```
void OgreBulletExample::createConcaveRigidBodyObject(Ogre::String mesh, Ogre::Vector3 position){
    //entity
    Ogre::Entity *objEntity = mSceneMgr->createEntity("Object_" +
            Ogre::StringConverter::toString(m_numEntities),
            mesh);

    //node
    Ogre::SceneNode *objNode = mSceneMgr->getRootSceneNode()->createChildSceneNode();
    objNode->attachObject(objEntity);

    //collision shape
    OgreBulletCollisions::StaticMeshToShapeConverter makeShape(objEntity);
    //Just make a sphere around the object
    //OgreBulletCollisions::CollisionShape *objShape = makeShape.createSphere();

    //This one only works with static shapes
    //OgreBulletCollisions::CollisionShape *objShape = makeShape.createConcave();

    //Decompose a concave object into a collection of convex objects
    OgreBulletCollisions::CollisionShape *objShape = makeShape.createConvexDecomposition();


    //rigid body
    OgreBulletDynamics::RigidBody *objBody =
            new OgreBulletDynamics::RigidBody("ObjRigid" +
                    Ogre::StringConverter::toString(m_numEntities),
                    m_world);

    objBody->setShape(objNode, objShape, 0.6f, 0.6f, 10.0f, position);
    //If you would rather have a static shape
    //objBody->setStaticShape(objNode, objShape, 0.6f, 0.6f, position);

    m_numEntities++;

    m_shapes.push_back(objShape);
    m_bodies.push_back(objBody);
}
```

#include <shapes/.....h>