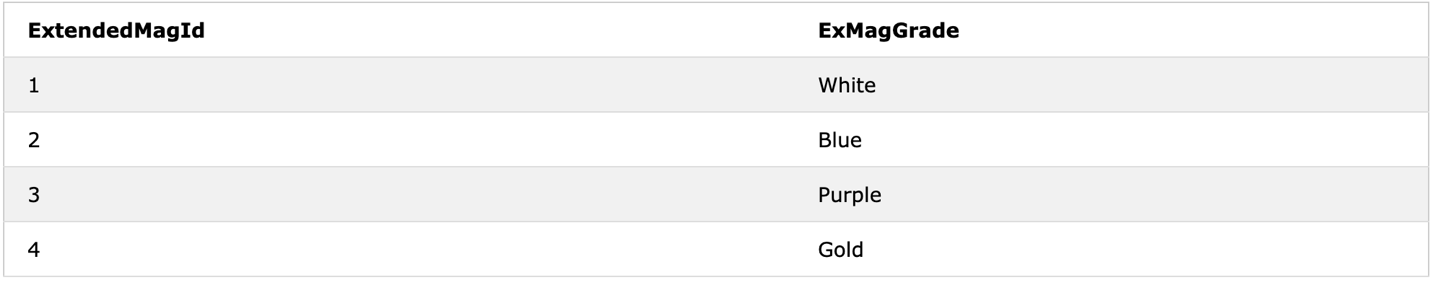
Helmet Table



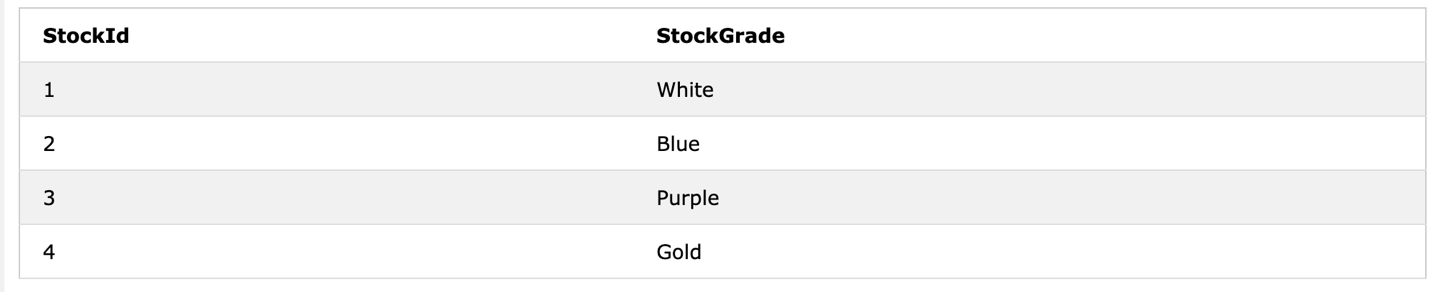
Shield Table



ExtendedMag Table



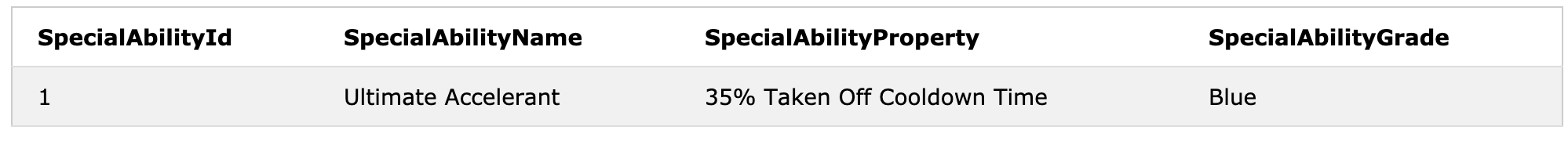
Stock Table



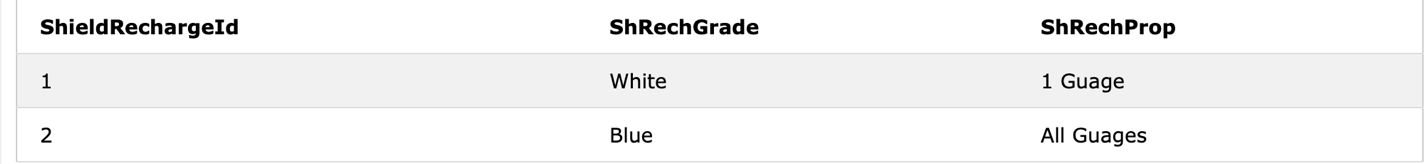
BarStab Table



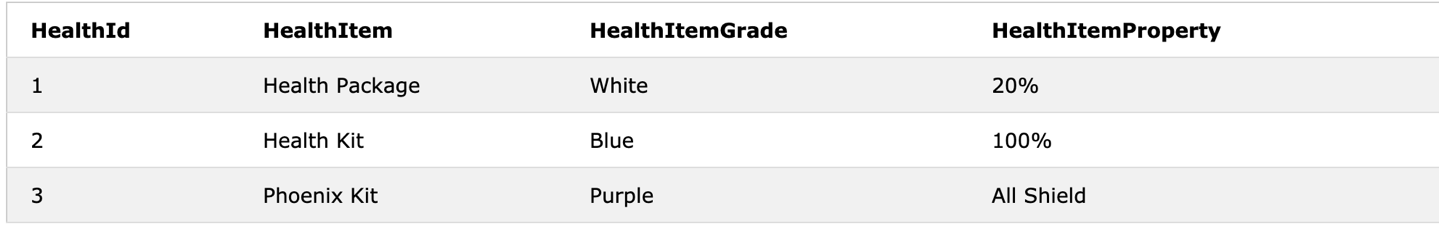
SpecialAbility Table



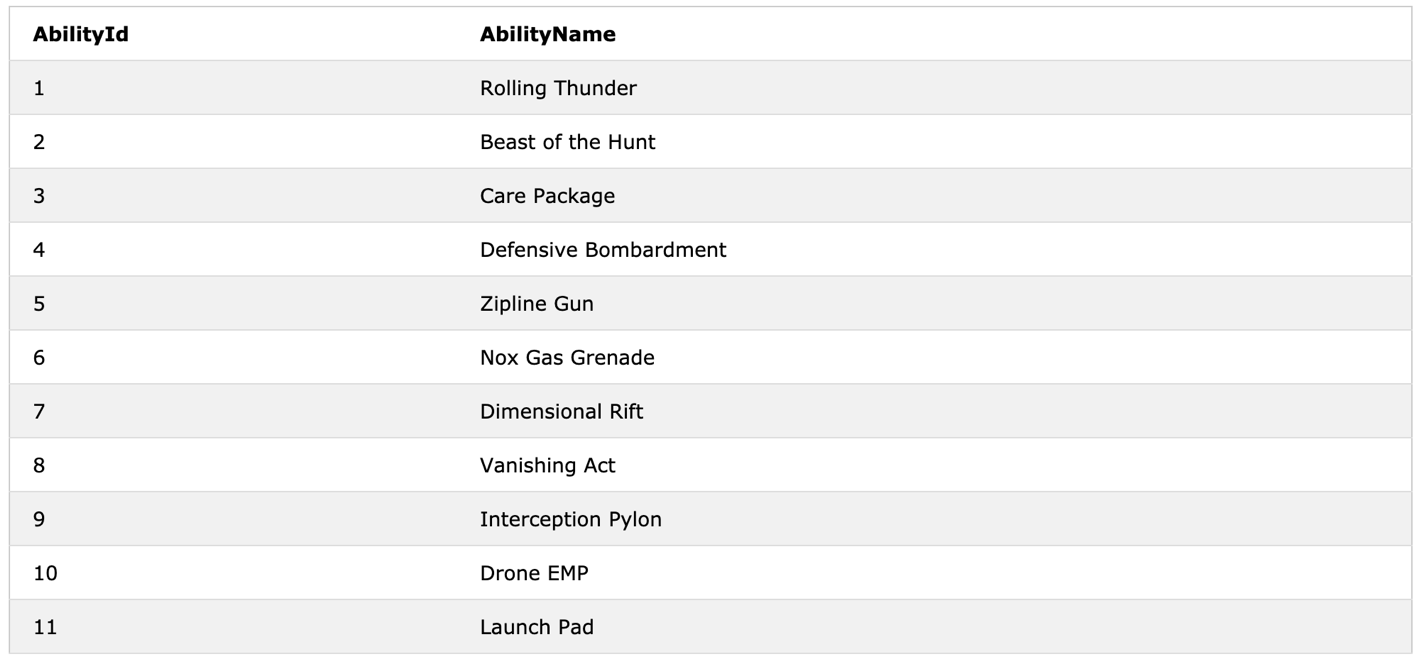
ShieldRecharge Table



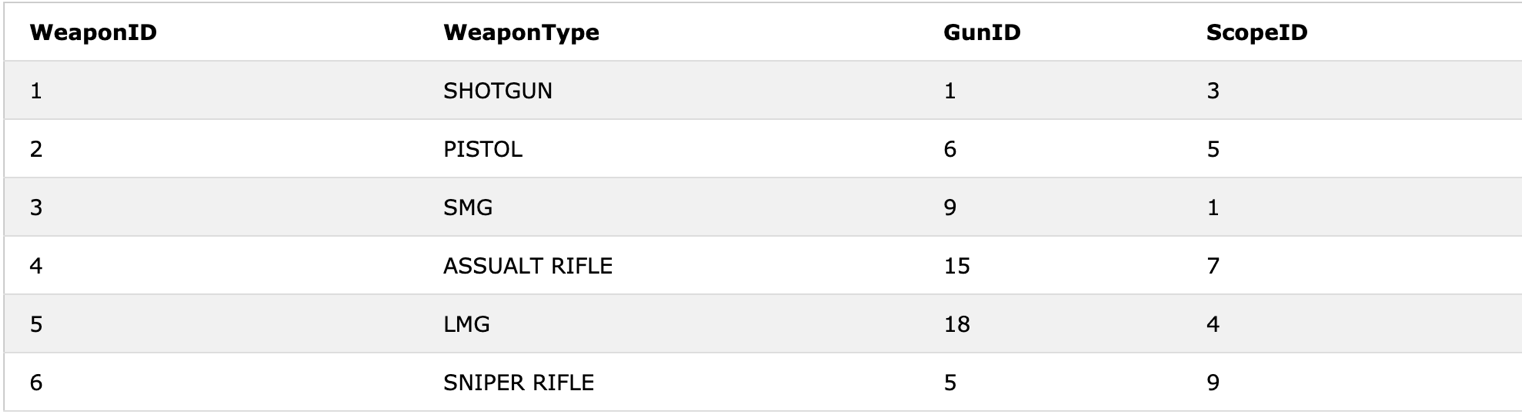
Health Table



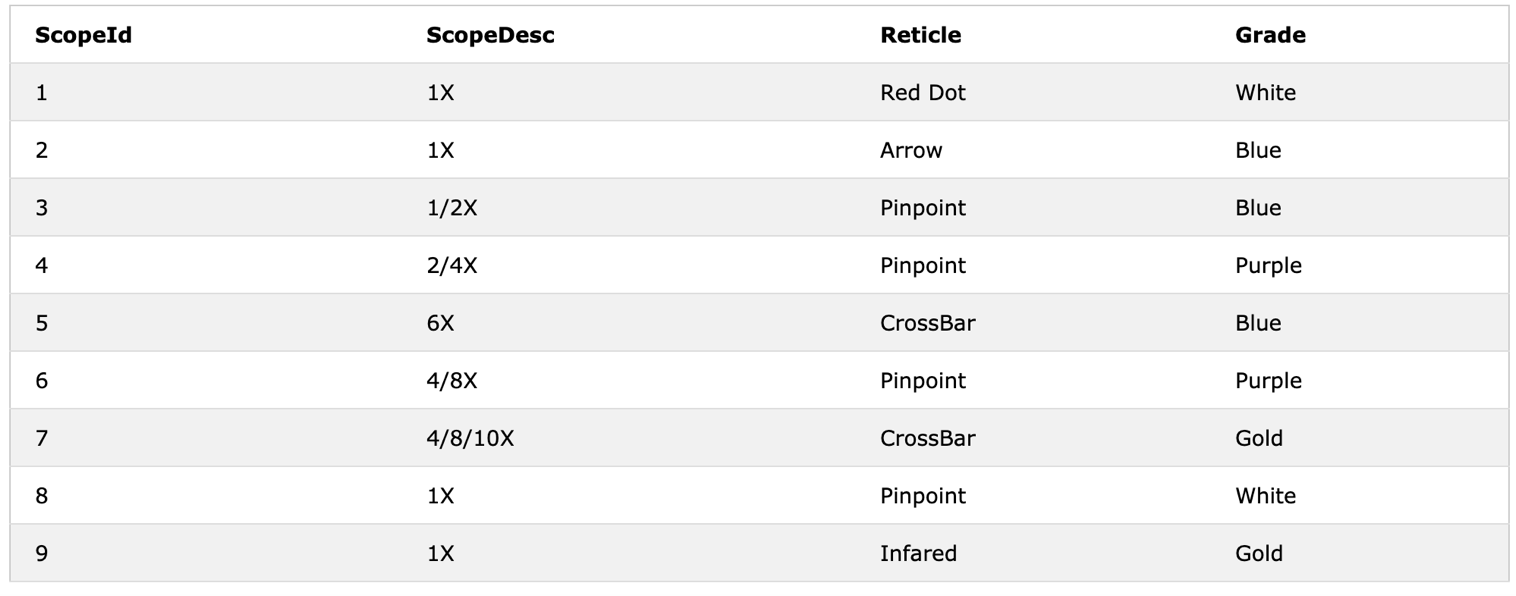
AbilityHero Table(On Page Below)



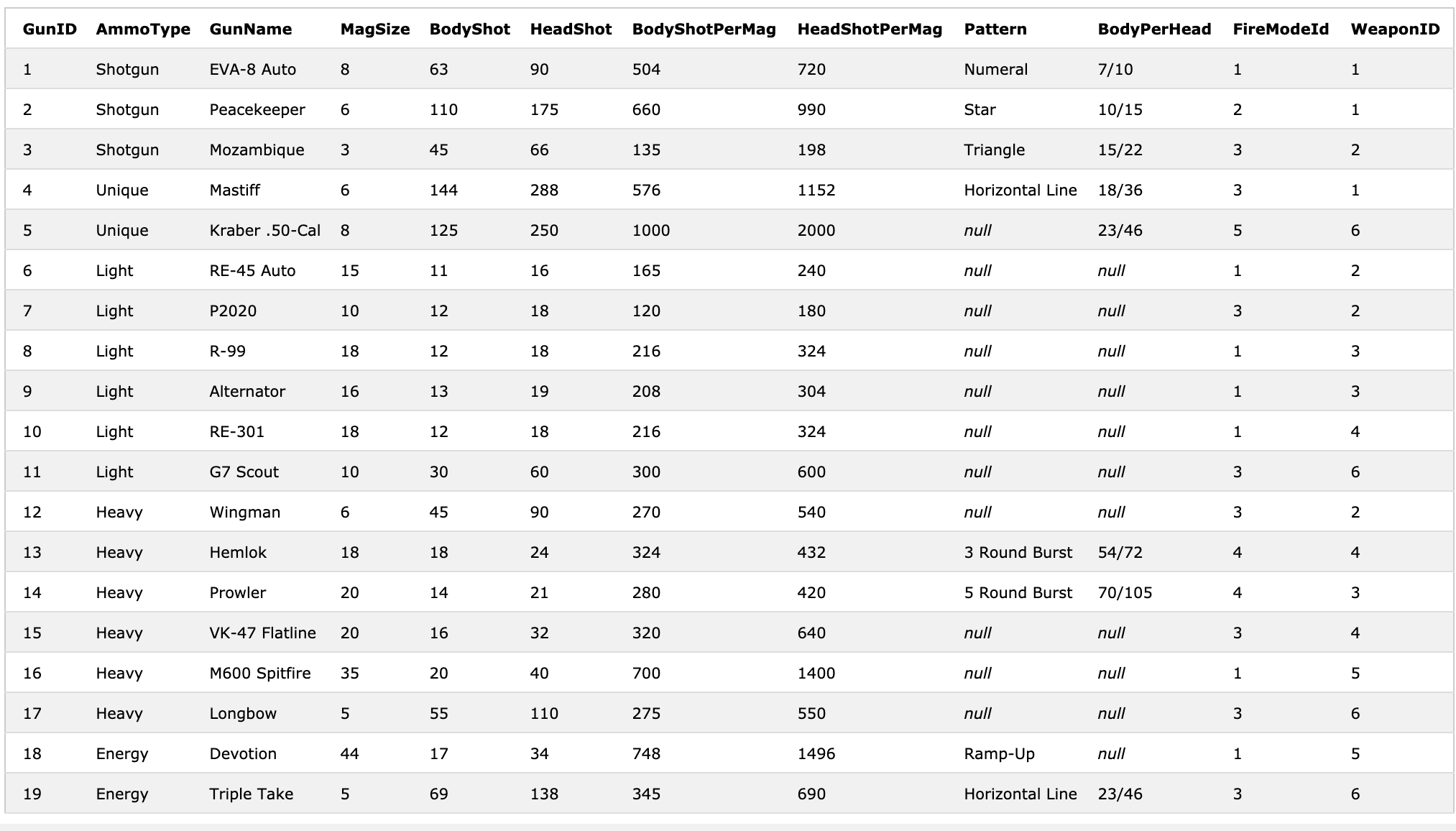
WeaponHero Table



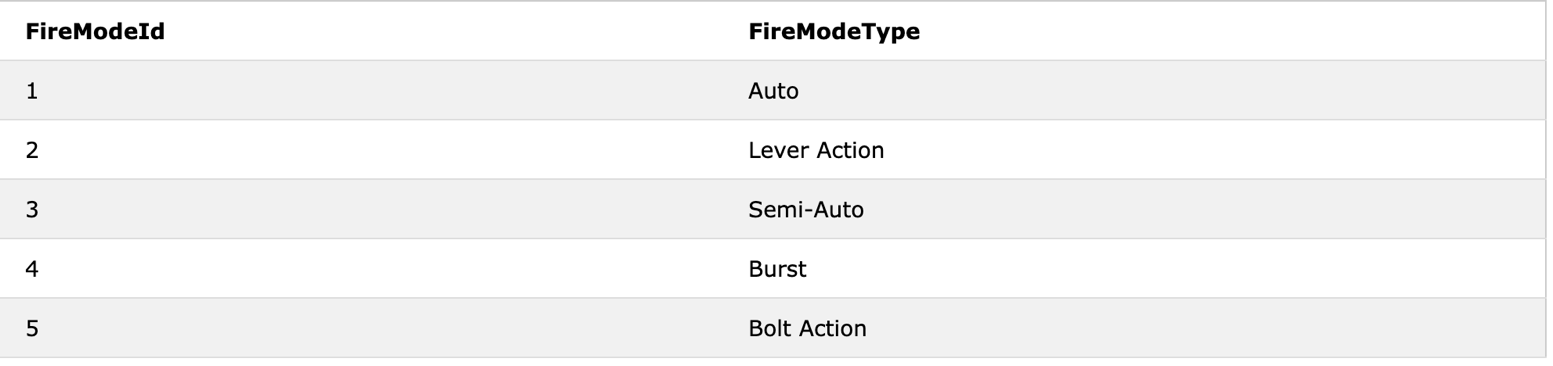
ScopeFactor Table



GunDamage Table

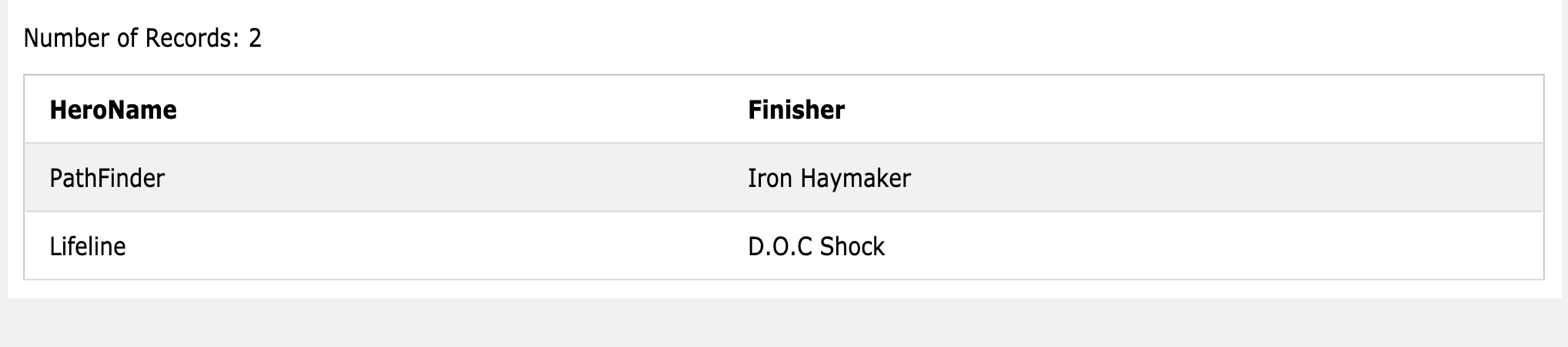


FireMode Table

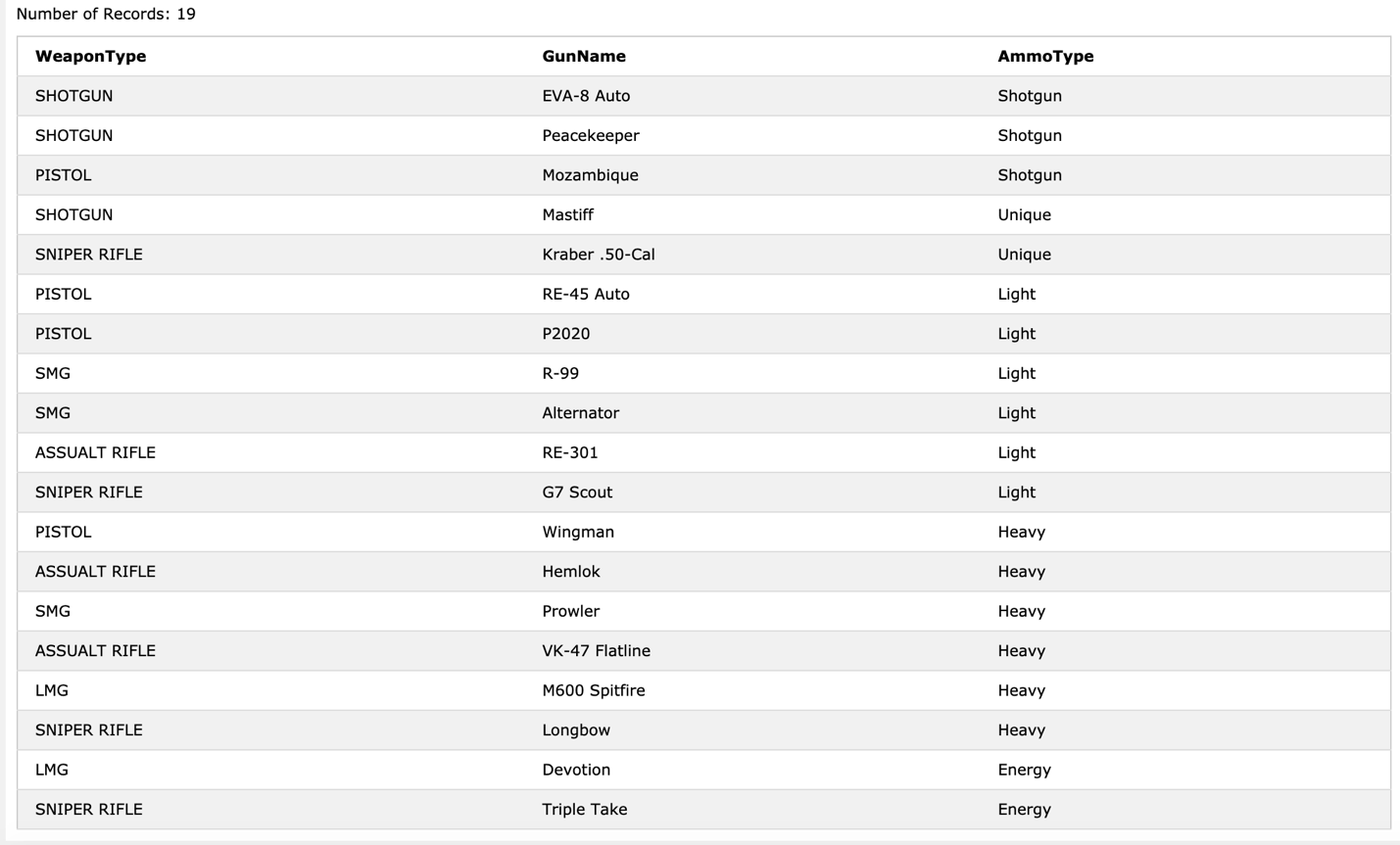


**7. SQL Queries**

SQL Query #1(Who are the Heroes And Their Respective Finishers for Class Support?)



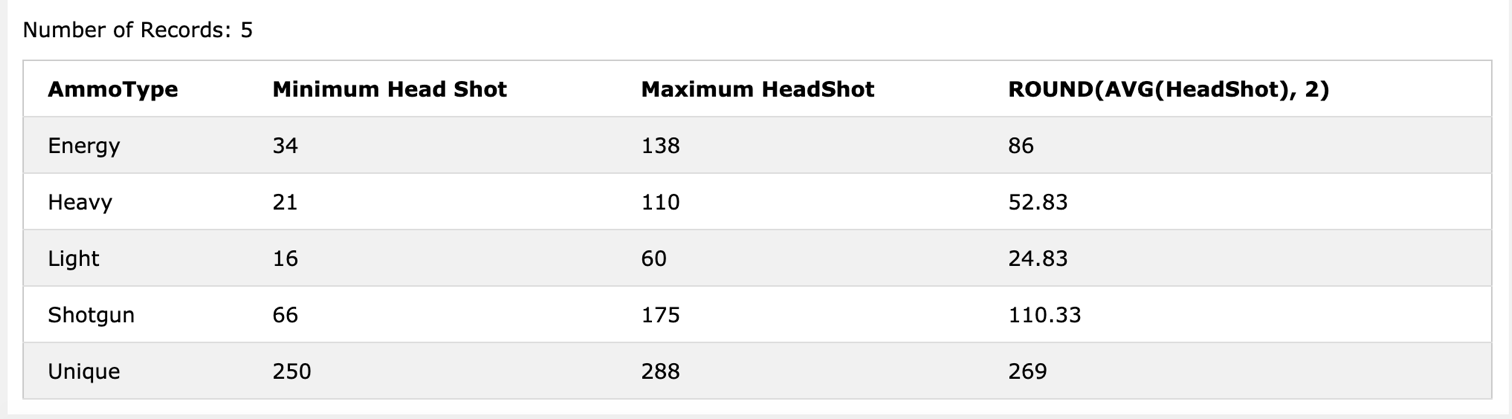
SQL Query #2(What are the different Weapon Types, Gun Types, And Ammo Types in the Database?)



SQL Query #3(What specific Item Combination Grades does a Hero have at right now?)



SQL Query #4(What are the minimum, maximum, and average HeadShot



SQL Query #5(What WeaponTypes have a BodyShotPerMag over 500 and HeadShotPerMag over 700?)

