MySQL Final Project

Database Design and Analysis - Part 1

Database Project Proposal

Key Dates

- Database Project Proposal - Due April 9 5:30PM
    - Feedback will be provided via coursework's. If required, updated project proposals will be due April 16h 11:59PM.
    - You will have time in class on April 16 to work on your project.

Project Proposal

1. Provide the names and UNIs of your team (2 students max - unless given instructor permission)

   Yixin Zheng: yz4993
   Jingxi Wang: jw4690

2. Please provide a brief description of the database you would like to build and the types of variables you would like to store.

We plan to build a dataset using information from the Stardew Valley Wiki. The dataset will include details on harvestable products in the game, such as name, price, season (start and end dates), and growth time. We'll also include product categories (e.g., dairy, vegetable), foods that can be made from each product (e.g., vegetable medley, hashbrowns), and corresponding recipes, listing ingredients and amounts.

3. If applicable provide any links for the public data you would like to store / model in your final project.

   We insert data by our own, the data are from Stardew Valley Wiki:

   https://stardewvalleywiki.com/Stardew_Valley_Wiki

4. You will be required to create a minimum of 3 tables in your final database. For each table:
    - Specify the table name

- Add appropriate variables to each table following best naming practices
- Choose an appropriate data type for each variable. It should be able to store all plausible values for that variable, while taking the least amount of storage space possible
- Identify primary keys, foreign keys (if applicable), indexes, and other column attributes (NOT NULL, UNSIGNED, DEFAULT ect.)

- Required Table Include:
  - Parent Table
  - Child Table (must be part of a one-to-many relationship with the Parent Table)
  - Lookup Table
- Optional Table(s):
  - Second Child or Grandchild Table

Database

- Schema (database) name: Stardew Valley Cookbook

Table [1]: Food

Field Name (Variable)	**Data Type**	**[PK / CPK] if part of a primary key [FK / CFK] if part of a foreign key [I] if part of a non-unique index [other column attribute(s)]**
food_id	INT AUTO_INCREMENT	PK
food_name	VARCHAR(255)	UI
food_price	SMALLINT (UNSIGNED)	
energy	SMALLINT (UNSIGNED)	

(add more rows as needed)

Table [2]: recipe

Field Name (Variable)	Data Type	[PK / CPK] if part of a primary key [FK / CFK] if part of a foreign key [I] if part of a non-unique index [other column attribute(s)]
recipe_id	INT AUTO_INCREMENT	PK
variant_id	INT	I
food_id	INT	FK
product_id	INT	FK
amount	TINYINT (UNSIGNED)	

(add more rows as needed)

Table [3]: product

Field Name (Variable)	Data Type	[PK / CPK] if part of a primary key [FK / CFK] if part of a foreign key [I] if part of a non-unique index [other column attribute(s)]
product_id	INT AUTO_INCREMENT	PK
category_id	INT	FK
product_name	VARCHAR(255)	UI
product_price	SMALLINT (UNSIGNED)	

(add more rows as needed)

Table [4]: growth_cycle

product_id	INT	PK, FK
season_start_date	DATE	
season_end_date	DATE	
growth_time	TINYINT (UNSIGNED)	note: days until 1st harvest
regrowth_time	TINYINT (UNSIGNED)	note: null if cannot be regrown
max_harvest	TINYINT (UNSIGNED)	note: null if unlimited
yield_per_harvet	TINYINT (UNSIGNED)	note: units produced per harvest
is_perennial	BOOLEAN DEFAULT FALSE	note: TRUE if source continues year-round

Table [5]: category

Field Name (Variable)	**Data Type**	**[PK / CPK] if part of a primary key [FK / CFK] if part of a foreign key [I] if part of a non-unique index [other column attribute(s)]**
category_id	INT AUTO_INCREMENT	PK
category_name	VARCHAR(255)	UI

(add more rows as needed)

Table […] add more tables as needed