Rigid Body Dynamics

(SENG 463 - Game Programming)

Dr.Çağatay ÜNDEĞER
Research and Innovation Director
SimBT Inc.

e-mail :
cagatay.undeger@simbt.com.tr
cagatay@undeger.com

Outline

- Rigid Body Dynamics
  - General Principles
  - Linear Dynamics
  - Angular Dynamics
  - Collision Response
  - Friction
- Rigid Body Dynamics In Unity

Solid Objects In Real Life

- In Real Life, solid objects generally avoid doing impossible things,
  - Like passing through one another, all by themselves.
- But in a game world, objects don't do anything unless we tell them to do,
- Game programmers must make an explicit effort to ensure that objects do not pass through one another.

Rigid Body Dynamics

- In a game physics engine, we are particularly concerned with the kinematics of objects to simulate how they move over time.
- These simulations apply numerical methods to existing theories to obtain results that are as close as possible to what we observe in the real world.
- In Games, we usually deal with rigid objects.
- The physic computations on these rigid objects is called Rigid Body Dynamics.

Rigid Body Dynamics

- Game physics systems usually focuses on Classical Rigid Body Dynamics:
  - In Classical (Newtonian) mechanics, objects are assumed to obey Newton's laws of motion. The objects are large and slow enough that there are no quantum effects.
  - All objects in the simulation are perfectly solid rigid bodies and cannot be deformed. In other words, their shapes are constant.

Rigid Body Dynamics

- The physics system usually shares the collision world data structure, and
- Usually drives the execution of the collision detection algorithm as part of its time step update routine.
- The rigid bodies in the physics engine are typically distinct from the logical objects that make up the game world.

Rigid Body Dynamics

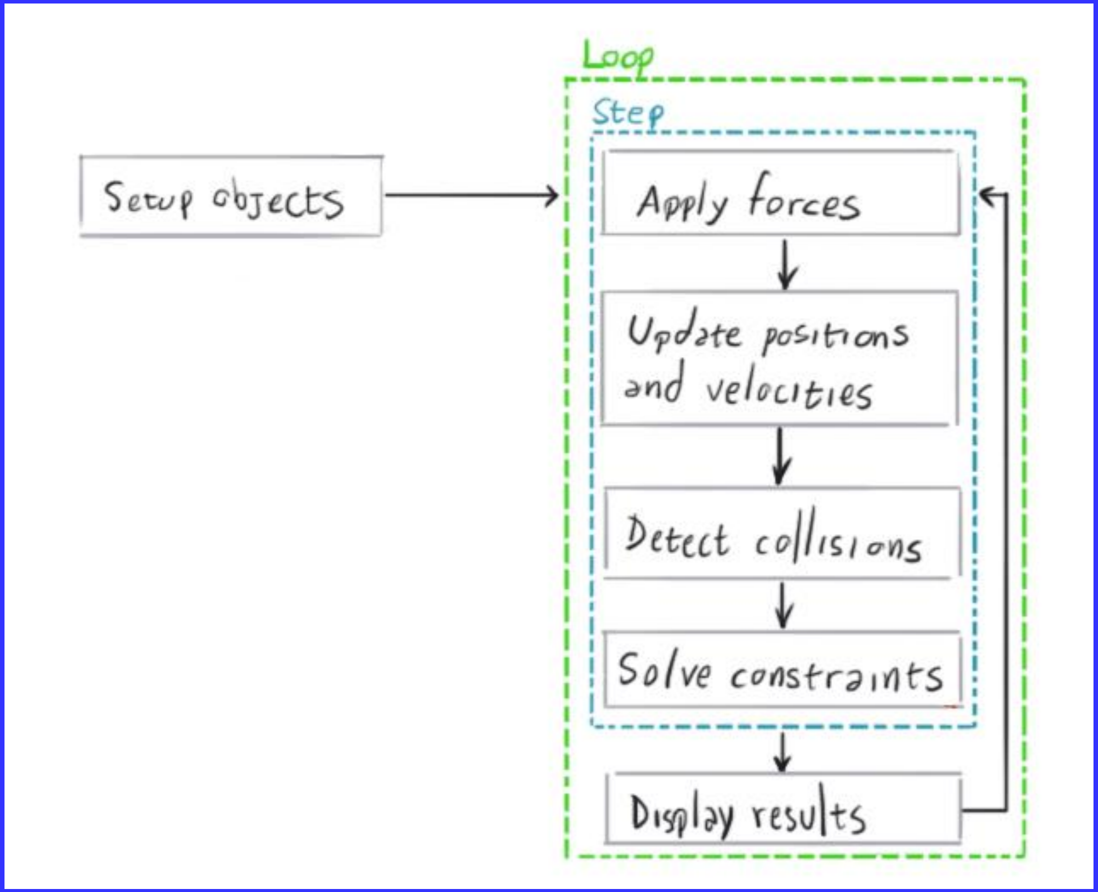
- The positions and orientations of game objects can be driven by the physics simulation.
- We query the physics engine every frame for the transform of each rigid body,
- Apply rigid body to the transform of the corresponding game object.

Newtonian Machanics

- Isaac Newton's **Three Laws of Motion**:
  - **Inertia**: If no force is applied on an object, its velocity & direction shall not change.
  - **Force, Mass, and Acceleration**: Force acting on an object is equal to mass of object multiplied by its acceleration ***(F = ma)***
  - **Action and Reaction**: "For every action there is an equal and opposite reaction." Whenever one body exerts a force on another, the second body exerts a force of the same magnitude and opposite direction on the first.

Simple Physics Engine Loop

- A high level overview of the general procedure of a physics engine



Units in Computations

- Most rigid body dynamics simulations operate in the MKS system of units.
- In this system,
  - Distance is measured in meters (m),
  - Mass is measured in kilograms (kg) and
  - Time is measured in seconds (s).

Degrees of Freedom

- An unconstrained rigid body is one
  - That can translate freely along all three Cartesian axes (x,y,z of Position) and
  - That can rotate freely about these three axes (euler angles of Rotation).

- We say that such a body has six degrees of freedom (6 - DOF).

Separability of Linear and Angular Dynamics

- Motion of an unconstrained rigid body can be separated into two independent components:
  - Linear dynamics:
    - Linear motion of the body when we ignore all rotational effects (describes the motion of an idealized point mass)
  - Angular dynamics:
    - Rotational motion of the body.
- Ability to separate linear and angular components of a rigid body's motion is extremely helpful.

Center of Mass / Gravity

- For the purposes of linear dynamics,
  - An unconstrained rigid body acts as if all of its mass were at a single point known as the center of mass (CM or COM).
- CM is essentially the balancing point of the body for all possible orientations.
- The mass of a rigid body is distributed evenly around its CM in all directions.

Center of Mass / Gravity

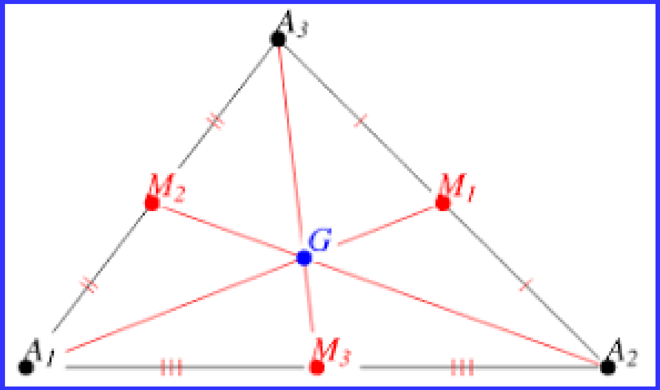
- For a body with uniform density, the center of mass lies at the centroid of the body.
- If we are simulating particals with simple shapes such as quad, cube, sphere,
  - CMs will be equal to positions of particals.
- If the body's density is not uniform,
  - Position of each little piece would need to be weighted by piece's mass (weighted average)

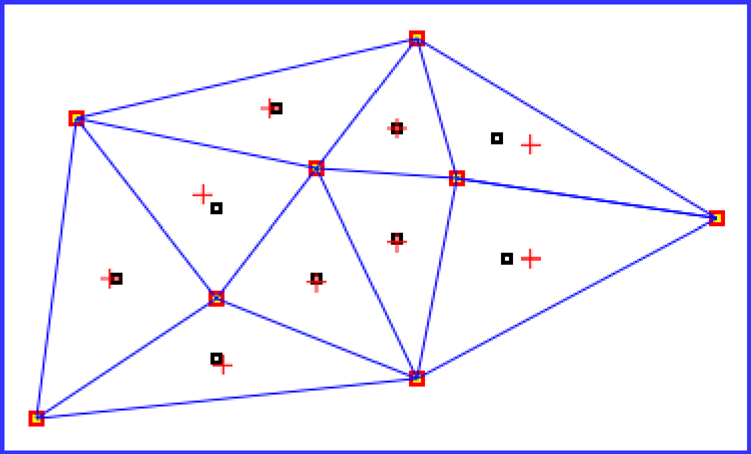
$$\mathbf{r}_{\mathrm{CM}} = \frac{\sum_{\forall i} m_i \mathbf{r}_i}{\sum_{\forall i} m_i} = \frac{\sum_{\forall i} m_i \mathbf{r}_i}{m},$$

symbol r represents a radius vector or position vector (World coordinate)

Center of Mass (Triangle)

- The center of gravity of a triangle (= centroid):
  - Is where the three medians intersect,
  - But since the medians only intersect in one point, you can use two of medians.

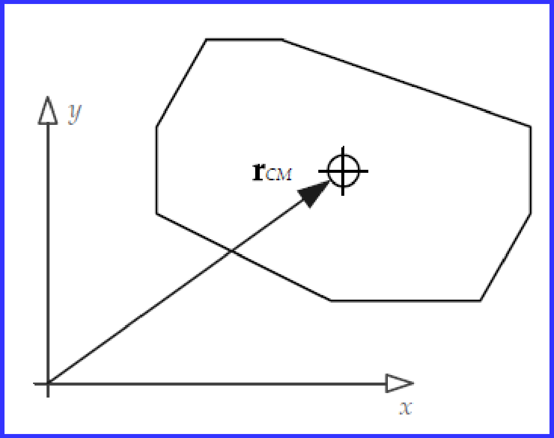




center of gravity =
weighted sum of center of gravity of triangles

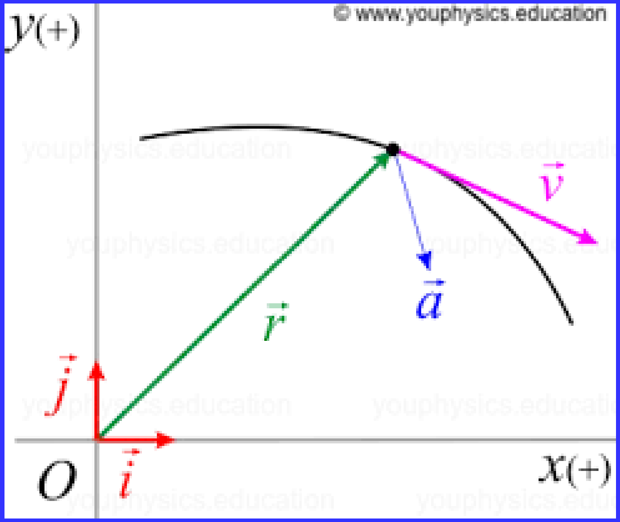
Linear Dynamics

- Position of a rigid body is described by a position vector r_{CM}
- The linear velocity of a rigid body defines the speed and direction in which the body's CM is moving.
- A velocity vector is measured in m/s



Linear Dynamics

- Accelaration (a) describes the linear velocity change per second
- An accelaration vector is measured in m/s^2



Linear Dynamics

- A force (F) (kg.m/s^2) is defined as anything that causes an object with mass to accelerate or decelerate.
- A force has both a magnitude and a direction in space, so all forces are represented by vectors.
- You can sum up force vectors to find their total force on the body’s linear motion.

$$\mathbf{F}(t) = m\mathbf{a}(t)$$

$$\mathbf{F}_{\text{net}} = \sum_{i=1}^{N} \mathbf{F}_i .$$

Linear Dynamics

- When we multiply a body's linear velocity by its mass,
  - The result is a quantity known as linear momentum (p) (kg.m/s).

$$\mathbf{p}(t) = m\mathbf{v}(t).$$

Linear Dynamics

- The velocity vector is the time derivative of the position vector

$$\vec{v} = \frac{d\vec{r}}{dt} = \frac{dx}{dt}\vec{i} + \frac{dy}{dt}\vec{j} + \frac{dz}{dt}\vec{k}\ (m/s)$$

- The acceleration vector is the time derivative of the velocity vector

$$\vec{a} = \frac{d\vec{v}}{dt} = \frac{dv_x}{dt}\vec{i} + \frac{dv_y}{dt}\vec{j} + \frac{dv_z}{dt}\vec{k}\ (m/s^2)$$

- To simplify differential equations we use numerical computations applied to the theoretical laws of physics

Numerical Computations

- Solve our differential equations in a time-stepped manner
- Using the solution from a previous time step
- To arrive at the solution for the next time step.
- Duration of the time step is usually taken to be a constant “Delta Time”

$$t_2 = t_1 + \Delta t.$$

Next position

$$\mathbf{r}(t_2) = \mathbf{r}(t_1) + \mathbf{v}(t_1)\Delta t.$$

$$\mathbf{a}(t) = \frac{\mathbf{F}_{\text{net}}(t)}{m}$$

Next velocity

$$\mathbf{v}(t_2) = \mathbf{v}(t_1) + \frac{\mathbf{F}_{\text{net}}(t)}{m}\Delta t.$$

Explicit Euler Method

Angular Dynamics

- Since a rigid body can also rotate,
  - We have to introduce its angular properties, which are analogous to its linear properties.
- We start with 2D angular dynamics first.
- In 2D, every rigid body can be treated as a thin sheet of material.

Angular Dynamics

- Angular velocity measures the rate at which a body's rotation angle changes over time.
- Angular velocity is $\omega(t)$ and measured in radians per second (rad/s).

- Angular Accelaration is $\alpha(t)$ and describes angular velocity change per second (rad/s^2).

- To gain angular accelaration and velocity the body needs to receive some rotational force called Torque τ.

Angular Dynamics

- Torque τ is analogous to Force.

$$\mathbf{F} = m\mathbf{a} \longrightarrow \tau = I\alpha$$

- The rotational equivalent of mass is the moment of inertia (I).
- Mass describes how easy or difficult it is to change the linear velocity of a point mass.
- The moment of inertia measures how easy or difficult it is to change the angular speed of a rigid body about a particular axis.

Angular Dynamics

- Moment of inertia of a 2D box with mass *m*, width *w* and height *h* is (Axis of rotation at center)

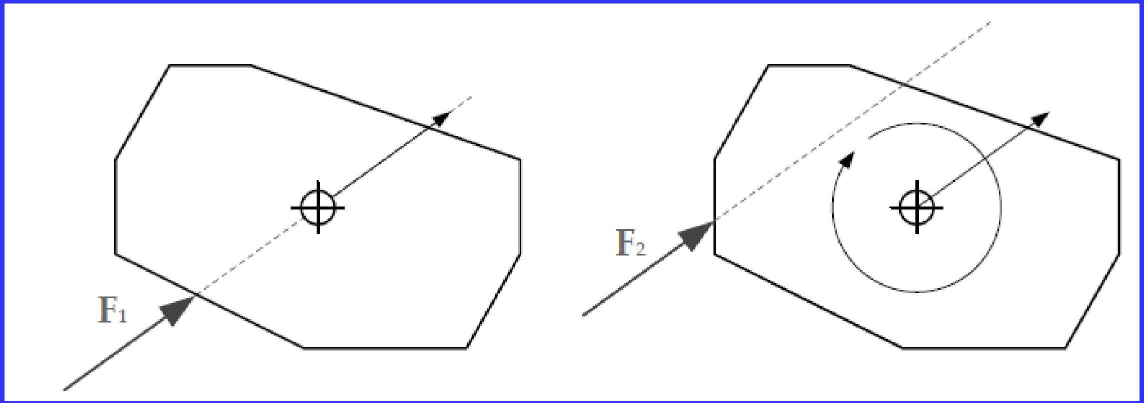
$$I = \frac{m(h^2 + w^2)}{12}$$

- You can find a list of formulas to compute the moment of inertia for a bunch of shapes:

  https://en.wikipedia.org/wiki/List_of_moments_of_inertia

Angular Dynamics

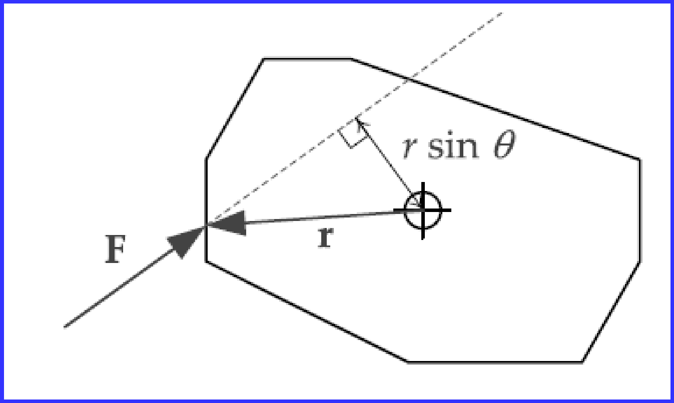
- In general, forces can be applied at arbitrary points on a body.
- If the line of action of a force passes through the body’s center of mass,
  - Then the force will produce linear motion only
  - Otherwise, the force will introduce a rotational force known as a torque in addition.



Angular Dynamics

- We can calculate torque using a cross product.
- We express the location at which the force is applied as a vector r extending from CM
- The torque N caused by a force F applied at a location r is cross product of r and F.

$$\mathbf{N} = \mathbf{r} \times \mathbf{F}.$$

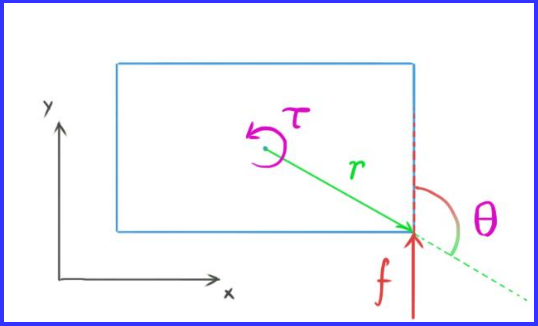


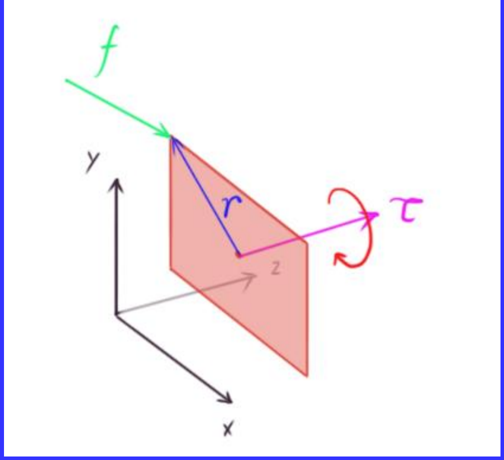
Torque vector is along Z axis,
Mean it will rotate around Z axis

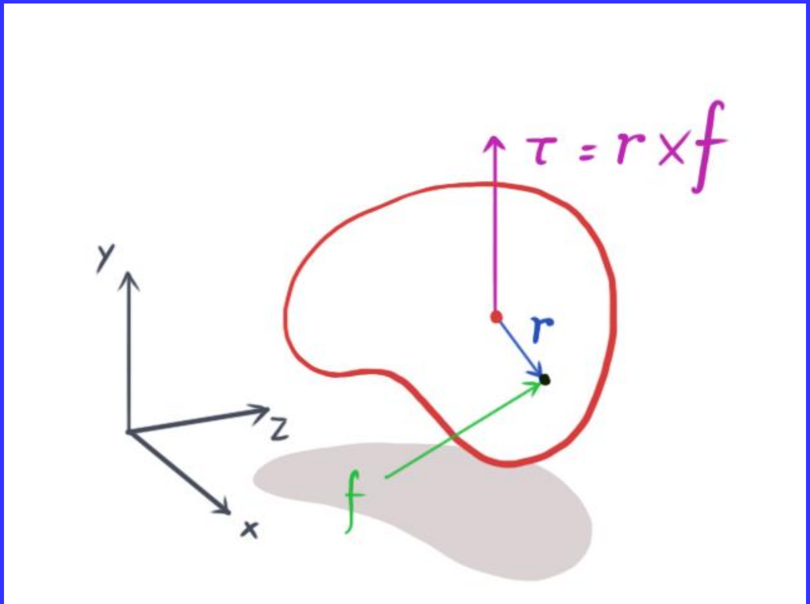
- Equation implies that torque increases as the force is applied farther from the center of mass.

Angular Dynamics

- When we use cross product,
  - The result torque vector is along Z axis,
  - That means torque will rotate around Z axis.







Angular Dynamics

- When two or more forces are applied to a rigid body,
  - The torque vectors produced by each one can be summed, just as we can sum forces.
- Torque is related to angular acceleration and moment of inertia
  - In much the same way that force is related to linear acceleration and mass.

Angular Dynamics

- Angular dynamics in 3D is more complex than its 2D counterpart, although the basic concepts are very similar.
- A rigid body may have a very different distribution of mass about the three coordinate axes.
- So we should expect a body to have different moments of inertia about different axes.

Angular Dynamics

- In 3D, the rotational mass of a rigid body is represented by a 3×3 matrix called Inertia Tensor.

$$\mathbf{I} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}$$

- Elements lying along the diagonal of this matrix are the moments of inertia about its three principal axes I_{xx}, I_{yy}, I_{zz}.
- The off-diagonal elements are called products of inertia.
- They are zero when the body is symmetrical about all three principal axes.

Collision Response

- So far we assume that our rigid bodies are
  - neither colliding with anything,
  - nor is their motion constrained in any way.
- When bodies collide with one another,
  - The dynamics simulation must take steps to ensure that they respond realistically to the collision.

Collision Response

- Work represents a change in energy:
  – A force either adds energy to a system of rigid bodies (e.g., an explosion) or
  – Removes energy from the system (e.g., friction).
- When two bodies collide in the real world,
  – The bodies compress slightly and then rebound,
  – changing their velocities and
  – losing energy to sound and heat in the process.

Collision Response

- Simplifying assumptions:
  - Collision force acts over a very short period of time, turning it into what we call an idealized impulse.
  - This causes the velocities of the bodies to change instantaneously as a result of the collision.
  - There is no friction at the point of contact.
  - Complex submolecular interactions among bodies can be approximated by a single quantity called coefficient of restitution (ε)

Collision Response

- Coefficient of restitution (ε) describes how much energy is lost during the collision.
- When ε = 1,
  - the collision is perfectly elastic and no energy is lost.
- When ε = 0,
  - the collision is perfectly inelastic,
  - Also known as perfectly plastic and the kinetic energy of both bodies is lost.
  - The bodies will stick together after the collision.

Collision Response

- All collision analysis is based around the idea that linear momentum is conserved.

$$\mathbf{p}_1 + \mathbf{p}_2 = \mathbf{p}_1' + \mathbf{p}_2', \qquad \text{or}$$
$$m_1\mathbf{v}_1 + m_2\mathbf{v}_2 = m_1'\mathbf{v}_1' + m_2'\mathbf{v}_2'$$

- But we must account for the energy lost due to heat and sound by introducing an additional energy loss term T_{lost}.

$$\tfrac{1}{2}m_1v_1^2 + \tfrac{1}{2}m_2v_2^2 = \tfrac{1}{2}m_1'{v_1'}^2 + \tfrac{1}{2}m_2'{v_2'}^2 + T_{\text{lost}}.$$

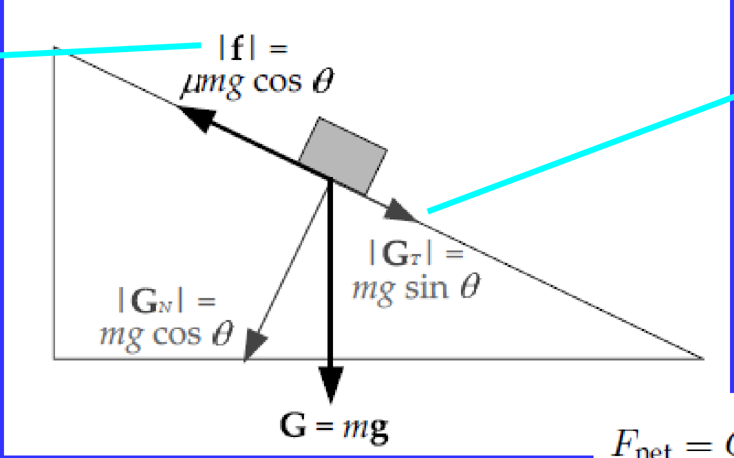
Friction

- Friction is a force that arises between two bodies that are in continuous contact,
  - Resisting their movement relative to one another.
- Static friction is the resistance one feels when trying to start a stationary object sliding along a surface.
- Dynamic friction is a resisting force that arises when objects are actually moving relative to one another.

Friction

- Linear sliding friction is proportional to the component of an object's weight that is acting normal to the surface on which it is sliding.
- The weight of an object is just the force due to gravity (G = mg),
  – Which is always directed downward.

Friction force (μ is coefficient of friction)



gravitational force acting tangent to the surface

$$F_{\text{net}} = G_T - f = mg(\sin\theta - \mu\cos\theta).$$

Rigid Body Dynamics In Unity

- To Put a Rigid Body Dynamic to a GameObject,
  - Put a collider to the GameObject.
  - If it is a MeshCollider,
    - It shall be convex.
  - Put a RigidBody to the GameObject.
  - Set mass of the RigidBody.
  - Set drag coefficient of the RigidBody.
  - Set angular drag coefficient of the RigidBody.

Rigid Body Dynamics In Unity

- For other static GameObjects
  – Just put a convex or concave collider.

- For changing surface frictions, assign Physics Materials to colliders

- Add a force or torque to the RigidBody at any position and direction in every FixedUpdate.