

Relational Schemas and Integrity Constraints

Player(Account_Number: integer, Forename: string, surname: string, ExperiencePoints: integer), **ICs:** Account_Number (Primary Key).

Table is in 1NF. Account_Number determines attributes like forename and surname.

Create_Character(Character_Name: string, DefenceScore: integer, Max_Health: integer, Money_bank: integer, Health: integer, Character_Type: string, Level: integer, Character_CreationDate: string, AttackinScore: integer, Account_Number: integer), **ICs:** {Character_Name, Account_Number} (Primary Key), Account_Number (Foreign Key referencing Player, ON DELETE CASCADE), Account_Number NOT NULL

Table is in 1NF. Character_Name determines attributes like Character_Type.

Have(Character_Name; string, Damage: integer, Result: string, Attacker: string), **ICs:** Character_Name (Foreign Key referencing Create_Character, ON DELETE CASCADE), {Damage, Result, Attacker} (Foreign Key referencing Combat), Character_Name NOT NULL

Table is in 3NF. All keys are prime attributes and the prime attributes do not determine one another.

Combat(Attacker: string, Result: string, , Damage: integer ,BattleNo: integer, BattleDate: string, Weapon: string, Defender: string), **ICs:** {Attacker, Result, Damage} (Primary Key)

Table is in 3NF. Candidate keys like Attacker do not determine attributes like Weapon.

Non-prime attributes like BattleDate do not determine BattleNo. Meanwhile, prime attributes like Damage do not determine any attributes that are non-prime.

Possess_Inventory(Item: string, Item_Type: string, Quantity: integer, Character_Name: string) **ICs:** {Item, Item_Type} Primary Key, Character_Name (Foreign Key referencing Create_Character).

Table is in 1NF. Prime keys like Item determines non-prime attributes like quantity.

Equip(Item: string, Item_Type: string, Price: integer) **ICs:** {Item, Item_Type} (Foreign Key referencing Possess_Inventory), Price (Foreign Key referencing Weapon).

Table is in 3NF. All keys are prime attributes.

Weapon(, Item: string, Item_Type: string , , Price: integer , Range: integer, AttackScore: integer) **ICs:** {Item, Item_Type} Foreign Key referencing Possess_Inventory

Table is in 3NF. Only 1 non-prime key which is WeaponType.

Has(Item: string, ItemType: string)

ICs: {Item, Item_Type} Primary Key

Table is in 3NF. All keys are prime attributes.

Supplies(Item: string, Item_Type: string ,HealingScore: integer, ManaScore: integer)

ICs: {Item, Item_Type} Primary Key referencing Possess_Inventory

Table is in 3NF. There are no non-prime attributes.

Wears(Item: string, ItemType: string)

ICs: {Item, Item_Type} Primary Key

Table is in 3NF. All keys are prime attributes.

Armour(Item: string, Item_Type: string , BodyPart: string, wearable: integer ,DefendScore: integer) **ICs:** {Item, Item_Type} Primary Key referencing Possess_Inventory

Table is in 3NF. There are no non-prime attributes.

DDL Statements written in SQL

```
CREATE TABLE Player (  
    Account_Number INTEGER,  
    Forename VARCHAR(20),  
    surname VARCHAR(20),  
    ExperiencePoints INTEGER,  
    PRIMARY KEY (Account_Number));
```

```
CREATE TABLE Create_Character(  
    Character_Name VARCHAR(30),  
    DefenceScore INTEGER,  
    Max_Health INTEGER,  
    Money_bank INTEGER,  
    Health INTEGER,  
    Character_Type VARCHAR(20),  
    Level INTEGER,  
    Character_CreationDate VARCHAR(20),  
    AttackinScore INTEGER,  
    Account_Number INTEGER NOT NULL,  
    PRIMARY KEY (Character_Name, Account_Number),  
    FOREIGN KEY (Account_Number) REFERENCES Player (Account_Number),  
    ON DELETE CASCADE);
```

```
CREATE TABLE Combat(  
    Attacker VARCHAR(20),  
    Result VARCHAR(20),  
    Damage INTEGER,  
    BattleNo INTEGER,  
    BattleDate VARCHAR(20),  
    Weapon VARCHAR(20),  
    Defender VARCHAR(20),  
    PRIMARY KEY (Attacker, Result, Damage));
```

```
CREATE TABLE Possess_Inventory(  
    Item VARCHAR(20),  
    Item_Type VARCHAR(20),
```

```
Quantity INTEGER,  
Character_Name VARCHAR(30),  
PRIMARY KEY (Item, Item_Type),  
FOREIGN KEY (Character_Name) REFERENCES Create_Character (Character_Name));
```

```
CREATE TABLE Equip_Weapon(  
    Item VARCHAR(20),  
    Item_Type VARCHAR(20),  
    Price INTEGER,  
    Range: integer,  
    AttackScore integer,  
    PRIMARY KEY (Item, Item_Type, Price));
```

```
CREATE TABLE Has_Supplies(  
    Item VARCHAR(20),  
    Item_Type VARCHAR(20),  
    HealingScore INTEGER,  
    ManaScore INTEGER,  
    PRIMARY KEY (Item, Item_Type));
```

```
CREATE TABLE Wears_Armour(  
    Item VARCHAR(20),  
    Item_Type VARCHAR(20),  
    BodyPart VARCHAR(20),  
    wearable VARCHAR(20),  
    DefendScore INTEGER,  
    PRIMARY KEY (Item, Item_Type));
```