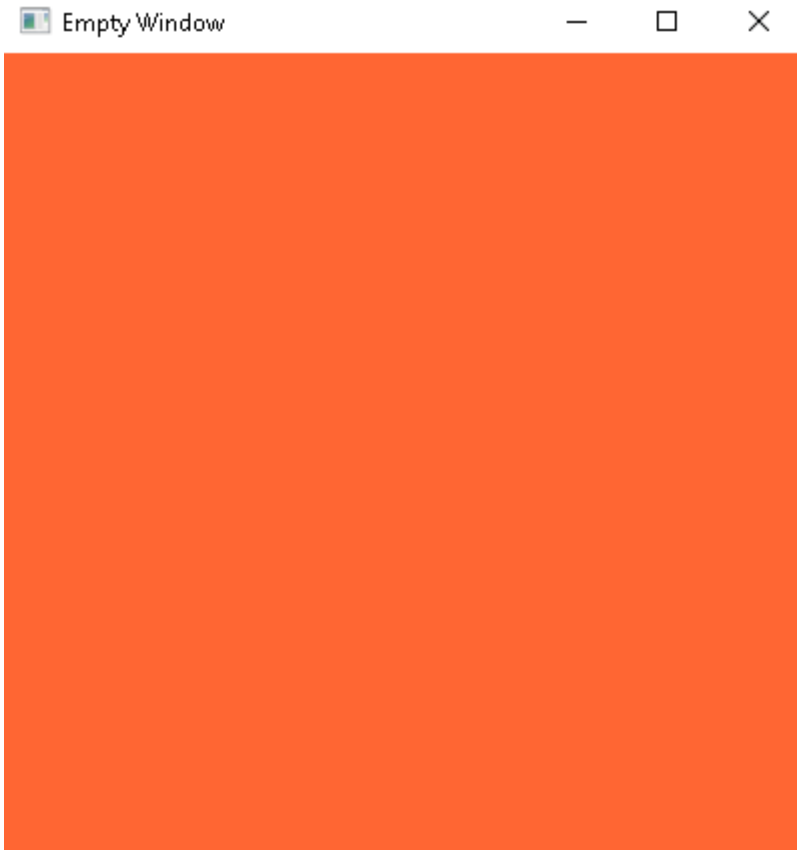
ASSIGNMENT -1
COMPUTER GRAPHICS

- Shubham Tiwari
101916126
3CS12

1. Create an Empty Window

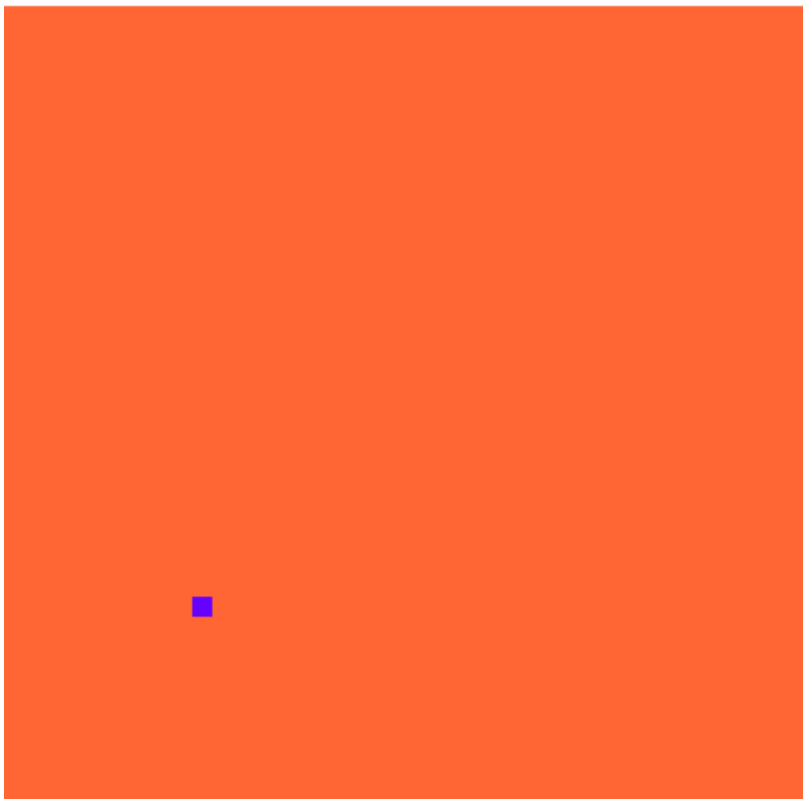
```
#include<GL\glut.h>
#include<iostream>
#include<windows.h>

using namespace std;

void myInit()
{
	glClearColor(0, 1.0, 0.2, 1.0);
	glColor3f(0.0, 1.0, 0.9);
	glMatrixMode(GL_PROJECTION);
	glLoadIdentity();
	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}


void myDisplay()
{
	glClear(GL_COLOR_BUFFER_BIT);
	glFlush();
}

int main(int argc, char** argv)
{

	glutInit(&argc, argv);
	glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
	glutInitWindowSize(400, 400);
	glutInitWindowPosition(0, 0);
	glutCreateWindow("Empty Window");
	glutDisplayFunc(myDisplay);
	myInit();
	glutMainLoop();
	return 0;
}
```


Empty Window


Empty Window

2. Draw a point of width 10 pixel

```
#include<GL\glut.h>
#include<iostream>
#include<windows.h>

using namespace std;

void myInit()
{
	glClearColor(1.0, 0.4, 0.2, 1.0);
	glColor3f(0.0, 1.0, 0.9);
	glMatrixMode(GL_PROJECTION);
	glLoadIdentity();
	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}


void myDisplay()
{
	glClear(GL_COLOR_BUFFER_BIT);
	glColor3f(0.4, 0.0, 1.0);
	glPointSize(10);
	glBegin(GL_POINTS);
	glVertex2i(100, 100);
	glEnd();
	glFlush();
}

int main(int argc, char** argv)
{

	glutInit(&argc, argv);
	glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
	glutInitWindowSize(400, 400);
	glutInitWindowPosition(0, 0);
	glutCreateWindow("Empty Window");
	glutDisplayFunc(myDisplay);
	myInit();
	glutMainLoop();
	return 0;
}
```



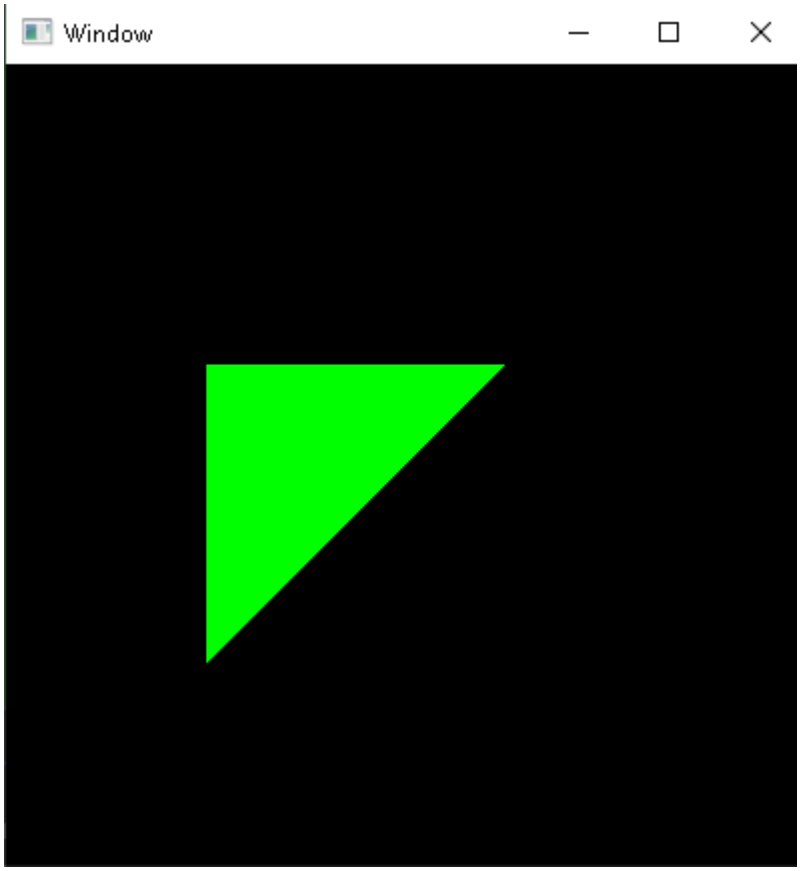
3. Draw a green color line from (10,10) to (50,50)

```
#include<GL\glut.h>
#include<iostream>
#include<windows.h>

using namespace std;

void myInit()
{
	glClearColor(0.0, 0.0, 0.0, 1.0);
	glColor3f(0.0, 1.0, 0.9);
	glMatrixMode(GL_PROJECTION);
	glLoadIdentity();
	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}


void myDisplay()
{
	glClear(GL_COLOR_BUFFER_BIT);
	glColor3f(0.0, 1.0, 0.0);
	//glPointSize(10);
	glLineWidth(5);
	glBegin(GL_LINE);
	glVertex3i(10, 10, 0);
	glVertex3i(50, 50, 0);
	glEnd();
	glFlush();
}

int main(int argc, char** argv)
{
```

```
	glutInit(&argc, argv);
	glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
	glutInitWindowSize(400, 400);
	glutInitWindowPosition(200, 200);
	glutCreateWindow("Window");
	glutDisplayFunc(myDisplay);
	myInit();
	glutMainLoop();
	return 0;
}
```

4. Draw a triangle on black background

```
#include<GL\glut.h>
#include<iostream>
#include<windows.h>

using namespace std;

void myInit()
{
	glClearColor(0.0, 0.0, 0.0, 1.0);
	glColor3f(0.0, 1.0, 0.9);
	glMatrixMode(GL_PROJECTION);
	glLoadIdentity();
	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}


void myDisplay()
{
	glClear(GL_COLOR_BUFFER_BIT);
	glColor3f(0.0, 1.0, 0.0);
	//glPointSize(10);
	glLineWidth(5);
	glBegin(GL_POLYGON);
	glVertex2i(100, 100);
	glVertex2i(250, 250);
	glVertex2i(100, 250);
	glEnd();
	glFlush();
}

int main(int argc, char** argv)
{

	glutInit(&argc, argv);
	glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
	glutInitWindowSize(400, 400);
	glutInitWindowPosition(200, 200);
	glutCreateWindow("Window");
	glutDisplayFunc(myDisplay);
	myInit();
	glutMainLoop();
	return 0;
}
```



5. Draw a rectangle on black background

```
include<GL\glut.h>
#include<iostream>
#include<windows.h>

using namespace std;

void myInit()
{
	glClearColor(0.0, 0.0, 0.0, 1.0);
	glColor3f(0.0, 1.0, 0.9);
	glMatrixMode(GL_PROJECTION);
	glLoadIdentity();
	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}


void myDisplay()
{
	glClear(GL_COLOR_BUFFER_BIT);
```

```
        glColor3f(0.0, 1.0, 0.0);
        glLineWidth(5);
        glBegin(GL_POLYGON);
        glVertex2i(100, 100);
        glVertex2i(100, 250);
        glVertex2i(250, 250);
        glVertex2i(250, 100);

        glEnd();
        glFlush();
}

int main(int argc, char** argv)
{

        glutInit(&argc, argv);
        glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
        glutInitWindowSize(400, 400);
        glutInitWindowPosition(200, 200);
        glutCreateWindow("Window");
        glutDisplayFunc(myDisplay);
        myInit();
        glutMainLoop();
        return 0;
}
```