Player(Account_Number:Integer, Character_Name:String, email_Address:String, Forename:String, surname:String), ICs: Account_Number(PrimaryKey), Character_Name NOTNULL (Foreign key referencing HasCharacter)

Player - Second normal form normalisation - 2NF

HasCharacter (Account_Number:Integer, Character_Creation_Date: Date, Character_Expirary_Date: Date, Character_Name :String, Character_Type String, Level:Integer, ExperiencePoints:Integer, Max_Health:Integer, Health:Integer, AttackinScore:Integer, Defence Score:Integer, StealthScore:Integer, ManaScore:Integer, Money_bank:Float, Money_wallet Float), ICs: {Account_Number, Character_Name} (Primary key), Account_Number NOTNULL (Foreign key referencing Player, ONDELETE CASCADE)

HasCharacter - Second normal form normalisation - 2NF

ContainsItemSpaceHas(Character:String, Item:String, Quantity:Integer, Worn:Boolean, Equipped:Boolean), ICs: {Character, Item} (Primary Key), Item NOTNULL (Foreign key referencing Object, ONDELETE CASCADE), Character NOTNULL (Foreign key referencing HasCharacter)

ContainsItemSpaceHas - Second normal form normalisation - 2NF

Object (Item:String, Item_Type:String, WeaponType:String, Range:Integer, Price:Float, DefendScore:Integer, AttackeScore:Integer, HealingScore:Integer, ManaScore:Integer, SingleUse:Boolean, Wearable:Boolean, BodyPart:String), ICs: Item (Primary key)

Object - Second normal form normalisation - 2NF

BattleParticipate(BattleNo:Integer,Attacker:String, Defender:String, Result:String, BattleDate:String, Damage:Integer, Weapon:String), ICs: Attacker NOTNULL (Foreign key referencing HasCharacter), Defender NOTNULL(Foreign key referencing HasCharacter) ,BattleNo (Primary key)

BattleParticipate - Third normal form normalisation - 3NF

SQL

```
CREATE TABLE Player(
Account_Number INTEGER NOT NULL,
Forename TEXT,
surname TEXT,
email_Address TEXT,
PIRMARY KEY(AccountNumber)
);
```

```
CREATE TABLE HasCharacter(
Account_Number INTEGER NOT NULL,
Character_Creation_Date DATE,
Character_Expirary_Date DATE,
Character_Name TEXT NOT NULL,
Character_Type TEXT,
Level INTEGER,
ExperiencePoints INTEGER,
Max_Health INTEGER,
Health INTEGER,
AttackinScore INTEGER,
DefenceScore INTEGER,
StealthScore INTEGER,
ManaScore INTEGER,
Money_bank FLOAT,
Money_wallet FLOAT,
FOREIGN KEY (Account_Number) REFERENCES Player(Account_Number),
PRIMARY KEY(Character_Name, Account_Number)
);

ALTER TABLE Player TO Player_Old;

CREATE TABLE Player (
Account_Number INTEGER NOT NULL,
Character_Name TEXT NOT NULL,
Forename TEXT,
surname TEXT,
email_Address TEXT,
FOREIGN KEY (Character_Name) REFERENCES HasCharacter(Character_Name),
PRIMARY KEY (Account_Number)
);

CREATE TABLE Object(
Item TEXT NOT NULL,
Item_Type TEXT,
WeaponType TEXT,
Range INTEGER,
Price FLOAT,
DefendScore INTEGER,
AttackScore INTEGER,
HealingScore INTEGER,
ManaScore INTEGER,
SingleUse BOOLEAN CHECK (SingleUse = 1 OR SingleUse = 0),
Wearable BOOLEAN CHECK (Wearable = 1 OR Wearable = 0),
BodyPart TEXT,
PIMARY KEY(Item)
);
```

```
CREATE TABLE ContainsItemSpaceHas(
Character TEXT NOT NULL,
Item TEXT NOT NULL,
Quantity INTEGER,
Worn BOOLEAN CHECK (Worn = 1 OR Worn = 0),
Equipped BOOLEAN CHECK (Equipped = 1 OR Equipped = 0),
PRIMARY KEY(Character, Item),
FOREIGN KEY (Character) REFERENCES HasCharacter(Charcter_Name),
FOREIGN KEY (Item) REFERENCES Object(Item)
);

CREATE TABLE BattleParticipate(
BattleDate DATE,
BattleNo INTEGER NOT NULL,
Attacker TEXT NOT NULL,
Defender TEXT NOT NULL,
Weapon TEXT,
Result TEXT,
Damage INTEGER,
PRIMARY KEY (BattleNo),
FOREIGN KEY (Attacker) REFERENCES HasCharacter(Character_Name),
FOREIGN KEY (Defender) REFERENCES HasCharacter(Character_Name)
);
```