Esteban Bravo
Michael Sciancalepore
Gaming Database Report

The database consists of five tables of data about video games. The first table is the game table, which holds the video game's title, the publisher, the year it was released, the genre, the series, and the type of console it's compatible with. The second table consists of the vendor's data, such as their company name, website, phone number, and email address. The third table holds the game developer company data like the company name, the location, the president, and the company's net worth. The fourth table holds the customer information like their last name, first name, phone number, and email address. Finally, the fifth table has the sale information such as the SaleID, Customer ID, VendorID, sale date, subtotal, tax, and total. The IDs of each table act as the primary key. The foreign keys in the database include the CompanyID in the game table and the CustomerID and VendorID in the sale table. With this data, we can analyze or mine different factors such as the increase in video game price or the increase in video game production.

To make the database, Django, Python, and MySQL were used. Django was used to set up and run the database servers and connect the python files to MySQL. MySQL was used as an option to add data to the tables, view the tables, and generate the E-R diagram. The database website was made using python and HTML. HTML was used to set up the text and buttons on each page, and python was used to create URLs, set up the CRUD operation, and activate where all the buttons were redirected. The database's home page has a list of all five tables, and clicking on the respective table redirects the user to the CRUD operations for that table. The create button acts as the Create operation for CRUD, where it creates a new instance of the data. The read operation for CRUD is represented by the home page of each table, where it lists all of the data about each instance. The update button acts as the Update operation for CRUD, allowing you to update the instance of the data click-on. The delete button acts as the Delete operation for CRUD, where it deletes the instance of the data click-on. The update and delete tasks are connected to the ID of the data, so it updates and deletes based on that key. All the IDs auto increment by one from the previous piece of data. The back button on each table takes the user back to the home screen of the database, where they can select which table they want to see. Screenshots of each database page, including each table's CRUD operations, are given below.

Screenshots:
Home Page:



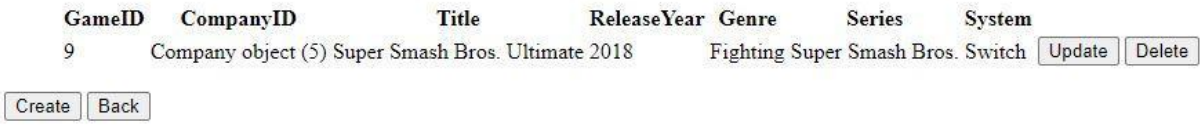
CRUD Screenshots:
Game Table:
Home:



Create:



Read:



Update:

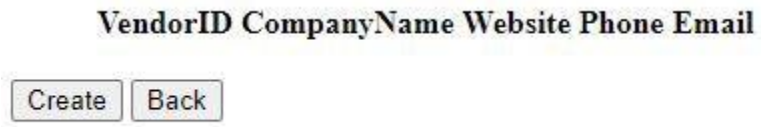


Delete:



Vendor Table:

Home:



Create:



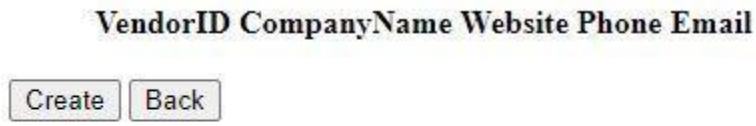
Read:



Update:

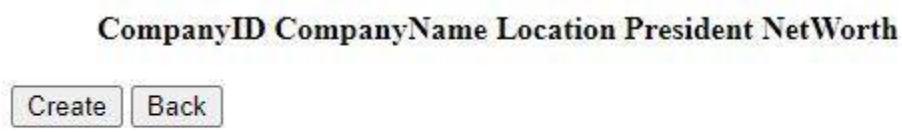


Delete:



Company Table:

Home:



Create:



Read:

CompanyID	CompanyName	Location	President	NetWorth		
6	Nintendo	Redmond, WA	Reggie Fils-Aime	5000000.0	Update	Delete

Create Back

Update:

CompanyName: Nintendo Location: Redmond, WA President: Reggie Fils-Aime NetWorth: 5000000.0 Submit

Delete:

CompanyID	CompanyName	Location	President	NetWorth

Create Back

Customer Table:
Home:

CustomerID	LastName	FirstName	Phone	Email

Create Back

Create:

LastName: FirstName: Phone: Email: Submit

Read:

CustomerID	LastName	FirstName	Phone	Email		
6	Sciancalepore	Michael	203-555-0123	msciancal@hartford.edu	Update	Delete

Create Back

Update:

LastName: Sciancalepore FirstName: Michael Phone: 2034444258 Email: mikeshank8800@gmail.com Submit

Delete:

CustomerID LastName FirstName Phone Email

Create Back

Sale Table:
Home:

SaleID CustomerID VendorID SaleDate SubTotal Tax Total

Create Back

Create:



Read:

SaleID	CustomerID	VendorID	SaleDate	SubTotal	Tax	Total		
4	Customer object (7)	Vendor object (7)	12/5/21	500.0	2.5	502.5	Update	Delete

Create Back

Update:



Delete:

SaleID CustomerID VendorID SaleDate SubTotal Tax Total

Create Back