



6/14/2021

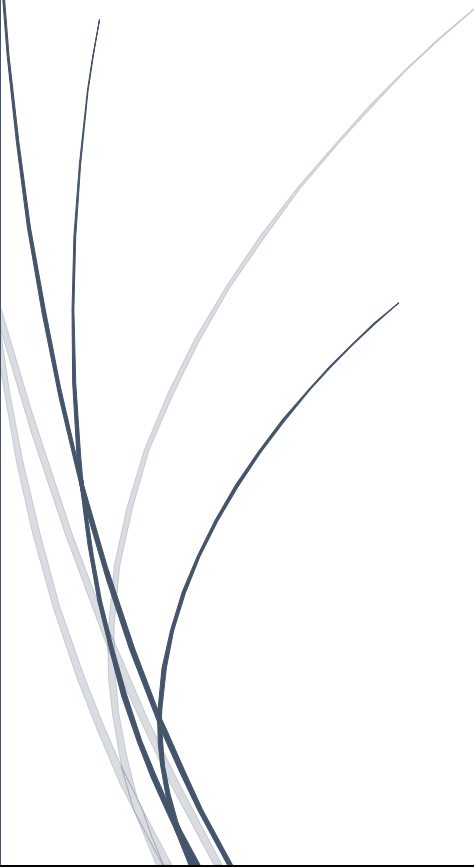
# OOP Semester Project

Spring 2021

## Submitted to:

- Dr Imran Ahmed  
Siddique
- Sir Sikandar Hayat

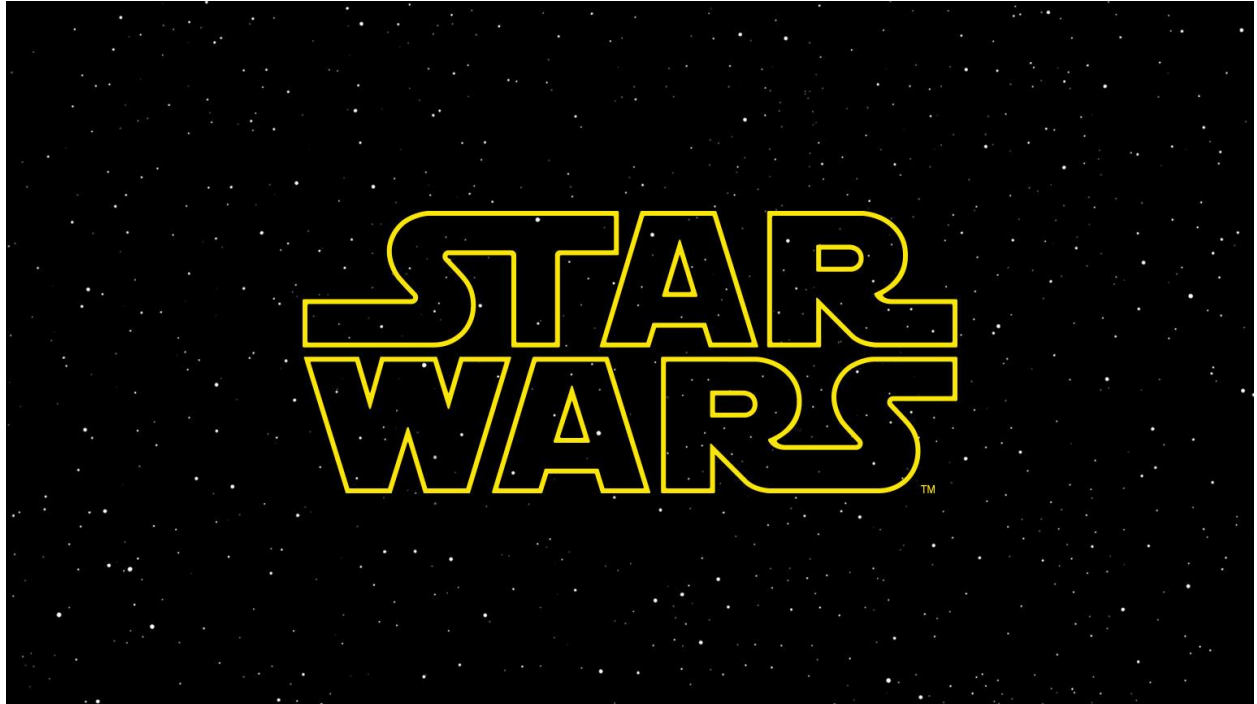
## Submitted by:

- Mubeen Ahmed Jawad
  - 01-134182-081
  - BSCS 2-A
- 

## Table of Contents

<b>Overview:</b>	<b>2</b>
<b>Tools Used:</b>	<b>3</b>
<b>Class Diagram:</b>	<b>3</b>
<b>Code:</b>	<b>4</b>
Source.cpp	4
Planets.h	24
Cities.h	25
Jedi.h	25
Sith.h	26
Forces.h	26
Battles.h	26
Planets.cpp	27
Cities.cpp	28
Jedi.cpp	29
Sith.cpp	30
Forces.cpp	31
Battles.cpp	32
<b>Outputs:</b>	<b>33</b>
Intro Screen:	33
Main Menu:	34
Planets Menu:	34
Adding Data to Planet using file handling:	35
Data Saved in Text File:	35
Printing Data in console using file handling:	36
<b>Error Fixing References:</b>	<b>36</b>

## Overview:



### Multi Planetary Galactic Empire Management System

based on Star Wars Franchise. This system manages and holds records of the progress made by the empire after Palpatine has managed to crown himself king. He is a sith lord(evil) in disguise who wants to Eliminate the Jedi Forces(good) and conquer the galaxy.

#### Planets:

- Status: taken over or not
- no of cities
- Population of planet

#### Jedi Details:

- Registered Jedi
- Registered Sith Lords

#### Main Battleships/Starships of the Galactic Empire

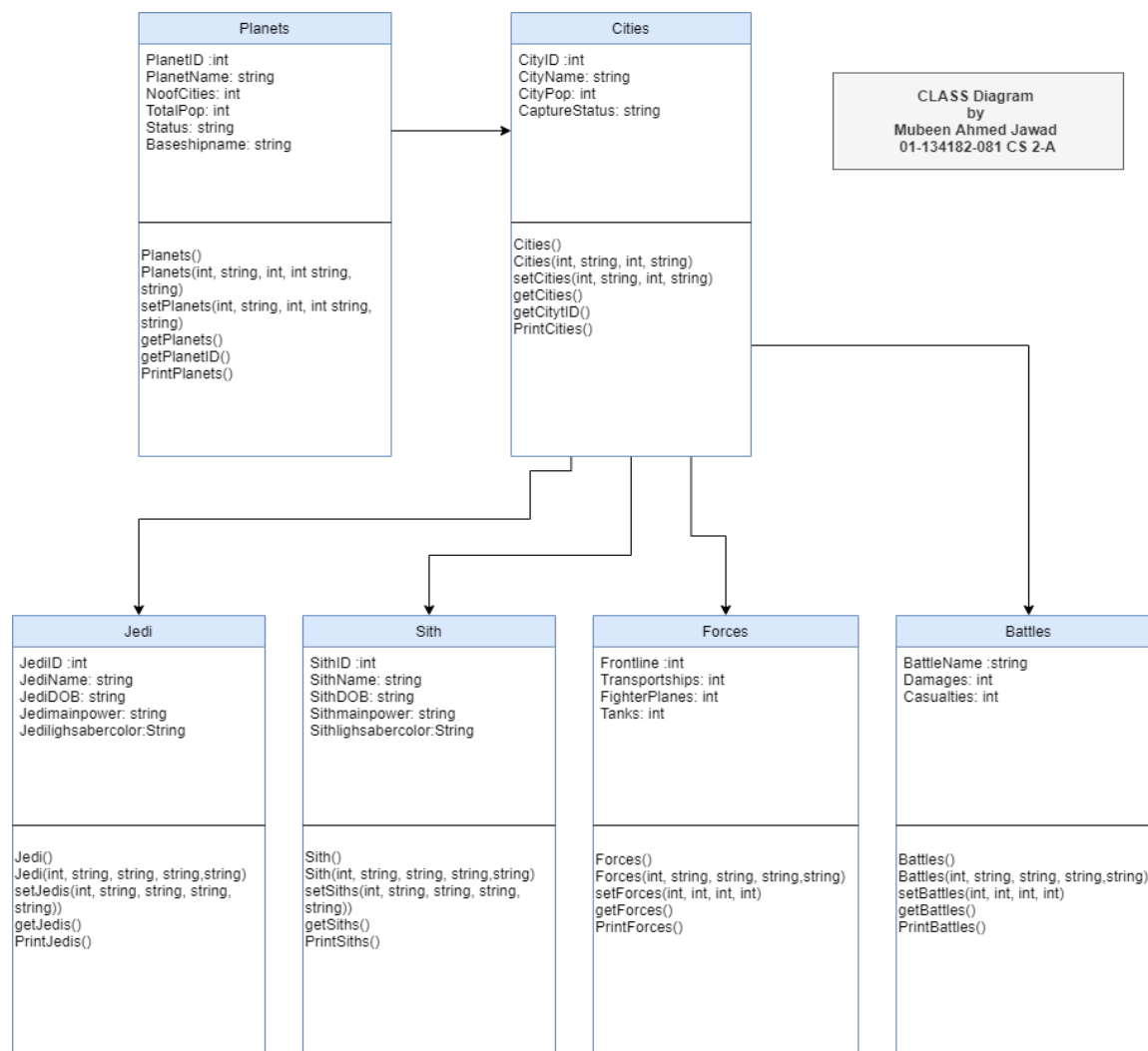
- Fighters Records
- Frontline Forces assigned to the planet
- Transport Ships assigned to the planet
- Fighter Planes assigned to the planet
- On ground tanks assigned to the planet

Records of Battles that have been lost or won

- Includes data of damages, casualties, location, planet etc

**Tools Used:** Microsoft Visual Studio, MS Word, Snipping Tool, Google Chrome, **Draw.IO**

## Class Diagram:



## Code:

### Source.cpp

```
#include<iostream>

#include<conio.h>

#include<string>

#include<fstream>

#include"Planets.h"

#include"Cities.h"

#include"Jedi.h"

#include"Sith.h"

#include"Forces.h"

#include"Battles.h"

using namespace std;


void mainmenu();

void PlanetMenu();

void CityMenu();

void JediMenu();

void SithMenu();

void ForcesMenu();

void BattlesMenu();

void AddPlanet();

void AddCity();

void AddJedis();

void AddSiths();

void AddForcess();

void AddBattles();


int main()
```

```
{

    char p;

    cout << "*****" << endl;

    cout << "Welcome to the MultiPlanetary Galactic Empire Management System" << endl;

    cout << endl;

    cout << "        Object Oriented Programming Project        " << endl;

    cout << endl;

    cout << "        Submitted to: 1. Dr Imran Ahmed Siddique        " << endl;

    cout << "                2. Sir Sikandar Hayat                " << endl;

    cout << endl;

    cout << "        Submitted by: Mubeen Ahmed Jawad        " << endl;

    cout << "                01-134182-081                " << endl;

    cout << "                BSCS 2-A                " << endl;

    cout << "*****" << endl;

    cout << endl;

    cout << "*****" << endl;

    cout << "        Press 'G' to continue        " << endl;

    cout << "*****" << endl;

    cin >> p;

    if (p == 'g')
    {

        system("cls");

        mainmenu();

    }

    _getch();

    return 0;

}
```

```
void mainmenu()
{
    char c;

    cout << "*****" << endl;
    cout << endl;
    cout << "Please Select an Option" << endl;
    cout << endl;
    cout << "    1. Planets                " << endl;
    cout << "    2. Cities                  " << endl;
    cout << "    3. Jedi                   " << endl;
    cout << "    4. Sith                   " << endl;
    cout << "    5. Forces                 " << endl;
    cout << "    6. Battles                " << endl;
    cout << "    7. Go Back                " << endl;
    cout << endl;
    cout << "*****" << endl;
    cin >> c;
    switch (c)
    {
        case '1':
            system("cls");
            PlanetMenu();
            break;
        case '2':
            system("cls");
            CityMenu();
            break;
        case '3':
            system("cls");
```

```
        JediMenu();
        break;
    case '4':
        system("cls");
        SithMenu();
        break;
    case '5':
        system("cls");
        ForcesMenu();
        break;
    case '6':
        system("cls");
        BattlesMenu();
        break;
    case '7':
        system("cls");
        main();
        break;
    }

}

void PlanetMenu()
{
    char c;

    cout << "*****" << endl;
    cout << endl;
    cout << "          Planets          " << endl;
    cout << endl;
```



```
        cout << "Please Select an Option" << endl;

        cout << endl;

        cout << "        1. Add Data                " << endl;
        cout << "        2. Print Data                " << endl;
        cout << "        3. Go Back                " << endl;

        cout << endl;

        cout << "*****" <<
endl;

        cin >> c;

        switch (c)
        {
        case '1':
        {
                AddPlanet();

                break;

        }
        case '2':
        {
                Planets obj1;

                obj1.PrintPlanets();

                break;

        }
        case '3':
                system("cls");

                mainmenu();

        }
}

void CityMenu()
```

```
{  
  
    char c;  
  
    cout << "*****" << endl;  
  
    cout << endl;  
  
    cout << "          Cities          " << endl;  
  
    cout << endl;  
  
    cout << "Please Select an Option" << endl;  
  
    cout << endl;  
  
    cout << "    1. Add Data                " << endl;  
    cout << "    2. Print Data              " << endl;  
    cout << "    3. Go Back                " << endl;  
  
    cout << endl;  
  
    cout << "*****" <<  
endl;  
  
    cin >> c;  
  
    switch (c)  
    {  
    case '1':  
        AddCity();  
        break;  
  
    case '2':  
    {  
        Cities obj2;  
        obj2.PrintCities();  
        break;  
    }  
  
    case '3':  
        system("cls");  
        mainmenu();  
    }
```

```
        break;
    }
}

void JediMenu()
{
    char c;

    cout << "*****" << endl;
    cout << endl;
    cout << "        Jedi        " << endl;
    cout << endl;
    cout << "Please Select an Option" << endl;
    cout << endl;
    cout << "    1. Add Data        " << endl;
    cout << "    2. Print Data      " << endl;
    cout << "    3. Go Back         " << endl;
    cout << endl;
    cout << "*****" <<
endl;

    cin >> c;
    switch (c)
    {
    case '1':
        AddJedis();
        break;
    case '2':
    {
        Jedi obj3;
        obj3.PrintJedis();
    }
    }
}
```

```
        break;
    }
    case '3':
        system("cls");
        mainmenu();
    }
}

void SithMenu()
{
    char c;
    cout << "*****" << endl;
    cout << endl;
    cout << "          Siths          " << endl;
    cout << endl;
    cout << "Please Select an Option" << endl;
    cout << endl;
    cout << "    1. Add Data          " << endl;
    cout << "    2. Print Data       " << endl;
    cout << "    3. Go Back          " << endl;
    cout << endl;
    cout << "*****" <<
endl;
    cin >> c;
    switch (c)
    {
    case '1':
        AddSiths();
        break;
```

```
        case '2':
        {
            Sith obj4;
            obj4.PrintSiths();
            break;
        }
        case '3':
            system("cls");
            mainmenu();
        }
    }

void ForcesMenu()
{
    char c;
    cout << "*****" << endl;
    cout << endl;
    cout << "          Forces          " << endl;
    cout << endl;
    cout << "Please Select an Option" << endl;
    cout << endl;
    cout << "    1. Add Data          " << endl;
    cout << "    2. Print Data        " << endl;
    cout << "    3. Go Back           " << endl;
    cout << endl;
    cout << "*****" << endl;
    cin >> c;
```

```
        switch (c)
        {
        case '1':
            AddForcess();
            break;
        case '2':
        {
            Forces obj5;
            obj5.PrintForces();
            break;
        }
        case '3':
            system("cls");
            mainmenu();
        }
    }

void BattlesMenu()
{
    char c;
    cout << "*****" << endl;
    cout << endl;
    cout << "            Battles            " << endl;
    cout << endl;
    cout << "Please Select an Option" << endl;
    cout << endl;
    cout << "    1. Add Data            " << endl;
    cout << "    2. Print Data         " << endl;
    cout << "    3. Go Back           " << endl;
```

```
        cout << endl;

        cout << "*****" <<
endl;

        cin >> c;

        switch (c)
        {
        case '1':
                AddBattles();

                break;

        case '2':
        {
                Battles obj6;

                obj6.PrintBattles();

                break;

        }

        case '3':
                system("cls");

                mainmenu();

        }

}

void AddPlanet()
{

        char c;

        string Pname, cstatus, ship;

        int Pid, cities, plpopulation;

        cout << "*****" << endl;

        cout << endl;

        cout << "                Planets                " << endl;
```

```
cout << endl;

cout << "Please Enter the following:          " << endl;

cout << endl;

cout << "    Planet ID: "; cin >> Pid;

cout << endl;

cout << "    Planet Name: "; cin >> Pname;

cout << endl;

cout << "    No of Cities: "; cin >> cities;

cout << endl;

cout << "    Planet Population: "; cin >> plpopulation;

cout << endl;

cout << "    Planet Status: "; cin >> cstatus;

cout << endl;

cout << "    Planet BaseshipName: "; cin >> ship;

cout << endl;


ofstream myfile("Planets.txt");

if (myfile.is_open())
{
    myfile << "    Planet ID: " << Pid;
    myfile << endl;
    myfile << "    Planet Name: " << Pname;
    myfile << endl;
    myfile << "    No of Cities: " << cities;
    myfile << endl;
    myfile << "    Planet Population: " << plpopulation;
    myfile << endl;
    myfile << "    Planet Status: " << cstatus;
    myfile << endl;
}
```



```
        myfile << "        Planet BaseshipName: " << ship;
        myfile << endl;
        myfile.close();
    }
    else cout << "Unable to open file";
    Planets obj1(Pid, Pname, cities, plpopulation, cstatus, ship);

    cout << endl;
    cout << "Press B to go back to main menu                " << endl;
    cout << endl;
    cin >> c;
    switch (c)
    {
    case 'b':
        system("cls");
        mainmenu();
    }
}

void AddCity()
{
    char c;
    string Cname, citystatus;
    int cityid, cpopulation;
    cout << "*****" << endl;
    cout << endl;
    cout << "                Cities                " << endl;
    cout << endl;
    cout << "Please Enter the following:                " << endl;
```

```
cout << endl;

cout << "    City ID: "; cin >> cityid;

cout << endl;

cout << "    City Name: "; cin >> Cname;

cout << endl;

cout << "    City Population: "; cin >> cpopulation;

cout << endl;

cout << "    City Status: "; cin >> citystatus;

ofstream myfile("Cities.txt");

if (myfile.is_open())
{
    myfile << "    City ID: " << cityid;
    myfile << endl;
    myfile << "    City Name: " << Cname;
    myfile << endl;
    myfile << "    City Population: " << cpopulation;
    myfile << endl;
    myfile << "    City Status: " << citystatus;
    myfile.close();
}

else cout << "Unable to open file";

Cities obj2(cityid, Cname, cpopulation, citystatus);

cout << endl;

cout << "Press B to go back to main menu" << endl;

cout << endl;

cin >> c;

switch (c)
```

```
{
    case 'b':
        system("cls");
        mainmenu();
    }
}

void AddJedis()
{
    char c;
    string Jname, jdob, jpower, jcolor;
    int jedid;

    cout << "*****" << endl;
    cout << endl;
    cout << "          Jedis          " << endl;
    cout << endl;
    cout << "Please Enter the following:          " << endl;
    cout << endl;
    cout << "    Jedi ID: "; cin >> jedid;
    cout << endl;
    cout << "    Jedi Name: "; cin >> Jname;
    cout << endl;
    cout << "    Jedi Date of Birth: "; cin >> jdob;
    cout << endl;
    cout << "    Jedi Power: "; cin >> jpower;
    cout << endl;
    cout << "    Jedi Light Saber Color: "; cin >> jcolor;

    ofstream myfile("Jedi.txt");
```

```
if (myfile.is_open())
{
    myfile << "    Jedi ID: " << jedid;
    myfile << endl;
    myfile << "    Jedi Name: " << Jname;
    myfile << endl;
    myfile << "    Jedi Date of Birth: " << jdob;
    myfile << endl;
    myfile << "    Jedi Power: " << jpower;
    myfile << endl;
    myfile << "    Jedi Light Saber Color: " << jcolor;
    myfile.close();
}
else cout << "Unable to open file";

Jedi obj3(jedid, Jname, jdob, jpower, jcolor);

cout << endl;
cout << "Press B to go back to main menu" << endl;
cout << endl;
cin >> c;
switch (c)
{
case 'b':
    system("cls");
    mainmenu();
}
}
```

```
void AddSiths()
```

```
{  
  
    char c;  
  
    string Sname, Sdob, Spower, Scolor;  
  
    int Sid;  
  
    cout << "*****" << endl;  
    cout << endl;  
    cout << "          Siths          " << endl;  
    cout << endl;  
    cout << "Please Enter the following:          " << endl;  
    cout << endl;  
    cout << "    Sith ID: "; cin >> Sid;  
    cout << endl;  
    cout << "    Sith Name: "; cin >> Sname;  
    cout << endl;  
    cout << "    Sith Date of Birth: "; cin >> Sdob;  
    cout << endl;  
    cout << "    Sith Power: "; cin >> Spower;  
    cout << endl;  
    cout << "    Sith Light Saber Color: "; cin >> Scolor;  
  
    ofstream myfile("Sith.txt");  
    if (myfile.is_open())  
    {  
        myfile << "    Sith ID: " << Sid;  
        myfile << endl;  
        myfile << "    Sith Name: " << Sname;  
        myfile << endl;  
        myfile << "    Sith Date of Birth: " << Sdob;  
        myfile << endl;  
    }
```

```
        myfile << "        Sith Power: " << Spower;
        myfile << endl;
        myfile << "        Sith Light Saber Color: " << Scolor;
        myfile.close();
    }
    else cout << "Unable to open file";
    Sith obj4(Sid, Sname, Sdob, Spower, Scolor);

    cout << endl;
    cout << "Press B to go back to main menu                " << endl;
    cout << endl;
    cin >> c;
    switch (c)
    {
    case 'b':
        system("cls");
        mainmenu();
    }
}

void AddForcess()
{
    char c;
    int front, transpo, fplanes, tanks;
    cout << "*****" << endl;
    cout << endl;
    cout << "                Forces                " << endl;
    cout << endl;
    cout << "Please Enter the following:                " << endl;
```

```
cout << endl;

cout << "    No of FrontLine Soldiers: "; cin >> front;

cout << endl;

cout << "    No of Transport Ships: "; cin >> transpo;

cout << endl;

cout << "    No of Fighter Planes: "; cin >> fplanes;

cout << endl;

cout << "    No of Tanks: "; cin >> tanks;


ofstream myfile("Forces.txt");

if (myfile.is_open())
{
    myfile << "    No of FrontLine Soldiers: " << front;
    myfile << endl;
    myfile << "    No of Transport Ships: " << transpo;
    myfile << endl;
    myfile << "    No of Fighter Planes: " << fplanes;
    myfile << endl;
    myfile << "    No of Tanks: " << tanks;
    myfile.close();
}

else cout << "Unable to open file";

Forces obj5(front, transpo, fplanes, tanks);

cout << endl;

cout << endl;

cout << "Press B to go back to main menu" << endl;

cout << endl;

cin >> c;

switch (c)
```

```
{
    case 'b':
        system("cls");
        mainmenu();
    }
}

void AddBattles()
{
    char c;
    string bname;
    int dmgs, casu;

    cout << "*****" << endl;
    cout << endl;
    cout << "          Battles          " << endl;
    cout << endl;
    cout << "Please Enter the following:          " << endl;
    cout << endl;
    cout << "    Battle Name ID: "; cin >> bname;
    cout << endl;
    cout << "    Damages in Dollars: "; cin >> dmgs;
    cout << endl;
    cout << "    No of Casualties: "; cin >> casu;
    cout << endl;

    ofstream myfile("Battles.txt");
    if (myfile.is_open())
    {
        myfile << "    Battle Name ID: "; cin >> bname;
```



```
        myfile << endl;
        myfile << "    Damages in Dollars: "; cin >> dmgs;
        myfile << endl;
        myfile << "    No of Casualties: "; cin >> casu;
        myfile.close();
    }
    else cout << "Unable to open file";
    Battles obj6(bname, dmgs, casu);

    cout << endl;
    cout << "Press B to go back to main menu" << endl;
    cout << endl;
    cin >> c;
    switch (c)
    {
        case 'b':
            system("cls");
            mainmenu();
            break;
    }
}
```

## Planets.h

```
#pragma once
#include<iostream>
#include <string>
using namespace std;
class Planets
{
protected:
    int PlanetID;
    string PlanetName;
    int NoofCities;
    int TotalPop;
    string Status;
    string BaseshipName;
```

```
public:
    Planets();
    Planets(int, string, int, int, string, string);
    void setPlonets(string, int, int, string, string);
    void setPlanetID(int pi);
    int getPlanetData();
    int getPlanetID();
    void PrintPlanets();

};
```

## Cities.h

```
#pragma once
#include "Planets.h"
#include <string>
using namespace std;
class Cities:public Planets
{
protected:
    int CityID;
    string CityName;
    int CityPop;
    string CaptureStatus;

public:
    Cities();
    Cities(int,string,int,string);
    void setCities(string, int, string);
    void setCityID(int);
    int getCityData();
    int getCityID();
    void PrintCities();

};
```

## Jedi.h

```
#pragma once
#include "Cities.h"
#include<iostream>
using namespace std;
class Jedi: public Cities
{
protected:
    int JediID;
    string JediName;
    string JDOB;
    string Jedimainpower;
    string Jedilightsabercolor;

public:
    Jedi();
    Jedi(int, string, string, string, string);
    void setJedis(int, string, string, string, string);
    int getJedis();
    void PrintJedis();

};
```

## Sith.h

```
#pragma once
#include "Cities.h"
using namespace std;
class Sith : public Cities
{
protected:
    int SithID;
    string SithName;
    string SDOB;
    string Sithmainpower;
    string Sithlightsabercolor;
public:
    Sith();
    Sith(int, string, string, string, string);
    void setSith(int, string, string, string, string);
    int getSith();
    void PrintSiths();
};
```

## Forces.h

```
#pragma once
#include "Cities.h"
using namespace std;
class Forces: public Cities
{
protected:
    int Frontline;
    int TransportShips;
    int FighterPlanes;
    int Tanks;
public:
    Forces();
    Forces(int, int, int, int);
    void setForces(int, int, int , int);
    int getForces();
    void PrintForces();
};
```

## Battles.h

```
#pragma once
#include "Cities.h"
using namespace std;
class Battles : public Planets, public Cities
{
protected:
    string BattleName;
    int Damages;
    int Casualties;
public:
    Battles();
    Battles(string, int, int);
    void setBattles(string, int , int);
};
```

```
        int getBattles();  
        void PrintBattles();  
};
```

## Planets.cpp

```
#include "Planets.h"  
#include <string>  
#include <iostream>  
#include <fstream>  
using namespace std;  
  
Planets::Planets()  
{  
    PlanetID = 0;  
    PlanetName = " ";  
    NoofCities = 0;  
    TotalPop = 0;  
    Status = " ";  
    BaseshipName = " ";  
}  
  
Planets::Planets(int pi, string pn, int nc, int tp, string ps, string bsn)  
{  
    this->PlanetID = pi;  
    this->PlanetName = pn;  
    this->NoofCities = nc;  
    this->TotalPop = tp;  
    this->Status = ps;  
    this->BaseshipName = bsn;  
}  
  
void Planets::setPlonets(string pn, int nc, int tp, string ps, string bsn)  
{  
    PlanetName = pn;  
    NoofCities = nc;  
    TotalPop = tp;  
    Status = ps;  
    BaseshipName = bsn;  
}  
  
void Planets::setPlanetID(int pi)  
{  
    PlanetID = pi;  
}  
  
int Planets::getPlanetData()  
{  
    return PlanetID;  
    cout << endl;  
    return NoofCities;  
    cout << endl;  
    return TotalPop;  
    cout << endl;  
}  
  
int Planets::getPlanetID()  
{  
    return PlanetID;  
}
```

```
}

void Planets::PrintPlanets()
{
    string line;
    ifstream myfile("Planets.txt");
    if (myfile.is_open())
    {
        while (!myfile.eof())
        {
            getline(myfile, line);
            cout << line << endl;
        }
        myfile.close();
    }
    else cout << "Unable to open file";
}
```

### Cities.cpp

```
#include "Cities.h"
#include <string>
#include <fstream>
#include "Planets.h"
#include <iostream>
using namespace std;

Cities::Cities()
{
    CityID = 0;
    CityName = "";
    CityPop = 0;
    CaptureStatus = "";
}

Cities::Cities(int ci, string cn, int cp, string cs)
{
    this->CityID = ci;
    this->CityName = cn;
    this->CityPop = cp;
    this->CaptureStatus = cs;
}

void Cities::setCities(string cn, int cp, string cs)
{
    CityName = cn;
    CityPop = cp;
    CaptureStatus = cs;
}

void Cities::setCityID(int ci)
{
    CityID = ci;
}

int Cities::getCityData()
{
    return CityPop;
}
```

```
}

int Cities::getCityID()
{
    return CityID;
}

void Cities::PrintCities()
{
    string line;
    ifstream myfile("Cities.txt");
    if (myfile.is_open())
    {
        while (!myfile.eof())
        {
            getline(myfile, line);
            cout << line << endl;
        }
        myfile.close();
    }
    else cout << "Unable to open file";
}
```

## Jedi.cpp

```
#include "Jedi.h"
#include "Cities.h"
#include <string>
#include <fstream>
#include <iostream>
using namespace std;

Jedi::Jedi()
{
    JediID = 0;
    JediName = "";
    JDOB = "";
    Jedimainpower = "";
    Jedilightsabercolor = "";
}

Jedi::Jedi(int ji, string jn, string db, string jp, string jl)
{
    this->JediID = ji;
    this->JediName = jn;
    this->JDOB = db;
    this->Jedimainpower = jp;
    this->Jedilightsabercolor = jl;
}

void Jedi::setJedis(int ji, string jn, string db, string jp, string jl)
{
    JediID = ji;
    JediName = jn;
    JDOB = db;
    Jedimainpower = jp;
    Jedilightsabercolor = jl;
}
```

```
}

int Jedi::getJedis()
{
    return JediID;
}

void Jedi::PrintJedis()
{
    string line;
    ifstream myfile("Jedi.txt");
    if (myfile.is_open())
    {
        while (!myfile.eof())
        {
            getline(myfile, line);
            cout << line << endl;
        }
        myfile.close();
    }
    else cout << "Unable to open file";
}
```

### Sith.cpp

```
#include "Sith.h"
#include <string>
#include <fstream>
#include <iostream>
using namespace std;

Sith::Sith()
{
    SithID = 0;
    SithName = "";
    SDOB = "";
    Sithmainpower = "";
    Sithlightsabercolor = "";
}

Sith::Sith(int si, string sn, string sb, string sp, string sl)
{
    this->SithID = si;
    this->SithName = sn;
    this->SDOB = sb;
    this->Sithmainpower = sp;
    this->Sithlightsabercolor = sl;
}

void Sith::setSith(int si, string sn, string sb, string jp, string sl)
{
    SithID = si;
    SithName = sn;
    SDOB = sb;
    Sithmainpower = jp;
    Sithlightsabercolor = sl;
}
```

```
int Sith::getSith()
{
    return SithID;
}

void Sith::PrintSiths()
{
    string line;
    ifstream myfile("Sith.txt");
    if (myfile.is_open())
    {
        while (!myfile.eof())
        {
            getline(myfile, line);
            cout << line << endl;
        }
        myfile.close();
    }
    else cout << "Unable to open file";
}
```

## Forces.cpp

```
#include "Forces.h"
#include "Cities.h"
#include <string>
#include <fstream>
#include <iostream>
using namespace std;

Forces::Forces()
{
    Frontline = 0;
    TransportShips = 0;
    FighterPlanes = 0;
    Tanks = 0;
}

Forces::Forces(int frl, int ts, int frp, int t)
{
    this->Frontline = frl;
    this->TransportShips = ts;
    this->FighterPlanes = frp;
    this->Tanks = t;
}

void Forces::setForces(int frl, int ts, int frp, int t)
{
    Frontline = frl;
    TransportShips = ts;
    FighterPlanes = frp;
    Tanks = t;
}

int Forces::getForces()
{
```



```
        return Frontline;
        return TransportShips;
        return FighterPlanes;
        return Tanks;
    }

    void Forces::PrintForces()
    {
        string line;
        ifstream myfile("Forces.txt");
        if (myfile.is_open())
        {
            while (!myfile.eof())
            {
                getline(myfile, line);
                cout << line << endl;
            }
            myfile.close();
        }
        else cout << "Unable to open file";
    }
```

## Battles.cpp

```
#include "Battles.h"
#include "Cities.h"
#include <string>
#include <fstream>
#include <iostream>
using namespace std;

Battles::Battles()
{
    BattleName = "";
    Damages = 0;
    Casualties = 0;
}

Battles::Battles(string bn, int bd, int bc)
{
    this->BattleName = bn;
    this->Damages = bd;
    this->Casualties = bc;
}

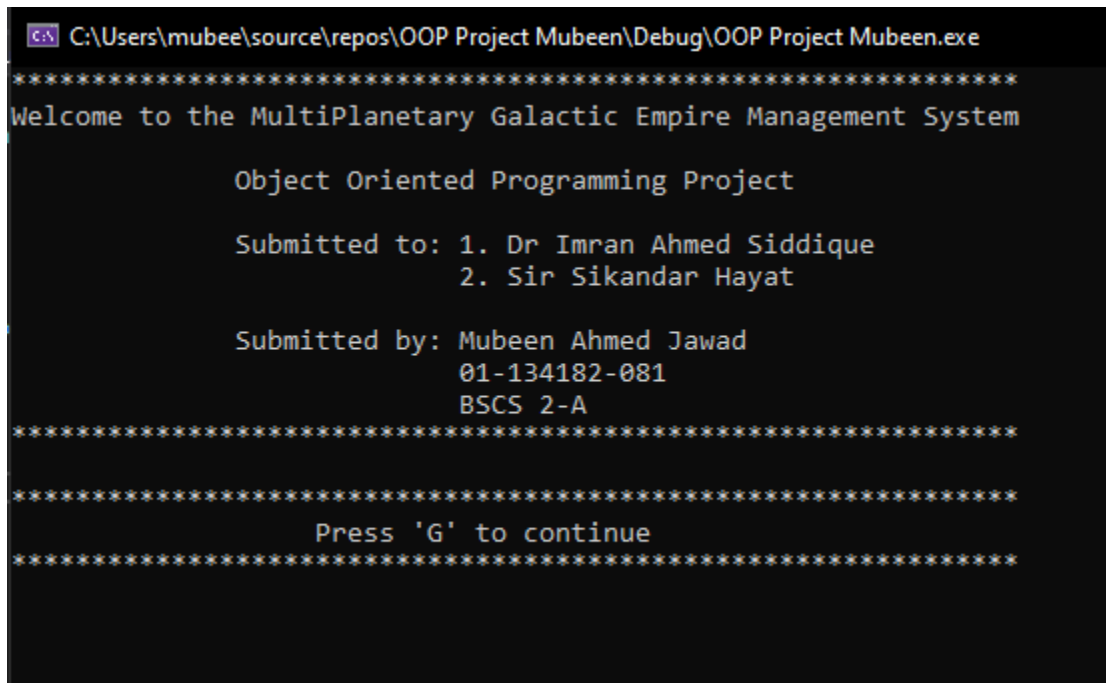
void Battles::setBattles(string bn, int bd, int bc)
{
    this->BattleName = bn;
    this->Damages = bd;
    this->Casualties = bc;
}

int Battles::getBattles()
{
    return Damages;
    return Casualties;
}
```

```
void Battles::PrintBattles()
{
    string line;
    ifstream myfile("Battles.txt");
    if (myfile.is_open())
    {
        while (!myfile.eof())
        {
            getline(myfile, line);
            cout << line << endl;
        }
        myfile.close();
    }
    else cout << "Unable to open file";
}
```

## Outputs:

### Intro Screen:



```
C:\Users\mubee\source\repos\OOP Project Mubeen\Debug\OOP Project Mubeen.exe
*****
Welcome to the MultiPlanetary Galactic Empire Management System

      Object Oriented Programming Project

Submitted to: 1. Dr Imran Ahmed Siddique
              2. Sir Sikandar Hayat

Submitted by: Mubeen Ahmed Jawad
              01-134182-081
              BSCS 2-A
*****
*****
      Press 'G' to continue
*****
```

### Main Menu:

```
C:\Users\mubee\source\repos\OOP Project Mubeen\Debug\OOP Project Mubeen.exe
*****

Please Select an Option

    1. Planets
    2. Cities
    3. Jedi
    4. Sith
    5. Forces
    6. Battles
    7. Go Back

*****
```

### Planets Menu:

```
C:\Users\mubee\source\repos\OOP Project Mubeen\Debug\OOP Project Mubeen.exe
*****

                        Planets

Please Select an Option

    1. Add Data
    2. Print Data
    3. Go Back

*****
```

### Adding Data to Planet using file handling:

```
C:\Users\mubee\source\repos\OOP Project Mubeen\Debug\OOP Project Mubeen.exe
Please Select an Option

    1. Add Data
    2. Print Data
    3. Go Back

*****
1
*****

                        Planets

Please Enter the following:

    Planet ID: 134
    Planet Name: Nabu
    No of Cities: 34
    Planet Population: 67000
    Planet Status: Captured
    Planet BaseshipName: Shamash

Press B to go back to main menu
```

### Data Saved in Text File:

```
Planets.txt - Notepad
File Edit Format View Help
Planet ID: 134
Planet Name: Nabu
No of Cities: 34
Planet Population: 67000
Planet Status: Captured
Planet BaseshipName: Shamash
```

Printing Data in console using file handling:

```
C:\Users\mubee\source\repos\OOP Project Mubeen\Debug\OOP Project Mubeen.exe
*****
                                Planets
Please Select an Option
    1. Add Data
    2. Print Data
    3. Go Back
*****
2
    Planet ID: 134
    Planet Name: Nabu
    No of Cities: 34
    Planet Population: 67000
    Planet Status: Captured
    Planet BaseshipName: Shamash
```

Error Fixing References:

- <http://www.cplusplus.com/forum/general/59789/>
- <https://stackoverflow.com/questions/25353829/c-error-member-enginex-is-not-a-type-name>
- <https://stackoverflow.com/questions/5136295/switch-transfer-of-control-bypasses-initialization-of-when-calling-a-function>