Milestone #2
Relational schema, Integrity Constraints, DDL & Normalisation

Relational Schema & Integrity Constraints:

Player (Forename: string, Surname: string, email: string, AccountNo: integer, level: integer, experiencePoints: integer, moneyWallet: integer, moneyBank: real, Name: string)
IC: Primary Key {AccountNo}
Foreign Key {Name} referencing Owns_Character, Name is UNIQUE CANNOT BE NULL, ON DELETE REJECT

Note: Name Here is unique based on the assumption made in Milestone #1 for reading the CSV Files

Owns_Character (AccountNo: integer, maxHealth: integer, stealthScore: integer, defenceScore: integer, type: string, health: integer, expiryDate: string, creationDate: string, attackingScore: integer, maneScore: integer, Name: string)
IC: Primary Key {Name, AccountNo}
Foreign Key {AccountNo} referencing Player, AccountNo CANNOT BE NULL, ON DELETE CASCADE

CombatInfo (BattleNo: integer, BattleDate: string)
IC: Primary Key {BattleNo}

char_combat_status (damage: integer, result: string, weapon: string, defender: string, Name: string, BattleNo: integer)
IC: Primary Key {BattleNo, Name}
Foreign Key Name referencing Owns_Character, Name CANNOT BE NULL, ON DELETE CASCADE
Foreign Key BattleNo referencing CombatInfo, ON DELETE REJECT

Inventory (Name: string)
IC: Primary Key {Name}
Foreign Key Name referencing Owns_Character(Name), Name CANNOT BE NULL, ON DELETE CASCADE

Weapon (item: string, type: string, range: integer, price: integer, damage_points: integer)
IC: Primary Key {item}

Armour (item: string, body_part: string, price: integer, defence_score: integer)
IC: Primary Key {item}

Supply (item: string, healing_score: integer, price: integer, mana_score: integer)
IC: Primary Key {item}

weapon_inv (quantity: integer, equipped: integer, Name: string, item: string)
IC: Primary Key {Name, item}
Foreign Key Name referencing Inventory, Name CANNOT BE NULL, ON DELETE CASCADE
Foreign Key item referencing weapon, item CANNOT BE NULL, ON DELETE CASCADE

armour_inv (quantity: integer, equipped: integer, Name: string, item: string)
IC: Primary Key {Name, item}
Foreign Key Name referencing Inventory, Name CANNOT BE NULL, ON DELETE CASCADE
Foreign Key item referencing Armour, item CANNOT BE NULL, ON DELETE CASCADE

supply_inv (quantity: integer, equipped: integer, Name: string, item: string)
IC: Primary Key {Name, item}
Foreign Key Name referencing Inventory, Name CANNOT BE NULL, ON DELETE CASCADE
Foreign Key item referencing Supply, item CANNOT BE NULL, ON DELETE CASCADE

DDL:

Player

```
CREATE TABLE Player (
        Forename TEXT,
        Surname TEXT,
        email TEXT,
        AccountNo INTEGER NOT NULL,
        level INTEGER,
        experiencePoints INTEGER,
        moneyWallet INTEGER,
        moneyBank REAL,
        Name TEXT NOT NULL,
        PRIMARY KEY (AccountNo),
        FOREIGN KEY (Name) REFERENCES Owns_Character(Name) ON DELETE RESTRICT
);
```

Character

```
CREATE TABLE Owns_Character (
        AccountNo INTEGER NOT NULL,
        maxHealth INTEGER,
        stealthScore INTEGER,
        defenceScore INTEGER,
        type TEXT,
```

```
        health INTEGER,
        expiryDate TEXT,
        creationDate TEXT,
        attackingScore INTEGER,
        maneScore INTEGER,
        Name TEXT NOT NULL,
        PRIMARY KEY (Name, AccountNo),
        FOREIGN KEY (AccountNo) REFERENCES Player(AccountNo) ON DELETE CASCADE
    );
```

CombatInfo

```
    CREATE TABLE CombatInfo (
        BattleNo INTEGER NOT NULL,
        BattleDate TEXT,
        PRIMARY KEY (BattleNo)
    );
```

char_combat_status

```
    CREATE TABLE char_combat_status (
        damage INTEGER,
        result TEXT,
        weapon TEXT,
        defender TEXT,
        Name TEXT NOT NULL,
        BattleNo INTEGER NOT NULL,
        PRIMARY KEY (BattleNo, Name),
        FOREIGN KEY (Name) REFERENCES Owns_Character(Name) ON DELETE CASCADE,
        FOREIGN KEY (BattleNo) REFERENCES CombatInfo(BattleNo) ON DELETE RESTRICT
```

```
);
```

Inventory

```
CREATE TABLE Inventory (
    Name TEXT NOT NULL,
    PRIMARY KEY (Name),
    FOREIGN KEY (Name) REFERENCES Owns_Character(Name) ON DELETE CASCADE
);
```

Weapon

```
CREATE TABLE Weapon (
    item TEXT NOT NULL,
    type TEXT,
    range INTEGER,
    price INTEGER,
    damage_points INTEGER,
    PRIMARY KEY (item)
);
```

Armour

```
CREATE TABLE Armour (
    item TEXT NOT NULL,
    body_part TEXT,
    price INTEGER,
    defence_score INTEGER,
    PRIMARY KEY (item)
);
```

Supply

```
CREATE TABLE Supply (
        item TEXT NOT NULL,
        healing_score INTEGER,
        price INTEGER,
        mana_score INTEGER,
        PRIMARY KEY (item)
);
```

weapon_inv

```
CREATE TABLE weapon_inv (
        quantity INTEGER,
        equipped INTEGER,
        Name TEXT NOT NULL,
        item TEXT NOT NULL,
        PRIMARY KEY (Name, item),
        FOREIGN KEY (Name) REFERENCES Inventory(Name) ON DELETE CASCADE,
        FOREIGN KEY (item) REFERENCES Weapon(item) ON DELETE CASCADE
);
```

armour_inv

```
CREATE TABLE armour_inv (
        quantity INTEGER,
        equipped INTEGER,
        Name TEXT NOT NULL,
        item TEXT NOT NULL,
        PRIMARY KEY (Name, item),
        FOREIGN KEY (Name) REFERENCES Inventory(Name) ON DELETE CASCADE,
        FOREIGN KEY (item) REFERENCES Armour(item) ON DELETE CASCADE
);
```

supply_inv

```
CREATE TABLE supply_inv (
        quantity INTEGER,
        equipped INTEGER,
        NameTEXT NOT NULL,
        item TEXT NOT NULL,
        PRIMARY KEY (Name, item),
        FOREIGN KEY (Name) REFERENCES Inventory(Name) ON DELETE CASCADE,
        FOREIGN KEY (item) REFERENCES Supply(item) ON DELETE CASCADE
);
```

Normalization Comments:

Player

- Player is in 1NF as there are no multivalued attributes and all the values are atomic
- Player is in 2NF as all non-key attributes are dependent on the primary key and we only have 1 primary key

Owns_Character=

- Owns_Character is in 1NF as there are no multivalued attributes and all the values are atomic
- Owns_Character is not in 2NF as some attributes depend on the Character Name while not depending on the AccountNo. For Example, the maxHealth of a Character depends on the Character Name only and not the AccountNo

CombatInfo

- CombatInfo is in 1NF as there are no multivalued attributes and all the values are atomic
- CombatInfo is in 2NF as all non-key attributes are dependent on the primary key and we only have 1 primary key
- CombatInfo is in 3NF because we only have 1 non-key attribute which is the BattleDate and it only depends on the BattleNo Primary Key
- CombatInfo is in BCNF because the BattleNo doesn't depend on the BattleDate

char_combat_status

- char_combat_status is in 1NF as there are no multivalued attributes and all the values are atomic
- char_combat_status is not in 2NF as some attributes depend on the Character Name while not depending on the BattleNo. For Example the Weapon used only depends on the Character Name and doesn't depend on the BattleNo

Inventory

- Inventory is in 1NF as there are no multivalued attributes and all the values are atomic
- Inventory is in 2NF as all non-key attributes are dependent on the primary key and we only have 1 primary key
- Inventory is in 3NF as Inventory has only 1 Attribute which is the Character Name and it is the only primary key so there doesn't exist evey any non-key attributes
- Inventory is in BCNF because there is no non-key attributes

Weapon

- Weapon is in 1NF as there are no multivalued attributes and all the values are atomic
- Weapon is in 2NF as all non-key attributes are dependent on the primary key and we only have 1 primary key
- Weapon is in 3NF as there is no non-key attribute that depends on another non-key attribute all non-key attributes do not depend on each other
- Weapon is in BCNF because there is no non-key attribute that determines the Item Name of the Weapon which is the primary key

Armour

- Armour is in 1NF as there are no multivalued attributes and all the values are atomic
- Armour is in 2NF as all non-key attributes are dependent on the primary key and we only have 1 primary key
- Armour is in 3NF as there is no non-key attribute that depends on another non-key attribute all non-key attributes do not depend on each other
- Armour is in BCNF because there is no non-key attribute that determines the Item Name of the Armour which is the primary key

Supply

- Supply is in 1NF as there are no multivalued attributes and all the values are atomic
- Supply is in 2NF as all non-key attributes are dependent on the primary key and we only have 1 primary key
- Supply is in 3NF as there is no non-key attribute that depends on another non-key attribute all non-key attributes do not depend on each other
- Supply is in BCNF because there is no non-key attribute that determines the Item Name of the Supply which is the primary key

weapon_inv

- weapon_inv is in 1NF as there are no multivalued attributes and all the values are atomic
- weapon_inv is in 2NF as all non-key attributes are dependent on the whole Candidate Keys and there is no non-key attribute that depends on a subset of the Candidate Keys
- weapon_inv is in 3NF as the is no non-key attribute that depends on another non-key attribute all non-key attributes do not depend on each other. And in this entity **quantity** and **equipped** do not depend on each other and they are the only non-key attributes in the entity
- weapon_inv is in BCNF because there is no non-key attribute that determines any values from the Super Keys, in other words, there is no non-key attribute that determines the value of Item Name or Character Name

armour_inv

- armour_inv is in 1NF as there are no multivalued attributes and all the values are atomic
- armour_inv is in 2NF as all non-key attributes are dependent on the whole Candidate Keys and there is no non-key attribute that depends on a subset of the Candidate Keys
- armour_inv is in 3NF as the is no non-key attribute that depends on another non-key attribute all non-key attributes do not depend on each other. And in this entity **quantity** and **equipped** do not depend on each other and they are the only non-key attributes in the entity
- armour_inv is in BCNF because there is no non-key attribute that determines any values from the Super Keys, in other words, there is no non-key attribute that determines the value of Item Name or Character Name

supply_inv

- supply_inv is in 1NF as there are no multivalued attributes and all the values are atomic
- supply_inv is in 2NF as all non-key attributes are dependent on the whole Candidate Keys and there is no non-key attribute that depends on a subset of the Candidate Keys
- supply_inv is in 3NF as the is no non-key attribute that depends on another non-key attribute all non-key attributes do not depend on each other. And in this entity **quantity** and **equipped** do not depend on each other and they are the only non-key attributes in the entity
- supply_inv is in BCNF because there is no non-key attribute that determines any values from the Super Keys, in other words, there is no non-key attribute that determines the value of Item Name or Character Name