



Sesi 2:
Primitive Drawing

Achmad Basuki

Ringkasan Materi

- Struktur Dasar Pemrograman OpenGL
 - Sistem Koordinat 2D
 - Menggambar Titik
 - Menggambar Garis
 - Menggambar Polyline
 - Menggambar Polygon
 - Mengganti Warna

Struktur Dasar Pemrograman OpenGL

```
#include <GLUT/glut.h>

void userdraw(void) {
    // Di sini tempat untuk menggambar
}

void display(void) {
    glClear( GL_COLOR_BUFFER_BIT);
    userdraw();
    glutSwapBuffers();
}

int main(int argc, char **argv){
    glutInit(&argc,argv);
    glutInitDisplayMode( GLUT_DOUBLE | GLUT_RGB );
    glutInitWindowPosition(100,100);
    glutInitWindowSize(640,480);
    glutCreateWindow("Drawing by Achmad Basuki");
    glClearColor(0.0, 0.0, 0.0, 0.0);
    gluOrtho2D(-320., 320., -240.0, 240.0);
    glutIdleFunc(display);
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```


Struktur Dasar Pemrograman OpenGL

- Buat project baru, beri nama **latihan1**
- Tuliskan/Copy program dengan struktur dasar pemrograman openGL
 - Build dan run
 - Hasilnya adalah sebuah jendela berwarna hitam. Jendela ini adalah tempat untuk menggambar.

Penjelasan

```
glutInitWindowPosition(100,100);  
glutInitWindowSize(640,480);
```

Membuat windows dengan ukuran(640,480) dan titik kiri atas jendela diletakkan pada posisi(100,100) di layar komputer.



Penjelasan

```
glClearColor(0.0, 0.0, 0.0, 0.0);
```

Mendefinisikan warna dari windows yang dibuat dengan warna(0,0,0) yaitu warna hitam. Angka terakhir biarkan 0.



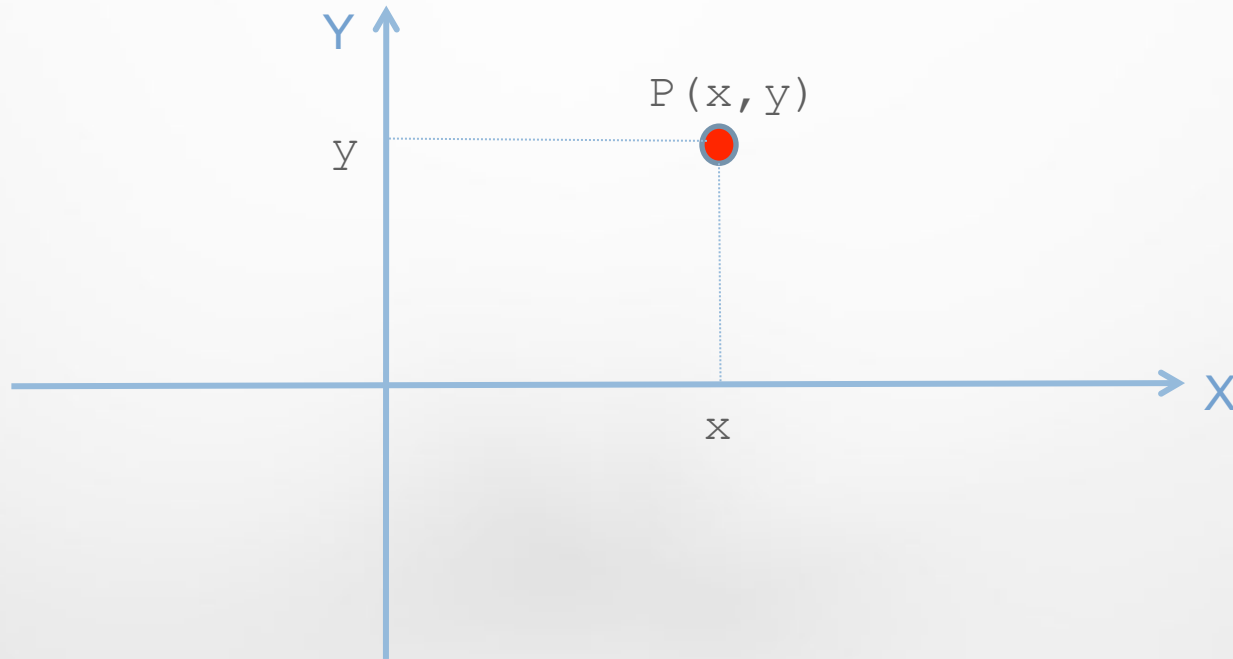
Penjelasan

```
gluOrtho2D(-320., 320., -240., 240.);
```

Mendefinisikan batas-batas sistem koordinat yang akan digunakan untuk menggambar dengan range sumbu X adalah $[-320, 320]$ dan range untuk sumbu Y adalah $[-240, 240]$.

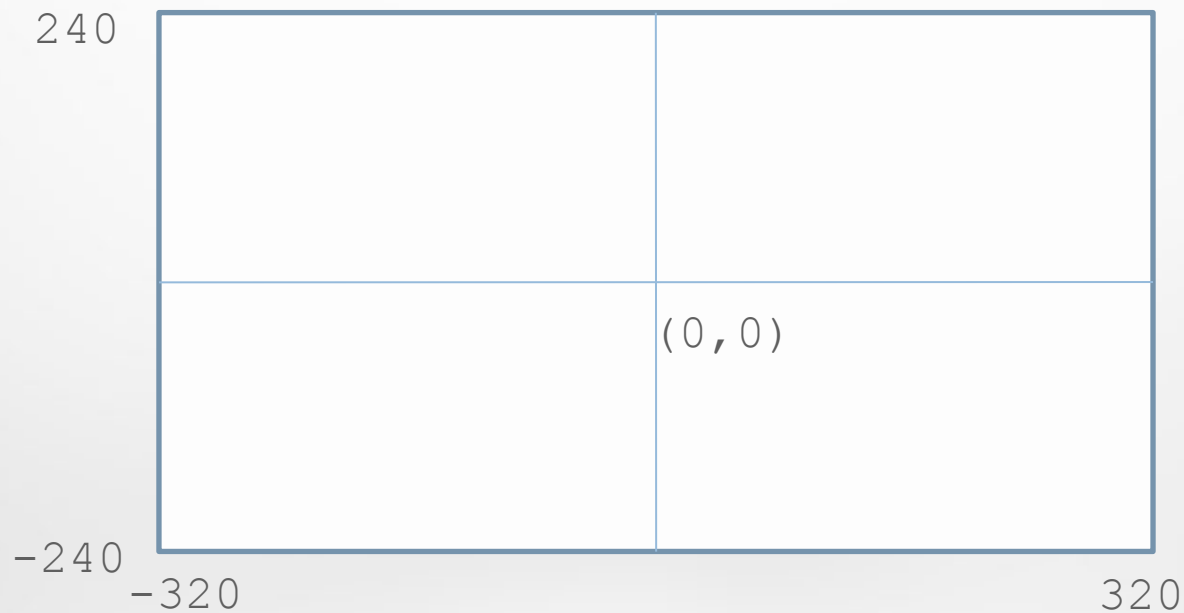


Sistem Koordinat 2D



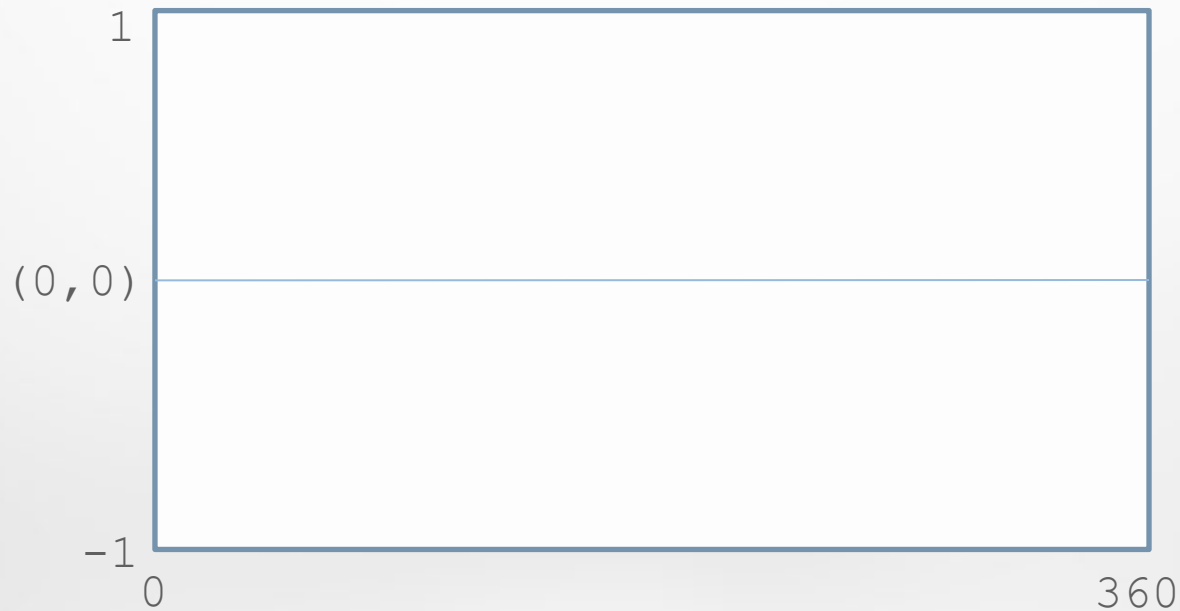
Penjelasan

```
gluOrtho2D(-320., 320., -240., 240.);
```



Penjelasan

```
gluOrtho2D(-0., 360., -1., 1.);
```



Menggambar Titik

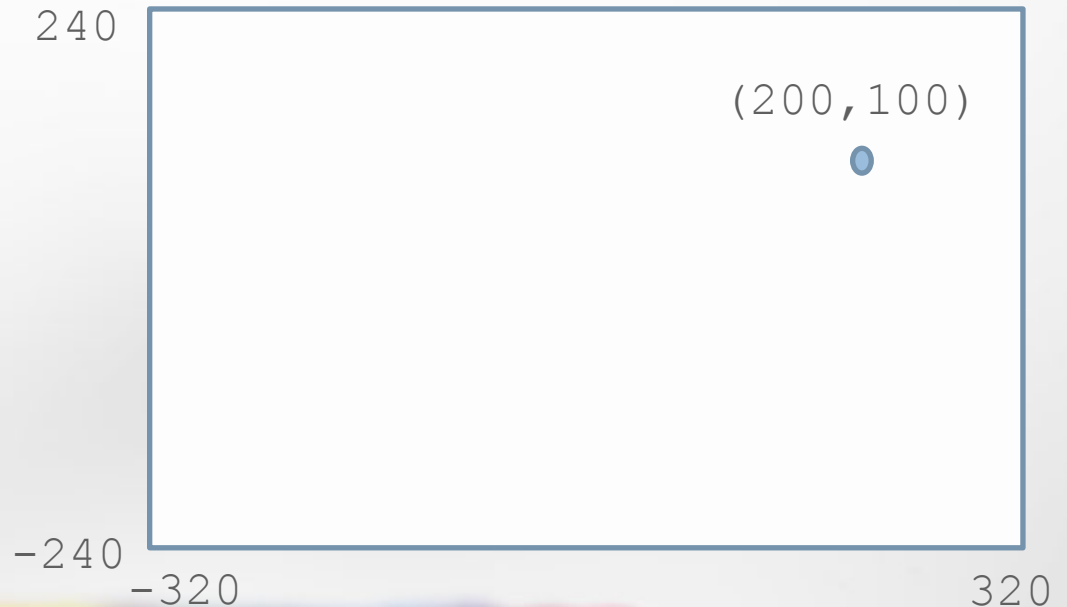
- Untuk menggambar titik dapat dilakukan dengan:

```
glBegin (GL_POINTS) ;  
glVertex2f (pos_x, pos_y) ;  
glEnd () ;
```

- Format pos_x dan pos_y adalah float.
- Untuk format integer menggunakan **glVertex2i(pos_x,pos_y);**
- Untuk format integer menggunakan **glVertex2d(pos_x,pos_y);**

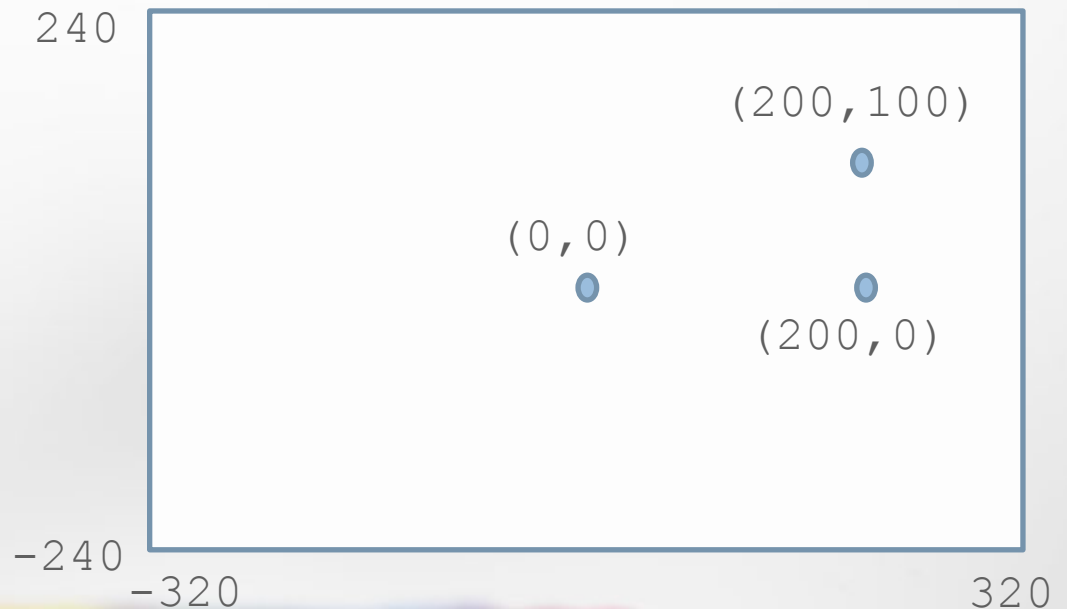
Contoh Menggambar Titik

```
glBegin (GL_POINTS) ;  
glVertex2f (200. , 100. ) ;  
glEnd () ;
```



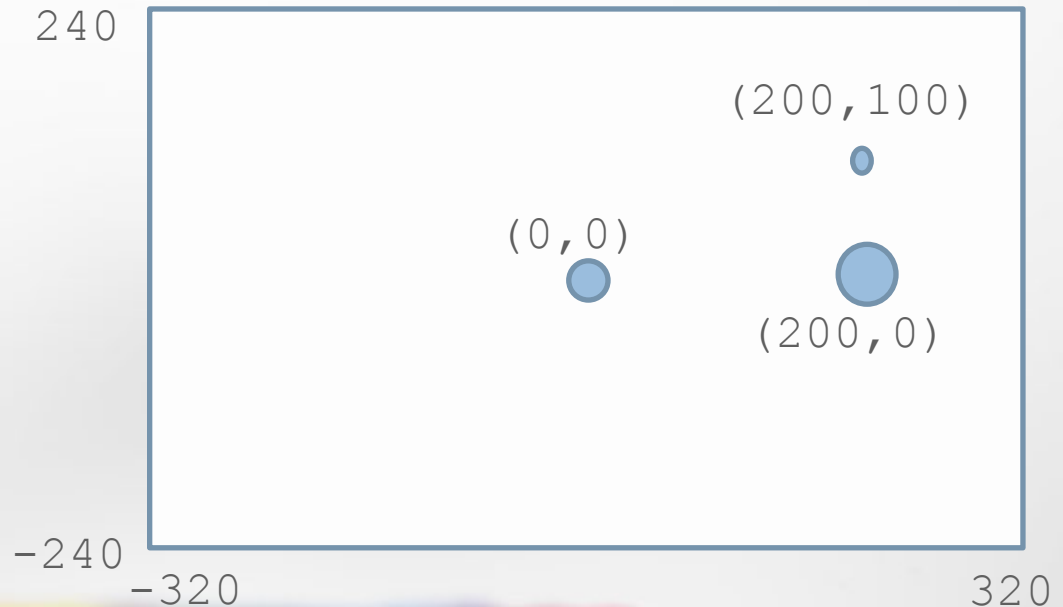
Contoh Menggambar Titik

```
glBegin(GL_POINTS);  
glVertex2f(200.,100.);  
glVertex2f(0.,0.);  
glVertex2f(200.,0.);  
glEnd();
```



Contoh Menggambar Titik

```
glBegin(GL_POINTS);  
glVertex2f(200.,100.);  
glPointSize(4);  
glVertex2f(0.,0.);  
glPointSize(8);  
glVertex2f(200.,0.);  
glEnd();
```



Menggambar Garis

- Untuk menggambar garis dapat dilakukan dengan:

```
glBegin (GL_LINES) ;
```

```
glVertex2f (x1 , y1) ;
```

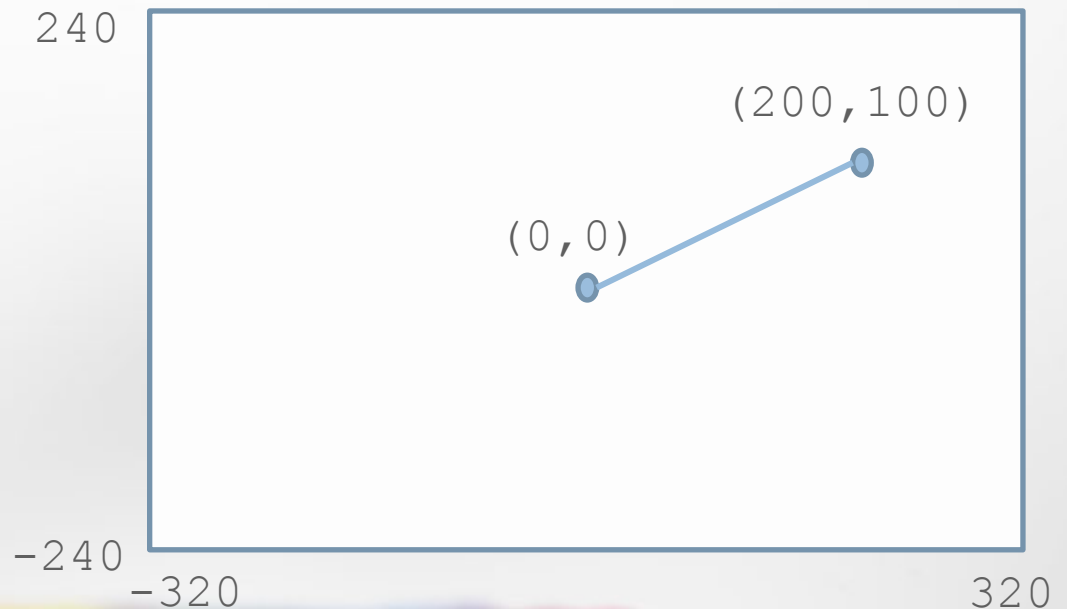
```
glVertex2f (x2 , y2) ;
```

```
glEnd () ;
```

- Garis dibentuk dari (x_1, y_1) ke (x_2, y_2)
- Format x_1 , y_1 , x_2 dan y_2 adalah float.

Contoh Menggambar Garis

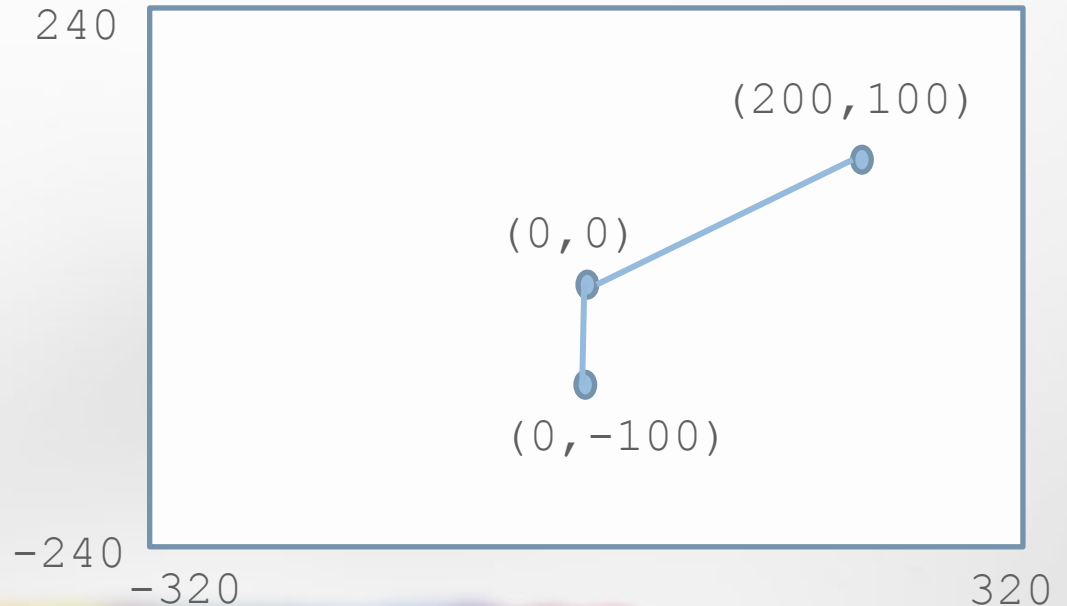
```
glBegin(GL_LINES);  
glVertex2f(200.,100.);  
glVertex2f(0.,0.);  
glEnd();
```



Contoh Menggambar Garis

```
glBegin(GL_LINES);  
glVertex2f(200.,100.);  
glVertex2f(0.,0.);
```

```
glVertex2f(0.,0.);  
glVertex2f(0.,-100.);  
glEnd();
```



Menggambar Polyline

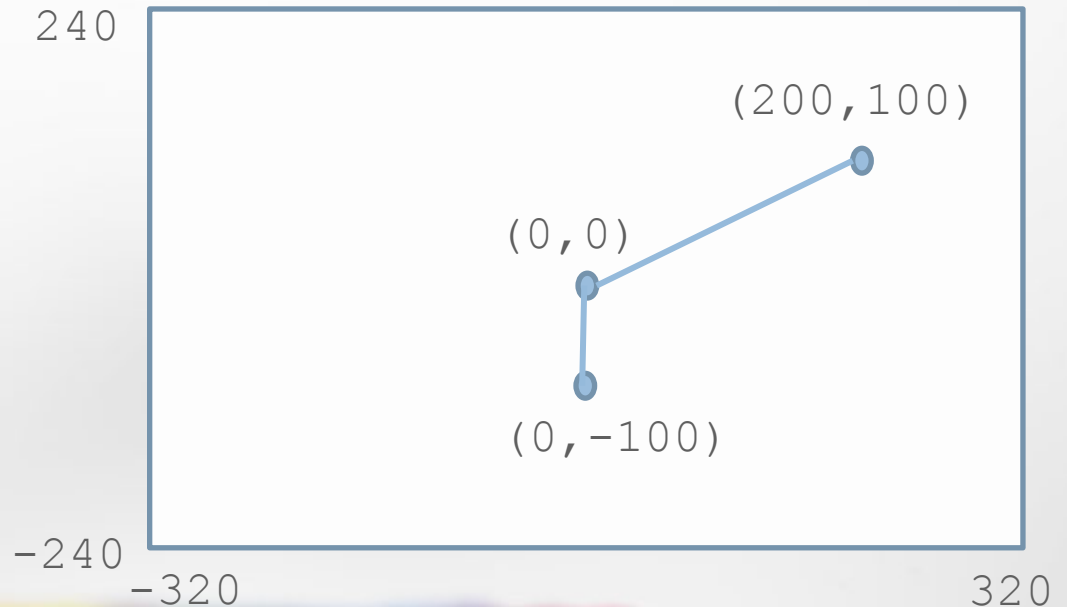
- Untuk menggambar polyline dapat dilakukan dengan:

```
glBegin (GL_LINE_STRIP) ;  
glVertex2f (x1 , y1) ;  
glVertex2f (x2 , y2) ;  
.....  
glVertex2f (xn , yn) ;  
glEnd () ;
```

- Polyline dibentuk dari (x_1, y_1) ke (x_n, y_n) secara berurutan.

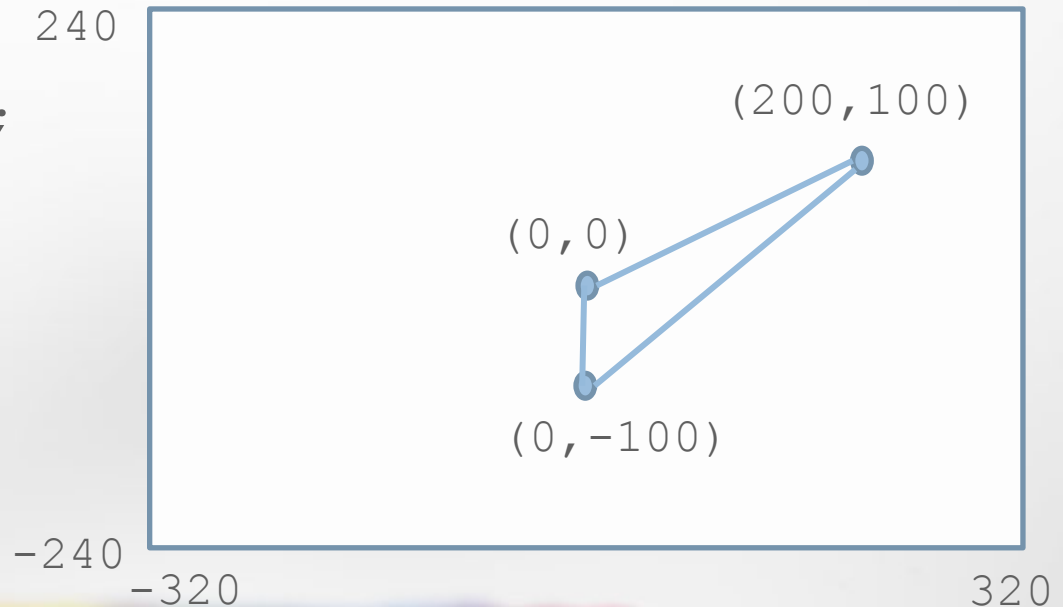
Contoh Menggambar Polyline

```
glBegin(GL_LINE_STRIP);  
glVertex2f(200.,100.);  
glVertex2f(0.,0.);  
glVertex2f(0.,-100.);  
glEnd();
```



Contoh Menggambar Polyline

```
glBegin(GL_LINE_STRIP);  
glVertex2f(200.,100.);  
glVertex2f(0.,0.);  
glVertex2f(0.,-100.);  
glVertex2f(200.,100.);  
glEnd();
```



Menggambar Polyline Tertutup

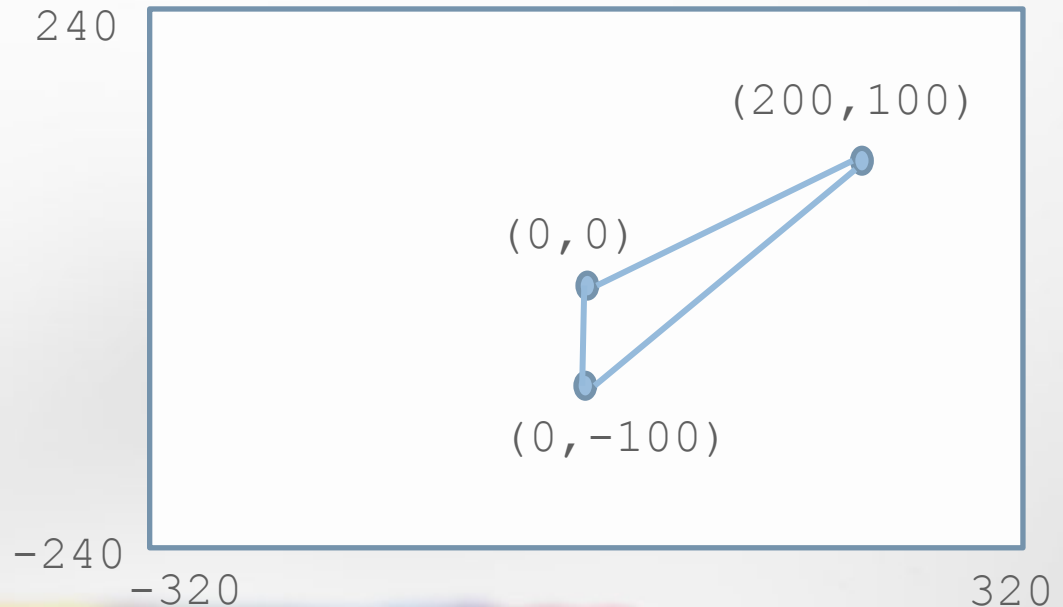
- Untuk menggambar polyline tertutup dapat dilakukan dengan:

```
glBegin (GL_LINE_LOOP) ;  
glVertex2f (x1 , y1) ;  
glVertex2f (x2 , y2) ;  
.....  
glVertex2f (xn , yn) ;  
glEnd () ;
```

- Polyline dibentuk dari (x_1, y_1) ke (x_n, y_n) secara berurutan, dan kembali ke (x_1, y_1) .

Contoh Menggambar Polyline Tertutup

```
glBegin(GL_LINE_LOOP) ;  
glVertex2f(200.,100.) ;  
glVertex2f(0.,0.) ;  
glVertex2f(0.,-100.) ;  
glEnd() ;
```



Menggambar Polygon

- Untuk menggambar polygon dapat dilakukan dengan:

```
glBegin (GL_POLYGON) ;
```

```
glVertex2f (x1 , y1) ;
```

```
glVertex2f (x2 , y2) ;
```

```
.....
```

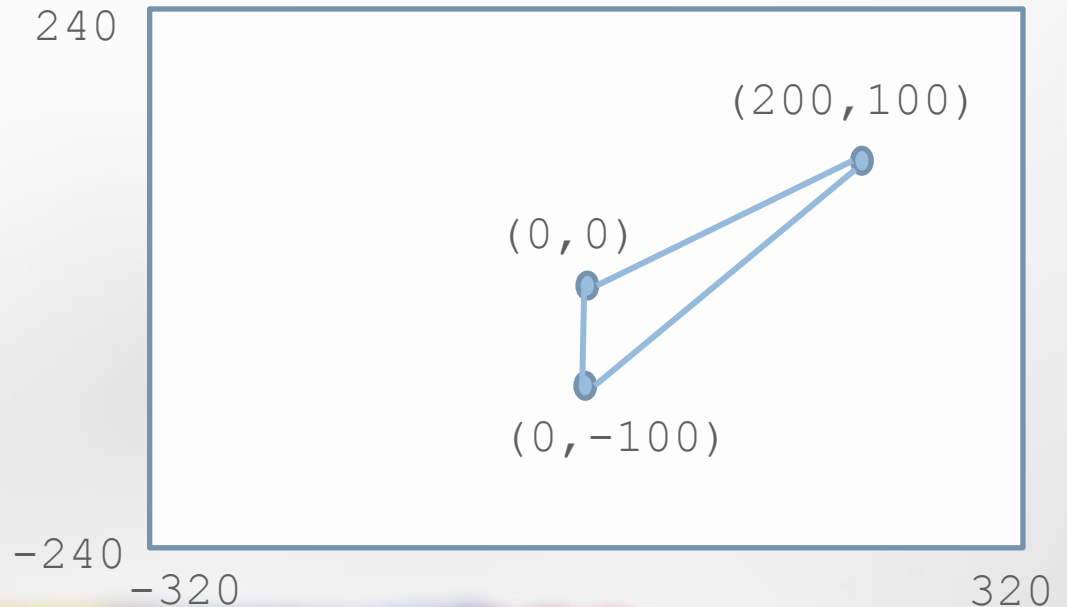
```
glVertex2f (xn , yn) ;
```

```
glEnd () ;
```

- Polygon dibentuk dari (x_1, y_1) ke (x_n, y_n) secara berurutan.

Contoh Menggambar Polygon

```
glBegin(GL_POLYGON);  
glVertex2f(200.,100.);  
glVertex2f(0.,0.);  
glVertex2f(0.,-100.);  
glEnd();
```



Mengganti Warna

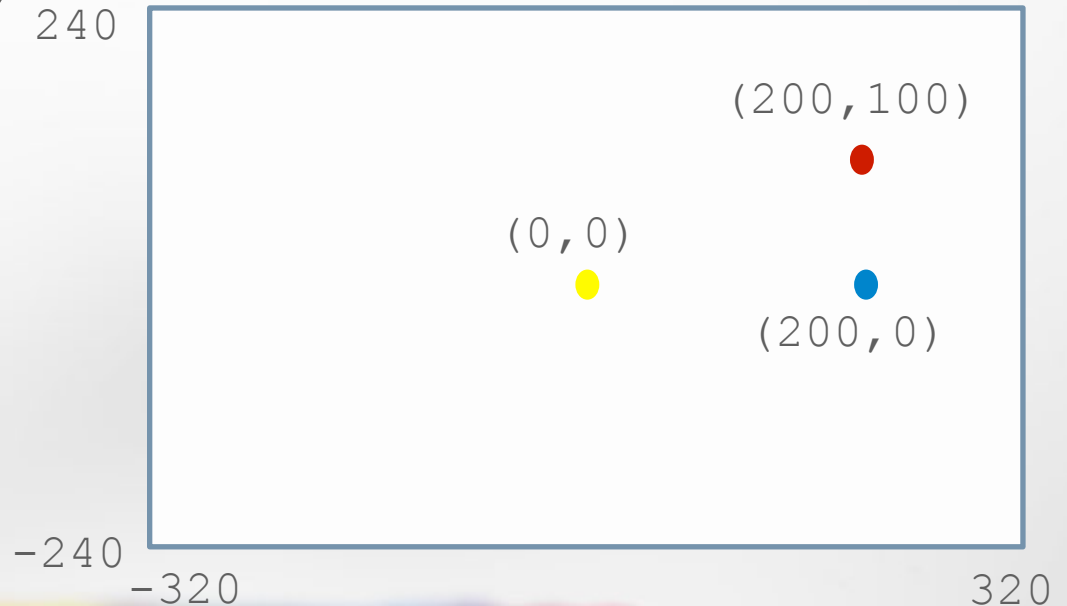
- Untuk mengganti warna dapat dilakukan dengan:
 - `glColor3f (r , g , b) ;`
- Perintah dituliskan sebelum menggambar obyek.
- Warna dibentuk dari tiga komponen; red, green dan blue.
- Masing-masing komponen r, g dan b mempunyai range nilai [0,1]

Warna

```
glColor3f(0.,0.,0.); //black  
glColor3f(0.,0.,1.); //blue  
glColor3f(0.,1.,0.); //green  
glColor3f(0.,1.,1.); //cyan  
glColor3f(1.,0.,0.); //red  
glColor3f(1.,0.,1.); //magenta  
glColor3f(1.,1.,0.); //yellow  
glColor3f(1.,1.,1.); //white
```

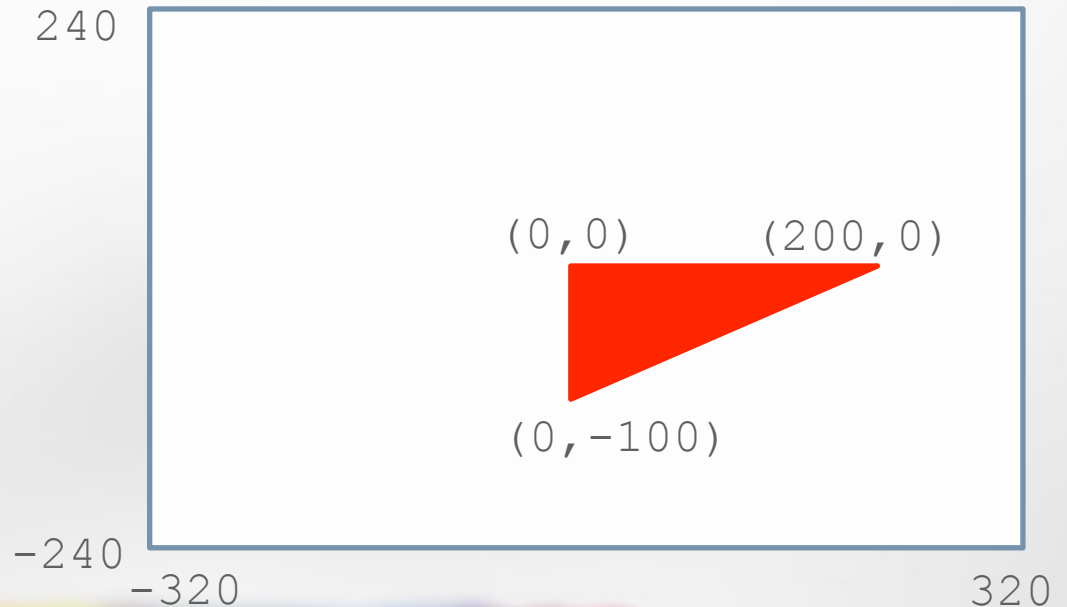
Mengubah Titik

```
glBegin(GL_POINTS);  
glColor3f(1,0,0);  
glVertex2f(200.,100.);  
glColor3f(1,1,0);  
glVertex2f(0.,0.);  
glColor3f(0,0,1);  
glVertex2f(200.,0.);  
glEnd();
```



Mewarnai Polygon

```
glColor3f(1,0,0);  
glBegin(GL_POLYGON);  
glVertex2f(200.,0.);  
glVertex2f(0.,0.);  
glVertex2f(0.,-100.);  
glEnd();
```





Sekian

- Hal tersulit dari sebuah pekerjaan adalah saat memulainya -