

Small Tutorial

The following sample will create an entity class called **Fan**.

- Create a new file with the extension **.ent**, for example `GameSDK\Entities\Fan.ent`. This entity definition file will be used to expose the entity to the engine.

```
<Entity
  Name="Fan"
  Script="Scripts/Entities/Fan.lua"
/>
```

- Create a new **.lua**, for example `GameSDK\Entities\Scripts\Fan.lua`. The lua file will define the entity logic.

```
Fan = {
  type = "Fan", -- can be useful for scripting

  -- instance member variables
  minrotspeed = 0,
  maxrotspeed = 1300,
  acceleration = 300,
  currrotspeed = 0,
  changespeed = 0,
  currangle = 0,

  -- the entries of the following table become automatically exposed to the editor and serialized (load/save)
  -- the type is defined by the prefix (for more prefix types search for s_paramTypes in Sandbox/Editor/Objects
/EntityScript.cpp)
  Properties = {
    bName = 0, -- boolean example, 0/1
    fName = 1.2, -- float example
    soundName = "", -- sound example
    fileModelName = "Objects/box.cgf", -- file model
  },

  -- optional editor information
  Editor = {
    Model = "Editor/Objects/Particles.cgf", -- optional 3d object that represents this
object in editor
    Icon = "Clouds.bmp", -- optional 2d icon that represents this object
in editor
  },
}

-- optional. Called only once on loading a level. Consider calling self:OnReset(not System.IsEditor()); here
function Fan:OnInit()
  self:SetName( "Fan" );
  self:LoadObject( "Objects/Indoor/Fan.cgf", 0, 0 );
  self:DrawObject( 0, 1 );
end

-- OnReset() is usually called only from the Editor, so we also need OnInit()
-- Note the parameter
function Fan:OnReset(bGameStarts)
end

-- optional. To start having this callback called, you should activate the entity like this:
-- self:Activate(1); -- Turn on OnUpdate() callback
function Fan:OnUpdate(dt)
  if ( self.changespeed == 0 ) then
    self.currrotspeed = self.currrotspeed - System.GetFrameTime() * self.acceleration;
    if ( self.currrotspeed < self.minrotspeed ) then
      self.currrotspeed = self.minrotspeed;
    end
  end
else
```

```

        self.currrotspeed = self.currrotspeed + System.GetFrameTime() * self.acceleration;
        if ( self.currrotspeed > self.maxrotspeed ) then
            self.currrotspeed = self.maxrotspeed;
        end
    end
    self.currangle = self.currangle + System.GetFrameTime() * self.currrotspeed;
    local a = { x=0, y=0, z=-self.currangle };
    self:SetAngles( a );
end

-- optional serialization
function Fan:OnSave(tbl)
    tbl.currangle = self.currangle;
end

-- optional serialization
function Fan:OnLoad(tbl)
    self.currangle = tbl.currangle;
end

-- optional
function Fan:OnSpawn()
end

-- optional
function Fan:OnDestroy()
end

-- optional
function Fan:OnShutDown()
end

-- optional
function Fan:OnActivate()
    self.changespeed = 1 - self.changespeed;
end

```

See Also

- [Entity Property Prefixes](#)