

Client

GET http://localhost:8000/client/paid_user

http://localhost:8000/client/paid_user

Get information on all paid users

BODY urlencoded

u_id	1
user_name	coolguy
password	password

GET http://localhost:8000/client/paid_user/:id

http://localhost:8000/client/paid_user/:id

Get information on a paid user based on user id

PATH VARIABLES

id	1
	The integer ID of the paid user

BODY urlencoded

u_id	1
user_name	coolguy
password	password

DEL http://localhost:8000/client/paid_user/:id

http://localhost:8000/client/paid_user/:id

Delete user from database by entering user ID

PATH VARIABLES

id	2
	The integer ID of the paid user

BODY urlencoded

u_id	1
user_name	coolguy
password	password

GET http://localhost:8000/client/f2pclient

http://localhost:8000/client/f2pclient

Get User ID of "free to play" users

BODY urlencoded

u_id	1
user_name	coolguy
password	password

Example Request

http://localhost:8000/client/paid_us

```
curl --location --request GET 'http://localhost:8000/client/paid_user'
```

Example Response

200 OK

```
Body Headers (7)
{
  "message": "success",
  "data": [
    {
      "U_ID": 2,
      "USER_NAME": "2",
      "PASSWORD": "b97488b4e0b8078c3b95cdec5a107e00"
    }
  ],
  {
    "HTTP": 1.
  }
}
```

View More

Example Request

http://localhost:8000/client/paid_user/

```
curl --location --request GET 'http://localhost:8000/client/paid_user/1'
```

Example Response

200 OK

```
Body Headers (7)
{
  "data": {
    "U_ID": 1,
    "USER_NAME": "idk",
    "PASSWORD": "b97488b4e0b8078c3b95cdec5a107e00"
  }
}
```

Example Request

http://localhost:8000/client/paid_user/

```
curl --location --request DELETE 'http://localhost:8000/client/paid_user/2'
```

Example Response

200 OK

```
Body Headers (7)
{
  "message": "deleted",
  "rows": 0
}
```

Example Request

http://localhost:8000/client/f2pcli

```
curl --location --request GET 'http://localhost:8000/client/f2pclient'
```

Example Response

200 OK

```
Body Headers (7)
{
  "message": "success",
  "data": [
    {
      "U_ID": 3
    }
  ]
}
```

POST http://localhost:8000/developer/:dname/dlocation_table/

http://localhost:8000/developer/:dname/dlocation_table/

Creating new location for Publisher with foreign key = param

PATH VARIABLES	
dname	Obsidian Studios
BODY urlencoded	
DName	Obsidian Studios
location	Paris, France

GET http://localhost:8000/developer/:dname/dlocation_table/

http://localhost:8000/developer/:dname/dlocation_table/

getting all Locations for Publisher with foregin key = param

PATH VARIABLES	
dname	Obsidian Studios
BODY urlencoded	
DName	Obsidian Studios
Location	Paris, France

DEL http://localhost:8000/developer/:dname/dlocation_table/:location

http://localhost:8000/developer/:dname/dlocation_table/:location

Deleting a Location for Publisher where foreignkey = param AND location = param

PATH VARIABLES	
dname	Obsidian Studios
location	Paris, France
BODY urlencoded	
DName	Obsidian Studios
Location	Paris, France

YOUT Double Column LANGUAGE cURL - cURL

POST http://localhost:8000/developer/:dname/develops/

http://localhost:8000/developer/:dname/develops/

Connecting Developer with a Video_Game using Foreign key dname = param

PATH VARIABLES	
dname	Obsidian Studios
BODY urlencoded	
DName	Obsidian Studios
Location	Paris, France
v_id	2

GET http://localhost:8000/developer/:dname/develops

http://localhost:8000/developer/:dname/develops

Getting all Video game developed by Developer with foreignkey DName = param

PATH VARIABLES	
dname	Obsidian Studios
BODY urlencoded	

Example Request http://localhost:8000/developer/:dname/dlocation_table/

```
curl --location --request POST 'http://localhost:8000/developer/Obsidian Studios/dlocation_table/' \
--data-urlencode 'location=Paris, France'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": {
    "u_id": "Obsidian Studios",
    "location": "Paris, France"
  }
}
```

Example Request http://localhost:8000/developer/:dname/dlocation_table/

```
curl --location --request GET 'http://localhost:8000/developer/Obsidian Studios/dlocation_table/' \
--data-urlencode 'Location=Paris, France'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": [
    {
      "DNAME": "Obsidian Studios",
      "LOCATION": "Paris, France"
    }
  ]
}
```

Example Request http://localhost:8000/developer/:dname/dlocation_table/:location

```
curl --location --request DELETE 'http://localhost:8000/developer/Obsidian Studios/dlocation_table/Paris, France' \
--data-urlencode 'Location=Paris, France'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "deleted",
  "rows": 1
}
```

Example Request http://localhost:8000/developer/:dname/develops/

```
curl --location --request POST 'http://localhost:8000/developer/Obsidian Studios/develops/' \
--data-urlencode 'v_id=2'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "U_ID": {
    "v_id": "2",
    "dname": "Obsidian Studios"
  }
}
```

Example Request http://localhost:8000/developer/:dname/develops

```
curl --location --request GET 'http://localhost:8000/developer/Obsidian Studios/develops'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": {
    "V_ID": 2,
    "DNAME": "Obsidian Studios",
    "DESCRIPTION": "Dota 2 is a multiplayer online battle arena (MOBA) video game in which two teams of five players compete to destroy the enemy team's base.",
    "VNAME": "Dota 2",
    "RELEASE_STATUS": 1
  }
}
```

Video Game, Similar Games, Reviews and Operating Platform

POST http://localhost:8000/Video_Game/

http://localhost:8000/Video_Game/

Adding new Video_Game row with data inserted in body

BODY urlencoded	
description	Delve yourself in the world of Cyber Punk 77
vname	Cyber Punk 77
rs	1

GET http://localhost:8000/Video_Game/

http://localhost:8000/Video_Game/

Getting all video games in the table

GET http://localhost:8000/Video_Game/v_id/

http://localhost:8000/Video_Game/v_id/

Getting Video game using Primary Key = param

PATH VARIABLES	
v_id	3

POST http://localhost:8000/Video_Game/Similar_To/v_id/

http://localhost:8000/Video_Game/Similar_To/v_id/

Linking a Video game to another with the Similar to Table using the Video Game ID of the original and the one that is similar to it

PATH VARIABLES	
v_id	3

BODY urlencoded	
sim_id	4

GET http://localhost:8000/Video_Game/Similar_To/v_id

http://localhost:8000/Video_Game/Similar_To/v_id

Getting all the similar games to the game with v_id = param

PATH VARIABLES	
v_id	3

Example Request http://localhost:8000/developer/dname/develop

```
curl --location --request POST 'http://localhost:8000/Video_Game/' \
--data-urlencode 'description=Delve yourself in the world of Cyber Punk 77' \
--data-urlencode 'vname=Cyber Punk 77' \
--data-urlencode 'rs=1'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": {
    "description": "Delve yourself in the world of Cyber Punk 77",
    "vname": "Cyber Punk 77",
    "rs": "1"
  }
}
```

Example Request http://localhost:8000/Video_Game/v_id/

```
curl --location --request GET 'http://localhost:8000/Video_Game/'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": [
    {
      "V_ID": 1,
      "DESCRIPTION": "League of Legends is a team-based strategy game where two teams of five powerful champions face each other in a 5v5 battle.",
      "VNAME": "League of Legends",
      "RELEASE_STATUS": 1
    }
  ],
  "count": 1
}
```

View More

Example Request http://localhost:8000/Video_Game/v_id/

```
curl --location --request GET 'http://localhost:8000/Video_Game/3/'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": {
    "V_ID": 3,
    "DESCRIPTION": "Delve yourself in the world of Cyber Punk 77",
    "VNAME": "Cyber Punk 77",
    "RELEASE_STATUS": 1
  }
}
```

Example Request http://localhost:8000/Video_Game/Similar_To/v_id/

```
curl --location --request POST 'http://localhost:8000/Video_Game/Similar_To/3/' \
--data-urlencode 'sim_id=4'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": {
    "v_id": "3",
    "sim_id": "4"
  }
}
```

Example Request http://localhost:8000/Video_Game/Similar_To/v_id

```
curl --location --request GET 'http://localhost:8000/Video_Game/Similar_To/3'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": {
    "V_ID": 3,
    "SIM_ID": 4
  }
}
```

Esports and Has

GET http://localhost:8000/esport/

http://localhost:8000/esport/

Getting all Esport values

POST http://localhost:8000/esport/

http://localhost:8000/esport/

Posting new Esport scene with the body being Non-null

BODY urlencoded

league	LEC
--------	-----

DEL http://localhost:8000/esport/:league

http://localhost:8000/esport/:league

Deleting a League row with Primary key = param

PATH VARIABLES

league	LEC
--------	-----

Example Request http://localhost:8000/esport/

```
curl --location --request GET 'http://localhost:8000/esport/'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": [
    {
      "LEAGUE": "LCK"
    },
    {
      "LEAGUE": "LCS"
    }
  ]
}
```

View More

Example Request http://localhost:8000/esport/

```
curl --location --request POST 'http://localhost:8000/esport/' \
--data-urlencode 'league=LEC'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": "LEC"
}
```

Example Request http://localhost:8000/esport/:league

```
curl --location --request DELETE 'http://localhost:8000/esport/LEC'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "deleted",
  "rows": 1
}
```

GET http://localhost:8000/Team/:tname/employs

http://localhost:8000/Team/:tname/employs

Gets all players on the Team where tname = param

PATH VARIABLES

tname Team Liquid

AYOUT Double Column LANGUAGE cURL - cURL

GET http://localhost:8000/Team/employs

http://localhost:8000/Team/employs

Gets all values in the Employs table

DEL http://localhost:8000/Team/:TName/employs

http://localhost:8000/Team/:TName/employs

Deletes a employs row where TName = param and PlayerName = body

PATH VARIABLES

TName Team Liquid

BODY urlencoded

PlayerName MaisterGW

Example Requesthttp://localhost:8000/Team/:tname/employs

curl --location --request GET 'http://localhost:8000/Team/Team Liquid/employs'

Example Response200 OK

BodyHeaders (7)

{
 "data": {
 "TNAME": "Team Liquid",
 "PLAYERNAME": "MaisterGW",
 "YEAR": 2019,
 "MONTH": 2,
 "DAY": 22,
 "AGE": 24,
 "NATIONALITY": "Mexican",
 "DESCRIPTION": "Mexican Smash Brothers Pro. No team but hsi GW skills can even topple over Pro Player Titans."
 }
}

Example Requesthttp://localhost:8000/Team/employs

curl --location --request GET 'http://localhost:8000/Team/employs'

Example Response200 OK

BodyHeaders (7)

{
 "message": "success",
 "data": [
 {
 "TNAME": "TSM",
 "PLAYERNAME": "Søren Bjerg",
 "YEAR": 2000,
 "MONTH": 2,
 "DAY": 22,
 "AGE": 24,
 "NATIONALITY": "Danish",
 "DESCRIPTION": "Danish Smash Brothers Pro. No team but hsi GW skills can even topple over Pro Player Titans."
 }
]
}

View More

Example Requesthttp://localhost:8000/Team/:TName/employs

curl --location --request DELETE 'http://localhost:8000/Team/Team Liquid/employs' \
--data-urlencode 'PlayerName=MaisterGW'

Example Response200 OK

BodyHeaders (7)

{
 "message": "deleted",
 "rows": 1
}

IGDB

PATCH Update Game Status

http://localhost:8000/Video_Game/:v_id

PATH VARIABLES

v_id	1
-------------	---

BODY urlencoded

rs	0
-----------	---

Example Request

Update Game Status

```
curl --location --request PATCH 'http://localhost:8000/Video_Game/1' \  
--data-urlencode 'rs=0'
```

GET http://localhost:8000/publisher/plocation_table/

http://localhost:8000/publisher/plocation_table/

Gets all PLocation_Table values from the Table

BODY urlencoded

pname Ubisoft
location Montreal

GET http://localhost:8000/publisher/:pname/plocation_table/

http://localhost:8000/publisher/:pname/plocation_table/

Gets all PLocation_Values with foreign key PName = Ubisoft

PATH VARIABLES

pname Ubisoft

BODY urlencoded

pname Ubisoft
The Publisher Name, used for finding games published by this Publisher

POST http://localhost:8000/publisher/:pname/publishes/

http://localhost:8000/publisher/:pname/publishes/

Adds new Publishes row with the foreign Key pname = ubisoft

PATH VARIABLES

pname Ubisoft

BODY urlencoded

pname Ubisoft
The Publisher Name, used for finding games published by this Publisher

v_id 1

GET http://localhost:8000/publisher/publishes/

http://localhost:8000/publisher/publishes/

Gets all values of Publishes in the table

BODY urlencoded

pname Ubisoft
The Publisher Name, used for finding games published by this Publisher

v_id 1

GET http://localhost:8000/publisher/:pname/publishes/

http://localhost:8000/publisher/:pname/publishes/

Gets all Publishes value and the Video Game value for Publisher with the foreign key pname = ubisoft. can specify which video game to show using v_id body request

PATH VARIABLES

pname Ubisoft

BODY urlencoded

pname Ubisoft
The Publisher Name, used for finding games published by this Publisher

v_id 1

DEL http://localhost:8000/publisher/:pname/publishes/

http://localhost:8000/publisher/:pname/publishes/

Deletes rows from Publisher where foreign key pname = Ubisoft. Can delete a specific Publishes row by once again using v_id as well

PATH VARIABLES

pname Ubisoft

BODY urlencoded

pname Ubisoft
The Publisher Name, used for finding games published by this Publisher

Example Request

http://localhost:8000/publisher/plocation_table/

curl --location --request GET 'http://localhost:8000/publisher/plocation_table/'

Example Response

200 O

Body Headers [7]

```
{
  "message": "success",
  "data": [
    {
      "PNAME": "Riot Games",
      "LOCATION": "West Los Angeles, California, United States"
    },
    {
      "PNAME": "Valve Corporation",
      "LOCATION": "Bellevue, Washington, United States"
    }
  ]
}
```

View More

Example Request

http://localhost:8000/publisher/:pname/plocation_table/

curl --location --request GET 'http://localhost:8000/publisher/Ubisoft/plocation_table/' \
--data-urlencode 'pname=Ubisoft'

Example Response

200 O

Body Headers [7]

```
{
  "message": "success",
  "data": {
    "PNAME": "Ubisoft",
    "LOCATION": "Montreal"
  }
}
```

Example Request

http://localhost:8000/publisher/:pname/publishes/

curl --location --request POST 'http://localhost:8000/publisher/Ubisoft/publishes/' \
--data-urlencode 'v_id=1'

Example Response

200 O

Body Headers [7]

```
{
  "message": "success",
  "data": {
    "pname": "Ubisoft",
    "v_id": "1"
  }
}
```

Example Request

http://localhost:8000/publisher/publishes/

curl --location --request GET 'http://localhost:8000/publisher/publishes/'

Example Response

200 O

Body Headers [7]

```
{
  "message": "success",
  "data": [
    {
      "PNAME": "Riot Games",
      "V_ID": 1,
      "DESCRIPTION": "League of Legends is a team-based strategy game where two teams of five powerful champions face off"
    }
  ]
}
```

Example Request

http://localhost:8000/publisher/:pname/publishes/

curl --location --request GET 'http://localhost:8000/publisher/Ubisoft/publishes/' \
--data-urlencode 'v_id=1'

Example Response

200 O

Body Headers [7]

```
{
  "data": {
    "PNAME": "Ubisoft",
    "V_ID": 1,
    "DESCRIPTION": "League of Legends is a team-based strategy game where two teams of five powerful champions face off",
    "VNAME": "League of Legends",
    "RELEASE_STATUS": 1
  }
}
```

Example Request

http://localhost:8000/publisher/:pname/publishes/

curl --location --request DELETE 'http://localhost:8000/publisher/Ubisoft/publishes/' \
--data-urlencode 'v_id=1'

Example Response

200 O

Body Headers [7]

```
{
  "message": "deleted",
  "rows": 1
}
```

Developer related Classes and Entities

POST http://localhost:8000/Developer/

http://localhost:8000/Developer/

Used to create new Developer with Primary Key parameter

BODY urlencoded	
DName	Obsidian Studios

GET http://localhost:8000/Developer/

http://localhost:8000/Developer/

getting all developer from the table

BODY urlencoded	
DName	Obsidian Studios

GET http://localhost:8000/Developer/:dname/

http://localhost:8000/Developer/:dname/

getting the developer with param = primary key

PATH VARIABLES	
dname	Obsidian Studios
BODY urlencoded	
DName	Obsidian Studios

DEL http://localhost:8000/Developer/:dname/

http://localhost:8000/Developer/:dname/

Deleting developer with param = primary key

PATH VARIABLES	
dname	Obsidian Studios
BODY urlencoded	
DName	Obsidian Studios

Example Request http://localhost:8000/Developer/

```
curl --location --request POST 'http://localhost:8000/Developer/' \
--data-urlencode 'DName=Obsidian Studios'
```

Example Response 200 OK

Body Headers (7)

```
{
  "DName": "Obsidian Studios"
}
```

Example Request http://localhost:8000/Developer/

```
curl --location --request GET 'http://localhost:8000/Developer/' \
--data-urlencode 'DName=Obsidian Studios'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": [
    {
      "DNAME": "Riot Games"
    },
    {
      "DNAME": "Valve Corporation"
    },
    {
      "DNAME": "Obsidian Studios"
    }
  ]
}
```

View More

Example Request http://localhost:8000/Developer/:dname/

```
curl --location --request GET 'http://localhost:8000/Developer/Obsidian Studios/' \
--data-urlencode 'DName=Obsidian Studios'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": {
    "DNAME": "Obsidian Studios"
  }
}
```

Example Request http://localhost:8000/Developer/:dname/

```
curl --location --request DELETE 'http://localhost:8000/Developer/Obsidian Studios/' \
--data-urlencode 'DName=Obsidian Studios'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "deleted",
  "rows": 1
}
```


Team, Employs

POST http://localhost:8000/Team/

http://localhost:8000/Team/

Creating a new Team into the Database

BODY urlencoded

tname	Team Liquid
description	Founded by Steve, not the best team but one of the most branded and riches orgs in the esports scene
year	2015
month	June
day	30

GET http://localhost:8000/Team/

http://localhost:8000/Team/

Gets all the Team in the Database

Example Request http://localhost:8000/Team/

```
curl --location --request POST 'http://localhost:8000/Team/' \
--data-urlencode 'tname=Team Liquid' \
--data-urlencode 'description=Founded by Steve, not the best team but one of the most branded and riches orgs in the esports scene' \
--data-urlencode 'year=2015' \
--data-urlencode 'month=June' \
--data-urlencode 'day=30'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": {
    "tname": "Team Liquid",
    "description": "Founded by Steve, not the best team but one of the most branded and riches orgs in the esports scene ",
    "year": "2015",
    "month": "June",
    "day": "30"
  }
}
```

Example Request http://localhost:8000/Team/

```
curl --location --request GET 'http://localhost:8000/Team/'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": [
    {
      "TNAME": "TSM",
      "DESCRIPTION": "TSM (previously known as Team SoloMid) is a North American esports team, created around the community of players who play League of Legends.",
      "YEAR": 2009,
      "MONTH": 9,
      "DAY": 1
    }
  ]
}
```

View More

DEL http://localhost:8000/client/f2pclient

http://localhost:8000/client/f2pclient

Delete free to play user

BODY urlencoded

u_id 3

POST http://localhost:8000/client/f2pclient/

http://localhost:8000/client/f2pclient/

Add new free to play user

BODY urlencoded

u_id 5

GET http://localhost:8000/client

http://localhost:8000/client/id

Add client to database

PATH VARIABLES

id 1

BODY urlencoded

GET http://localhost:8000/client/:id

http://localhost:8000/client/:id

Get user information from user id

Layout Double Column Language cURL - cURL

id 2

BODY urlencoded

POST http://localhost:8000/client/

http://localhost:8000/client/

Add client to database

BODY urlencoded

u_id 8
integer user id

DEL http://localhost:8000/client/:id

http://localhost:8000/client/:id

Delete user from database

PATH VARIABLES

id 8

BODY urlencoded

u_id integer user id

Example Request http://localhost:8000/client/f2pclient

```
curl --location --request DELETE 'http://localhost:8000/client/f2pclient' \
--data-urlencode 'u_id=3'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "deleted",
  "rows": 0
}
```

Example Request http://localhost:8000/client/f2pclient/

```
curl --location --request POST 'http://localhost:8000/client/f2pclient/' \
--data-urlencode 'u_id=5'
```

Example Request http://localhost:8000/client/

```
curl --location --request GET 'http://localhost:8000/client/1'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": {
    "U_ID": 1
  }
}
```

Example Request http://localhost:8000/client/:id

```
curl --location --request GET 'http://localhost:8000/client/2'
```

Example Response 200 OK

Body Headers (7)

```
"data": {
  "U_ID": 2
}
```

Example Request http://localhost:8000/client/

```
curl --location --request POST 'http://localhost:8000/client/' \
--data-urlencode 'u_id=8'
```

Example Response 200 OK

Body Headers (7)

```
{
  "U_ID": {}
}
```

Example Request http://localhost:8000/client/:id

```
curl --location --request DELETE 'http://localhost:8000/client/8' \
--data-urlencode 'u_id=8'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "deleted",
  "rows": 0
}
```

PATCH Update Client Username

```
http://localhost:8000/client/paid_user/:id
```

PATH VARIABLES

id	1
-----------	---

BODY urlencoded

username	Andrea
-----------------	--------

Example Request

Update Client Username

```
curl --location --request PATCH 'http://localhost:8000/client/paid_user/1' \  
--data-urlencode 'username=Andrea'
```

DEL http://localhost:8000/Video_Game/:v_id/has

http://localhost:8000/Video_Game/:v_id/has

Deletes a Has row with v_id = param and body = league

PATH VARIABLES

v_id	1
------	---

BODY urlencoded

league	LEC
--------	-----

Example Request

http://localhost:8000/Video_Game/:v_id/has

```
curl --location --request DELETE 'http://localhost:8000/Video_Game/1/has' \
--data-urlencode 'league=LEC'
```

Example Response

200 OK

Body Headers (7)

```
{
  "message": "deleted",
  "rows": 1
}
```

GET http://localhost:8000/Team/:tname

http://localhost:8000/Team/:tname

Gets the Team and it's values where tname = param

PATH VARIABLES

tname Team Liquid

DEL http://localhost:8000/Team/:tname/

http://localhost:8000/Team/:tname/

Deletes a Team where tname = param

PATH VARIABLES

tname Team Liquid

POST http://localhost:8000/Team/:tname/employs

http://localhost:8000/Team/:tname/employs

Creates a new Employment row in Employs where Tname = param and body PlayerName != null and org_less_flag = 0. Also will trigger the Patch method for Player where PlayerName = body

PATH VARIABLES

tname Team Liquid

BODY urlencoded

PlayerName MaisterGW

Year 2019

Month 9

Day 15

Example Request http://localhost:8000/Team/:tname

curl --location --request GET 'http://localhost:8000/Team/Team Liquid'

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": [
    {
      "TNAME": "Team Liquid",
      "DESCRIPTION": "Founded by Steve, not the best team but one of the most branded and riches orgs in the esports scene",
      "YEAR": 2015,
      "MONTH": "June",
      "DAY": 30
    }
  ]
}
```

View More

Example Request http://localhost:8000/Team/:tname/

curl --location --request DELETE 'http://localhost:8000/Team/Team Liquid/'

Example Response 200 OK

Body Headers (7)

```
{
  "message": "deleted",
  "rows": 1
}
```

Example Request http://localhost:8000/Team/:tname/employs

```
curl --location --request POST 'http://localhost:8000/Team/Team Liquid/employs' \
--data-urlencode 'PlayerName=MaisterGW' \
--data-urlencode 'Year=2019' \
--data-urlencode 'Month=9' \
--data-urlencode 'Day=15'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": {
    "TName": "Team Liquid",
    "PlayerName": "MaisterGW",
    "Year": "2019",
    "Month": "9",
    "Day": "15"
  }
}
```

POST http://localhost:8000/Video_Game/v_id/Review

http://localhost:8000/Video_Game/v_id/Review

Creates new User Rating for a video game using the V_id = param and u_id = body

PATH VARIABLES	
v_id	2
BODY urlencoded	
U_ID	2
Rating	8

GET http://localhost:8000/Video_Game/v_id/Review

http://localhost:8000/Video_Game/v_id/Review

Using the v_id = param to search for reviews for that game id

PATH VARIABLES	
v_id	2
BODY urlencoded	
U_ID	2
Rating	8

GET http://localhost:8000/Video_Game/Review

http://localhost:8000/Video_Game/Review

Gets all review there are in the system, (only one exist ATM)

DEL http://localhost:8000/Video_Game/v_id/Review

http://localhost:8000/Video_Game/v_id/Review

Deletes a review where v_id = param and u_id = body

PATH VARIABLES	
v_id	2
BODY urlencoded	
U_ID	2

POST http://localhost:8000/Video_Game/v_id/operating_platform

http://localhost:8000/Video_Game/v_id/operating_platform

Creates new Operating platform and links to the Video game with v_id = param

PATH VARIABLES	
v_id	2
BODY urlencoded	
platform	Google Stadia

GET http://localhost:8000/Video_Game/v_id/operating_platform

http://localhost:8000/Video_Game/v_id/operating_platform

Gets all operating platform for that Video game with v_id = param

PATH VARIABLES	
v_id	2

Example Request http://localhost:8000/Video_Game/v_id/Review

```
curl --location --request POST 'http://localhost:8000/Video_Game/2/Review' \
--data-urlencode 'U_ID=2' \
--data-urlencode 'Rating=8'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": {
    "U_ID": "2",
    "v_id": "2",
    "Rating": "8"
  }
}
```

Example Request http://localhost:8000/Video_Game/v_id/Review

```
curl --location --request GET 'http://localhost:8000/Video_Game/2/Review'
```

Example Response 200 OK

Body Headers (7)

```
{
  "data": [
    {
      "U_ID": 2,
      "V_ID": 2,
      "RATING": 8,
      "USER_NAME": "2",
      "PASSWORD": "b97488b4e0b8b78c3b95cdec5a107e00"
    }
  ]
}
```

View More

Example Request http://localhost:8000/Video_Game/Review

```
curl --location --request GET 'http://localhost:8000/Video_Game/Review'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": [
    {
      "U_ID": 2,
      "V_ID": 2,
      "RATING": 8,
      "USER_NAME": "2",
      "PASSWORD": "b97488b4e0b8b78c3b95cdec5a107e00"
    }
  ]
}
```

View More

Example Request http://localhost:8000/Video_Game/v_id/Review

```
curl --location --request DELETE 'http://localhost:8000/Video_Game/2/Review' \
--data-urlencode 'U_ID=2'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "deleted",
  "rows": 1
}
```

Example Request http://localhost:8000/Video_Game/v_id/operating_platform

```
curl --location --request POST 'http://localhost:8000/Video_Game/2/operating_platform' \
--data-urlencode 'platform=Google Stadia'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": {
    "v_id": "2",
    "platform": "Google Stadia"
  }
}
```

Example Request http://localhost:8000/Video_Game/v_id/operating_platform

```
curl --location --request GET 'http://localhost:8000/Video_Game/2/operating_platform'
```

Example Response 200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": [
    {
      "V_ID": 2,
      "PLATFORM": "Google Stadia"
    }
  ]
}
```

POST http://localhost:8000/Video_Game/:v_id/has

http://localhost:8000/Video_Game/v_id/has

Posting new has value that links a Video Game to a Esport league using V_id as key

PATH VARIABLES

v_id 2

BODY urlencoded

league LEC

genre MOBA

GET http://localhost:8000/Video_Game/has

http://localhost:8000/Video_Game/has

Getting all has table values

Example Request

http://localhost:8000/Video_Game/v_id/has

```
curl --location --request POST 'http://localhost:8000/Video_Game/2/has' \
--data-urlencode 'league=LEC' \
--data-urlencode 'genre=MOBA'
```

Example Request

http://localhost:8000/Video_Game/has

```
curl --location --request GET 'http://localhost:8000/Video_Game/has'
```

Example Response

200 OK

Body Headers (7)

```
{
  "message": "success",
  "data": [
    {
      "V_ID": 1,
```

```
    },
    {
      "V_ID": 1,
```

View More

Example Request

http://localhost:8000/Video_Game/v_id/has

```
curl --location --request GET 'http://localhost:8000/Video_Game/1/has'
```

Example Response

200 OK

Body Headers (7)

```
{
  "data": [
    {
      "V_ID": 1,
      "LEAGUE": "LCS",
      "GENRE": "Entertainment"
    },
    {
      "V_ID": 1,
      "LEAGUE": "LCK",
```

View More

Publisher Related Entities

Publisher, Publishes, PLocation_Table

POST http://localhost:8000/publisher/

http://localhost:8000/publisher/

BODY urlencoded

pname	Ubisoft
	The Publisher Name, used for finding games published by this Publisher

GET http://localhost:8000/publisher/

http://localhost:8000/publisher/

Gets all Publisher from the table

BODY urlencoded

pname	Ubisoft
	The Publisher Name, used for finding games published by this Publisher

GET http://localhost:8000/publisher/:pname/?pname=Ubisoft

http://localhost:8000/publisher/:pname/?pname=Ubisoft

Gets Publisher with key pname = Ubisoft

PARAMS

pname	Ubisoft
-------	---------

PATH VARIABLES

pname	Ubisoft
-------	---------

BODY urlencoded

pname	Ubisoft
	The Publisher Name, used for finding games published by this Publisher

POST http://localhost:8000/publisher/:pname/plocation_table/?pname=Ubisoft

http://localhost:8000/publisher/:pname/plocation_table/?pname=Ubisoft

Adds a Location with foreign key Pname

PARAMS

pname	Ubisoft
-------	---------

PATH VARIABLES

pname	Ubisoft
-------	---------

BODY urlencoded

pname	Ubisoft
location	Montreal

Example Request

http://localhost:8000/publish

```
curl --location --request POST 'http://localhost:8000/publisher/' \
--data-urlencode 'pname=Ubisoft'
```

Example Response

200

Body Headers (7)

```
{
  "message": "success",
  "data": {
    "pname": "Ubisoft"
  }
}
```

Example Request

http://localhost:8000/publish

```
curl --location --request GET 'http://localhost:8000/publisher/'
```

Example Response

200

Body Headers (7)

```
{
  "message": "success",
  "data": [
    {
      "PNAME": "Riot Games"
    },
    {
      "PNAME": "Valve Corporation"
    },
    {

```

View More

Example Request

http://localhost:8000/publisher/:pname/?pname=Ubis

```
curl --location --request GET 'http://localhost:8000/publisher/Ubisoft/?pname=Ubisoft' \
--data-urlencode 'pname=Ubisoft'
```

Example Response

200

Body Headers (7)

```
{
  "data": {
    "PNAME": "Ubisoft"
  }
}
```

Example Request

http://localhost:8000/publisher/:pname/plocation_table/?pname=Ubis

```
curl --location --request POST 'http://localhost:8000/publisher/Ubisoft/plocation_table/?pname=Ubisoft' \
--data-urlencode 'location=Quebec'
```

Example Response

200

Body Headers (7)

```
{
  "message": "success",
  "data": {
    "pname": "Ubisoft",
    "location": "Quebec"
  }
}
```


Competition, CLocation_Table

POST http://localhost:8000/Competition/

http://localhost:8000/Competition/

Creating new Competition where league != null

BODY: urlencoded	
cname	LCS Sprint PlayOffs
description	The Spring Split play off for the LCS, top 8 teams in the league duke it out in a double elimination bracket to see who the Spring LCS Champions are
league	LCS

GET http://localhost:8000/Competition/

http://localhost:8000/Competition/

Gets all Competition in the Database

GET http://localhost:8000/Competition/cname

http://localhost:8000/Competition/cname

Gets all competition where cname = param

PATH VARIABLES

cname	LCS Spring PlayOffs
-------	---------------------

POST http://localhost:8000/Competition/cname/clocation_table

http://localhost:8000/Competition/cname/clocation_table

Creates new Location for the Competition where cname = param, league = body, and league != null

PATH VARIABLES

cname	LCS Sprint PlayOffs
-------	---------------------

BODY: urlencoded

location	Maddison Square Garden, Toronto
league	LCS

GET http://localhost:8000/Competition/clocation_table

http://localhost:8000/Competition/clocation_table

Gets all Locations of Competitions in the database

GET http://localhost:8000/Competition/cname/clocation_table

http://localhost:8000/Competition/cname/clocation_table

Gets all Location for the competition where cname = param

ENVIRONMENT No environment LAYOUT Double Column LANGUAGE JavaScript - Fetch

COMPETITION, CLOCATION_TABLE

Introduction

- POST http://localhost:8000/Competition/
- GET http://localhost:8000/Competition/
- GET http://localhost:8000/Competition/...
- POST http://localhost:8000/Competition/...
- GET http://localhost:8000/Competition/...
- GET http://localhost:8000/Competition/...
- DEL http://localhost:8000/Competition/...

DEL http://localhost:8000/Competition/cname/clocation_table

http://localhost:8000/Competition/cname/clocation_table

Deletes a location in the database where cname = param and location = body

PATH VARIABLES

cname	LCS
-------	-----

BODY: urlencoded

	Maddison Square Garden, Toronto
--	---------------------------------

Example Request http://localhost:8000/Competition/

```
var urlencoded = new URLSearchParams();
urlencoded.append("cname", "LCS Sprint PlayOffs");
urlencoded.append("description", "The Spring Split play off for the LCS, top 8 teams in the league duke it out in a double elimination bracket to see who the Spring LCS Champions are");
urlencoded.append("league", "LCS");

var requestOptions = {
  method: 'POST',
  body: urlencoded,
  redirect: 'follow'
};
```

Example Response 200 OK

```
Body Headers ()

{
  "message": "success",
  "data": {
    "cname": "LCS Sprint PlayOffs",
    "description": "The Spring Split play off for the LCS, top 8 teams in the league duke it out in a double elimination bracket to see who the Spring LCS Champions are",
    "league": "LCS"
  }
}
```

Example Request http://localhost:8000/Competition/

```
var requestOptions = {
  method: 'GET',
  redirect: 'follow'
};

fetch("http://localhost:8000/Competition/", requestOptions)
.then(response => response.text())
.then(result => console.log(result))
.catch(error => console.log("error", error));
```

Example Response 200 OK

```
Body Headers ()

{
  "message": "success",
  "data": {
    "cname": "LCS Sprint PlayOffs",
    "description": "The Spring Split play off for the LCS, top 8 teams in the league duke it out in a double elimination bracket to see who the Spring LCS Champions are",
    "league": "LCS"
  }
}
```

Example Request http://localhost:8000/Competition/cname

```
var requestOptions = {
  method: 'GET',
  redirect: 'follow'
};

fetch("http://localhost:8000/Competition/LCS Sprint PlayOffs", requestOptions)
.then(response => response.text())
.then(result => console.log(result))
.catch(error => console.log("error", error));
```

Example Response 200 OK

```
Body Headers ()

{
  "message": "success",
  "data": {
    "cname": "LCS Sprint PlayOffs",
    "description": "The Spring Split play off for the LCS, top 8 teams in the league duke it out in a double elimination bracket to see who the Spring LCS Champions are",
    "league": "LCS"
  }
}
```

Example Request http://localhost:8000/Competition/cname/clocation_table

```
var urlencoded = new URLSearchParams();
urlencoded.append("location", "Maddison Square Garden, Toronto");
urlencoded.append("league", "LCS");

var requestOptions = {
  method: 'POST',
  body: urlencoded,
  redirect: 'follow'
};
```

Example Response 200 OK

```
Body Headers ()

{
  "message": "success",
  "data": {
    "location": "Maddison Square Garden, Toronto",
    "cname": "LCS Sprint PlayOffs",
    "league": "LCS"
  }
}
```

Example Request http://localhost:8000/Competition/clocation_table

```
var requestOptions = {
  method: 'GET',
  redirect: 'follow'
};

fetch("http://localhost:8000/Competition/clocation_table", requestOptions)
.then(response => response.text())
.then(result => console.log(result))
.catch(error => console.log("error", error));
```

Example Response 200 OK

```
Body Headers ()

{
  "message": "success",
  "data": {
    "location": "San Francisco",
    "cname": "LCS Sprint PlayOffs",
    "league": "LCS"
  }
}
```

Example Request http://localhost:8000/Competition/cname/clocation_table

```
var requestOptions = {
  method: 'GET',
  redirect: 'follow'
};
```

Example Response 200 OK

```
Body Headers ()

{
  "message": "success",
  "data": {
    "location": "Maddison Square Garden, Toronto",
    "cname": "LCS Sprint PlayOffs",
    "league": "LCS"
  }
}
```

Example Request http://localhost:8000/Competition/cname/clocation_table

```
var urlencoded = new URLSearchParams();
urlencoded.append("location", "Maddison Square Garden, Toronto");

var requestOptions = {
  method: 'DELETE',
  body: urlencoded,
  redirect: 'follow'
};

fetch("http://localhost:8000/Competition/LCS Sprint PlayOffs/clocation_table", requestOptions)
.then(response => response.text())
.then(result => console.log(result))
.catch(error => console.log("error", error));
```

Example Response 200 OK

```
Body Headers ()

{
  "message": "deleted",
  "data": {}
}
```