Monster Hunt Tutorial 3: Game

Loading Objects

```
#include "DarkGDK.h"

void DarkGDK ( void )
{
	float X=5000,Y=0,Z=5000,CameraAngleY=0,XTest=10,ZTest=10;
	dbSyncOn( );		dbSyncRate ( 60 );
	dbHideMouse();		dbAutoCamOff();
	dbBackdropOn();		dbSetCameraRange(1,5000);
	dbFogOn();		dbFogDistance(4000);
	dbFogColor(dbRGB(128,128,128));
	dbColorBackdrop(dbRGB(128,128,128));
	dbMakeMatrix(1,10000,10000,20,20);
	dbLoadImage("grass.bmp",1);
	dbPrepareMatrixTexture(1,1,1,1);
	dbFillMatrix(1,0,1);
	dbRandomizeMatrix(1,125);
	dbUpdateMatrix(1);

	dbLoadObject("IDLE.X",2);
	dbLoopObject(2);
	dbPositionObject(2,5000,dbGetGroundHeight(1,5000,5000),5000);
	dbPositionCamera(X,Y+35,Z-100);
	while ( LoopGDK ( ) )
	{
		dbSetCursor(0,0);
		dbPrint(X);
		dbPrint(Y);
		dbPrint(Z);
		CameraAngleY=dbCameraAngleY();
		if(dbUpKey()==1){
			XTest=dbNewXValue(X,CameraAngleY,20);
			ZTest=dbNewZValue(Z,CameraAngleY,20);
			if((XTest>500 && XTest<9500) &&
				(ZTest>500 && ZTest<9500)){
					dbMoveCamera(10);
			}
		}
		if(dbLeftKey()==1) dbYRotateCamera(dbWrapValue(CameraAngleY-5));
		if(dbRightKey()==1) dbYRotateCamera(dbWrapValue(CameraAngleY+5));
		X=dbCameraPositionX();
		Z=dbCameraPositionZ();
		Y=dbGetGroundHeight(1,X,Z);
		dbPositionCamera(X,Y+35,Z);

		dbSync();
	}
	return;
}
```

Mouse Look

```
#include "DarkGDK.h"

void DarkGDK ( void )
{
	float  X=5000,Y=0,Z=5000,CameraAngleY=0,CameraAngleX=0,
	       XTest=10,ZTest=10;
	dbSyncOn   ( );   dbSyncRate ( 60 );
	dbHideMouse();    dbAutoCamOff();
	dbBackdropOn();   dbSetCameraRange(1,5000);
	dbFogOn();        dbFogDistance(4000);
	dbFogColor(dbRGB(128,128,128));
	dbColorBackdrop(dbRGB(128,128,128));
	dbMakeMatrix(1,10000,10000,20,20);
	dbLoadImage("grass.bmp",1);
	dbPrepareMatrixTexture(1,1,1,1);
	dbFillMatrix(1,0,1);
	dbRandomizeMatrix(1,125);
	dbUpdateMatrix(1);
	dbLoadObject("IDLE.X",2);
	dbLoopObject(2);
	dbPositionObject(2,5000,dbGetGroundHeight(1,5000,5000),5000);
	dbPositionCamera(X,Y+35,Z-100);
	while ( LoopGDK ( ) )
	{
		dbSetCursor(0,0);
		dbPrint(X); dbPrint(Y); dbPrint(Z);
		float OldCamAngleY = CameraAngleY;
		float OldCamAngleX = CameraAngleX;
		CameraAngleY = dbWrapValue(CameraAngleY + dbMouseMoveX()*0.2);
		CameraAngleX = dbWrapValue(CameraAngleX + dbMouseMoveY()*0.2);
		if(dbUpKey()==1){
			XTest=dbNewXValue(X,CameraAngleY,10);
			ZTest=dbNewZValue(Z,CameraAngleY,10);
			if((XTest>500 && XTest<9500)&&(ZTest>500 && ZTest<9500)){
				X = XTest;  Z = ZTest;
			}
		}
		if(dbDownKey()==1) {
			XTest=dbNewXValue(X,dbWrapValue(CameraAngleY-180),10);
			ZTest=dbNewZValue(Z,dbWrapValue(CameraAngleY-180),10);
			if( (XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
				X=XTest;
				Z=ZTest;
			}
		}
		dbYRotateCamera(dbCurveAngle(CameraAngleY,OldCamAngleY,24));
		dbXRotateCamera(dbCurveAngle(CameraAngleX,OldCamAngleX,24));
		Y=dbGetGroundHeight(1,X,Z);
		dbPositionCamera(X,Y+50,Z);
		dbSync();
	}
	return;
}
```

Strafe

Insert these additional code:

```
if(dbLeftKey()==1){
	XTest = dbNewXValue(X,dbWrapValue(CameraAngleY-90),10);
	ZTest = dbNewZValue(Z,dbWrapValue(CameraAngleY-90),10);
	if((XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
		X=XTest;
		Z=ZTest;
	}
}

if(dbRightKey()==1){
	XTest = dbNewXValue(X,dbWrapValue(CameraAngleY+90),10);
	ZTest = dbNewZValue(Z,dbWrapValue(CameraAngleY+90),10);
	if( (XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
		X=XTest;
		Z=ZTest;
	}
}
```

Shooting Bullets

```
#include "DarkGDK.h"

void DarkGDK ( void )
{	dbSetWindowOff();
	float X=5000,Y=0,Z=5000,CameraAngleY=0,CameraAngleX=0,
	XTest=10,ZTest=10;
	int BulletLife=0;
	dbSyncOn   ( );
	dbSyncRate ( 60 );
	dbHideMouse();
	dbAutoCamOff();
	dbBackdropOn();
	dbSetCameraRange(1,5000);
	dbFogOn();
	dbFogDistance(4000);
	dbFogColor(dbRGB(128,128,128));
	dbColorBackdrop(dbRGB(128,128,128));
	dbMakeMatrix(1,10000,10000,20,20);
	dbLoadImage("grass.bmp",1);
	dbPrepareMatrixTexture(1,1,1,1);
	dbFillMatrix(1,0,1);
	dbRandomizeMatrix(1,125);
	dbUpdateMatrix(1);

	//dbLoadObject("IDLE.X",2);
	//dbLoopObject(2);
	//dbPositionObject(2,5000,dbGetGroundHeight(1,5000,5000),5000);
	dbPositionCamera(X,Y+35,Z-100);
```

```
//Make Gun
dbMakeObjectCylinder(1,2);
dbXRotateObject(1,30);
dbFixObjectPivot(1);
dbScaleObject(1,100,100,1000);
dbPositionObject(1,0,-8,15);
dbLockObjectOn(1);

//Make bullet
dbMakeObjectSphere(2,2);
dbHideObject(2);
while ( LoopGDK ( ) )
{
	dbSetCursor(0,0);
	dbPrint(X); dbPrint(Y); dbPrint(Z);
	float OldCamAngleY = CameraAngleY;
	float OldCamAngleX = CameraAngleX;
	CameraAngleY = dbWrapValue(CameraAngleY + dbMouseMoveX()*0.2);
	CameraAngleX = dbWrapValue(CameraAngleX + dbMouseMoveY()*0.2);
	if(dbUpKey()==1){
		XTest=dbNewXValue(X,CameraAngleY,10);
		ZTest=dbNewZValue(Z,CameraAngleY,10);
		if((XTest>500 && XTest<9500)&&(ZTest>500 && ZTest<9500)){
				X = XTest;
				Z = ZTest;
		}
	}
	if(dbDownKey()==1) {
		XTest=dbNewXValue(X,dbWrapValue(CameraAngleY-180),10);
		ZTest=dbNewZValue(Z,dbWrapValue(CameraAngleY-180),10);
		if( (XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
			X=XTest;
			Z=ZTest;
		}
	}
	if(dbLeftKey()==1){
		XTest = dbNewXValue(X,dbWrapValue(CameraAngleY-90),10);
		ZTest = dbNewZValue(Z,dbWrapValue(CameraAngleY-90),10);
		if((XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
				X=XTest;
				Z=ZTest;
			}
	}

	if(dbRightKey()==1){
		XTest = dbNewXValue(X,dbWrapValue(CameraAngleY+90),10);
		ZTest = dbNewZValue(Z,dbWrapValue(CameraAngleY+90),10);
		if( (XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
			X=XTest;
			Z=ZTest;
		}
	}
	dbYRotateCamera(dbCurveAngle(CameraAngleY,OldCamAngleY,24));
	dbXRotateCamera(dbCurveAngle(CameraAngleX,OldCamAngleX,24));
```

```
		Y=dbGetGroundHeight(1,X,Z);
		dbPositionCamera(X,Y+50,Z);

		if(dbMouseClick()==1 && BulletLife==0){
			dbPositionObject(2,X,Y+43,Z);
			dbSetObjectToCameraOrientation(2);
			BulletLife =25;
			dbShowObject(2);
		}
		if(BulletLife > 0){
			BulletLife--;
			dbMoveObject(2,20);
			if(BulletLife == 0) dbHideObject(2);
		}

		dbSync();
	}
	return;
}
```

Hunt and Shoot Monster

```
#include "DarkGDK.h"

void DarkGDK ( void )
{	dbSetWindowOff();
	float X=1000,Y=0,Z=1000,CameraAngleY=0,CameraAngleX=0,
	XTest=10,ZTest=10;

	int BulletLife=0;
	dbSyncOn   ( );
	dbSyncRate ( 60 );
	dbHideMouse();
	dbAutoCamOff();
	dbBackdropOn();
	dbSetCameraRange(1,5000);
	dbFogOn();
	dbFogDistance(4000);
	dbFogColor(dbRGB(128,128,128));
	dbColorBackdrop(dbRGB(128,128,128));
	dbMakeMatrix(1,10000,10000,20,20);
	dbLoadImage("grass.bmp",1);
	dbPrepareMatrixTexture(1,1,1,1);
	dbFillMatrix(1,0,1);
	dbRandomizeMatrix(1,125);
	dbUpdateMatrix(1);

	dbLoadObject("IDLE.X",300);
	dbLoopObject(300);
	dbPositionObject(300,5000,dbGetGroundHeight(1,5000,5000),5000);
	dbPositionCamera(X,Y+35,Z-100);

	//Make Gun
	dbMakeObjectCylinder(1,2);
	dbXRotateObject(1,30);
	dbFixObjectPivot(1);
	dbScaleObject(1,100,100,1000);
	dbPositionObject(1,0,-8,15);
	dbLockObjectOn(1);

	//Make bullet
	dbMakeObjectSphere(2,2);
	dbHideObject(2);
	dbSetObjectCollisionToBoxes(2);

	dbLoadSound( "crickets.wav",1);
	dbLoopSound (1);
	dbLoad3DSound ("fireball.wav",2);

	//rem make HUD
	dbMakeObjectPlain(200,1,1);
	dbPositionObject( 200,-2.7,1.9,4);
```

```
dbLockObjectOn(200);
dbGhostObjectOn( 200);
//rem Load and create hud bitmaps.
dbLoadBitmap( "radar.bmp",2);
dbCreateBitmap( 1,100,100);
while ( LoopGDK ( ) )
{
	//dbSetCursor(0,200);
	//dbPrint(X);	dbPrint(Y); dbPrint(Z);dbPrint(CameraAngleX);
	float OldCamAngleY = CameraAngleY;
	float OldCamAngleX = CameraAngleX;
	CameraAngleY = dbWrapValue(CameraAngleY + dbMouseMoveX()*0.2);
	CameraAngleX = dbWrapValue(CameraAngleX + dbMouseMoveY()*0.2);
	if(dbUpKey()==1){
		XTest=dbNewXValue(X,CameraAngleY,10);
		ZTest=dbNewZValue(Z,CameraAngleY,10);
		if((XTest>500 && XTest<9500) &&(ZTest>500 && ZTest<9500)){
				X = XTest;
				Z = ZTest;
		}
	}
	if(dbDownKey()==1) {
		XTest=dbNewXValue(X,dbWrapValue(CameraAngleY-180),10);
		ZTest=dbNewZValue(Z,dbWrapValue(CameraAngleY-180),10);
		if( (XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
			X=XTest;
			Z=ZTest;
		}
	}
	if(dbLeftKey()==1){
		XTest = dbNewXValue(X,dbWrapValue(CameraAngleY-90),10);
		ZTest = dbNewZValue(Z,dbWrapValue(CameraAngleY-90),10);
		if((XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
				X=XTest;
				Z=ZTest;
		}
	}

	if(dbRightKey()==1){
		XTest = dbNewXValue(X,dbWrapValue(CameraAngleY+90),10);
		ZTest = dbNewZValue(Z,dbWrapValue(CameraAngleY+90),10);
		if( (XTest>0 && XTest<10000) && (ZTest>0 && ZTest<10000)){
			X=XTest;
			Z=ZTest;
		}
	}
	//if(CameraAngleX>270){
	//	if(CameraAngleX-270>90) CameraAngleX=270;
//else if(CameraAngleX>90 && CameraAngleX-270<270) CameraAngleX=90;
	//}
	dbYRotateCamera(dbCurveAngle(CameraAngleY,OldCamAngleY,24));
	dbXRotateCamera(dbCurveAngle(CameraAngleX,OldCamAngleX,24));

	Y=dbGetGroundHeight(1,X,Z);
	dbPositionCamera(X,Y+50,Z);
```

```
		if(dbMouseClick()==1 && BulletLife==0){
			dbPositionObject(2,X,Y+43,Z);
			dbSetObjectToCameraOrientation(2);
			BulletLife =25;
			dbShowObject(2);
			dbLoopSound(2);
		}
		if(BulletLife > 0){
			BulletLife--;
			dbMoveObject(2,20);
			dbSetCursor( 10,10);
			if(dbObjectCollision(2,300)>0) {
				dbText( 240,220 ,"hit hit hit hit hit hit hit hit");
				BulletLife = 0;
			}
		}
		if(BulletLife == 0){
			dbHideObject(2);
			dbStopSound(2);
		}
		//Rem make radar
		dbCopyBitmap( 2,1);
		dbSetCurrentBitmap( 1);
		dbInk(dbRGB(0,0,255),dbRGB(0,0,0));
		float PRX=X/200;
		float PRZ=50-(Z/200);
		dbCircle( PRX,PRZ,1);
		dbInk(dbRGB(255,0,0),dbRGB(0,0,0));
		float mX = dbObjectPositionX (300);
		float mZ = dbObjectPositionZ (300);
		float MRX=mX/200;
		float MRZ=50-(mZ/200);
		dbCircle( MRX,MRZ,1);
		dbGetImage(200,0,0,100,100);
		dbSetCurrentBitmap( 0);
		dbTextureObject( 200,200);
		dbInk(dbRGB(255,128,128),dbRGB(0,0,0));
		dbSync();
	}

	return;
}
```

References

The Game Creators (2010). Monster Hunt Tutorial . Retrieved, Feb 8, 2010, from: http://developer.thegamecreators.com/?f=t01/3d_tutorial_index