Fotorealistyczna Grafika Komputerowa

Tekstury

Adam Błaszczyk 239636
Antonina Matuszek 239687

1 Hierarchical Index 1

1.1 Class Hierarchy . . . 1

2 Class Index 3

2.1 Class List . . . 3

3 Class Documentation 5

3.1 ray.Hit Class Reference . . . 5

3.1.1 Detailed Description . . . 5

3.2 light_intensity.LightIntensity Class Reference . . . 5

3.2.1 Detailed Description . . . 6

3.3 light_source.LightSource Class Reference . . . 6

3.3.1 Detailed Description . . . 7

3.3.2 Constructor & Destructor Documentation . . . 7

3.3.2.1 __init__() . . . 7

3.4 material.Material Class Reference . . . 8

3.4.1 Detailed Description . . . 8

3.4.2 Member Data Documentation . . . 8

3.4.2.1 ambientColour . . . 8

3.4.2.2 diffuseColour . . . 9

3.4.2.3 reflectColour . . . 9

3.4.2.4 specularColour . . . 9

3.5 mesh.Mesh Class Reference . . . 9

3.5.1 Detailed Description . . . 10

3.6 image.MyImage Class Reference . . . 10

3.6.1 Detailed Description . . . 11

3.7 orthogonal_camera.OrthogonalCamera Class Reference . . . 11

3.7.1 Detailed Description . . . 12

3.8 perspective_camera.PerspectiveCamera Class Reference . . . 12

3.8.1 Detailed Description . . . 13

3.9 plane.Plane Class Reference . . . 13

3.9.1 Detailed Description . . . 14

3.10 point_light_source.PointLightSource Class Reference . . . 15

3.10.1 Detailed Description . . . 15

3.11 primitive.Primitive Class Reference . . . 16

3.12 ray.Ray Class Reference . . . 17

3.12.1 Detailed Description . . . 17

3.12.2 Member Data Documentation . . . 17

3.12.2.1 direction . . . 18

3.12.2.2 length . . . 18

3.12.2.3 origin . . . 18

3.12.2.4 target . . . 18

3.13 sphere.Sphere Class Reference . . . 19

3.13.1 Detailed Description . . . 20
3.13.2 Member Function Documentation . . . 20
3.13.2.1 get_texture_color() . . . 20
3.14 tests.Test Class Reference . . . 20
3.14.1 Detailed Description . . . 22
3.15 texture.Texture Class Reference . . . 22
3.15.1 Detailed Description . . . 23
3.15.2 Member Function Documentation . . . 23
3.15.2.1 spherical_mapping() . . . 23
3.16 triangle.Triangle Class Reference . . . 23
3.16.1 Detailed Description . . . 24
3.17 vector.Vec2 Class Reference . . . 25
3.17.1 Detailed Description . . . 26
3.17.2 Member Data Documentation . . . 26
3.17.2.1 x . . . 26
3.17.2.2 y . . . 26
3.18 vector.Vec3 Class Reference . . . 26
3.18.1 Detailed Description . . . 28
3.18.2 Member Data Documentation . . . 28
3.18.2.1 x . . . 28
3.18.2.2 y . . . 28
3.18.2.3 z . . . 28

Index **29**

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

ray.Hit . . . 5
light_intensity.LightIntensity . . . 5
LightSource
    point_light_source.PointLightSource . . . 15
material.Material . . . 8
image.MyImage . . . 10
orthogonal_camera.OrthogonalCamera . . . 11
perspective_camera.PerspectiveCamera . . . 12
Primitive
    mesh.Mesh . . . 9
    plane.Plane . . . 13
    sphere.Sphere . . . 19
    triangle.Triangle . . . 23
ray.Ray . . . 17
TestCase
    tests.Test . . . 20
texture.Texture . . . 22
vector.Vec2 . . . 25
vector.Vec3 . . . 26
ABC
    light_source.LightSource . . . 6
    primitive.Primitive . . . 16

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

ray.Hit
Documentation for a class Hit . . . 5
light_intensity.LightIntensity
Documentation for a class LightIntensity . . . 5
light_source.LightSource
Documentation for a class LightSource . . . 6
material.Material
Documentation for a class Material . . . 8
mesh.Mesh
Documentation for a class Mesh . . . 9
image.MyImage
Documentation for a class MyImage . . . 10
orthogonal_camera.OrthogonalCamera
Class for othogonal camera . . . 11
perspective_camera.PerspectiveCamera
Class for othogonal camera . . . 12
plane.Plane
Documentation for a class Plane . . . 13
point_light_source.PointLightSource
Documentation for a class PointLightSource . . . 15
primitive.Primitive . . . 16
ray.Ray
Documentation for a class Ray . . . 17
sphere.Sphere
Documentation for a class Sphere . . . 19
tests.Test
Documentation for a class Test . . . 20
texture.Texture
Documentation for a class Texture . . . 22
triangle.Triangle
Documentation for a class Triangle . . . 23
vector.Vec2
Documentation for a class Vec2 . . . 25
vector.Vec3
Documentation for a class Vec3 . . . 26

Chapter 3

Class Documentation

3.1 ray.Hit Class Reference

Documentation for a class Hit.

Public Member Functions

- def __init__ (self, point, distance, color, primitive)

  Constructor.

Public Attributes

- **point**
- **distance**
- **color**
- **primitive**

3.1.1 Detailed Description

Documentation for a class Hit.

The documentation for this class was generated from the following file:

- ray.py

3.2 light_intensity.LightIntensity Class Reference

Documentation for a class LightIntensity.

Public Member Functions

- def **__init__** (self, color=[0, 0, 0])
- def **__add__** (self, other)
- def **__truediv__** (self, other)

Static Public Member Functions

- def **clamp01** (value)
- def **clamp_0_255** (value)
- def **remap_0_255** (value)
- def **clamp_color** (color)

Public Attributes

- **color**

3.2.1 Detailed Description

Documentation for a class LightIntensity.

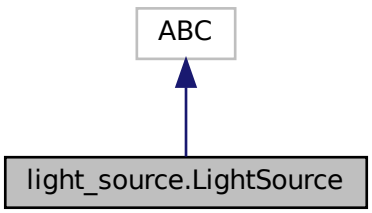
The documentation for this class was generated from the following file:

- light_intensity.py

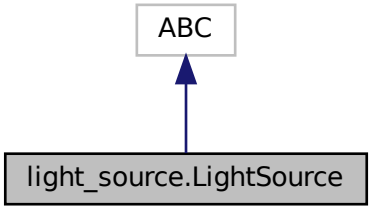
3.3 light_source.LightSource Class Reference

Documentation for a class LightSource.

Inheritance diagram for light_source.LightSource:



Collaboration diagram for light_source.LightSource:



Public Member Functions

- def __init__ (self, position=Vec3(0, 0, 0), color=[1, 1, 1], intensity=1)
  The constructor.
- def __str__ (self)
  Function returning object values in string format.

Public Attributes

- color
  Colour of light source.
- position
  Position of light source.
- intensity
  Intensity of light.

3.3.1 Detailed Description

Documentation for a class LightSource.

3.3.2 Constructor & Destructor Documentation

3.3.2.1 __init__()

```
def light_source.LightSource.__init__ (
              self,
              position = Vec3(0, 0, 0),
              color = [1, 1, 1],
              intensity = 1 )
```

The constructor.

Creates a LightSource with a specified Colour at a given Location.

The documentation for this class was generated from the following file:

- light_source.py

3.4 material.Material Class Reference

Documentation for a class Material.

Public Member Functions

- def __init__ (self, ambientColour=(1, 1, 1), diffuseColour=(1, 1, 1), reflectColour=(1, 1, 1), specularColour=(0, 0, 0), specularExponent=1, mirror_reflection_coefficient=1, diffuse_reflection_coefficient=1, texture=None)

  The constructor.
- def __str__ (self)

  Function returning object values in string format.

Public Attributes

- ambientColour

  Colour of Material under white ambient light.
- diffuseColour

  Colour of Material under direct white light.
- reflectColour

  Colour of reflected rays under direct white light.
- specularColour

  Colour of Material's specular highlights.
- specularExponent

  'Hardness' of Material's specular hightlights - high values give small, sharp highlights.
- **mirror_reflection_coefficient**
- **diffuse_reflection_coefficient**
- **texture**

3.4.1 Detailed Description

Documentation for a class Material.

3.4.2 Member Data Documentation

3.4.2.1 ambientColour

`material.Material.ambientColour`

Colour of Material under white ambient light.

Usually, but not always, the same as diffuseColour.

3.4.2.2 diffuseColour

`material.Material.diffuseColour`

Colour of Material under direct white light.

Usually, but not always, the same as ambientColour.

3.4.2.3 reflectColour

`material.Material.reflectColour`

Colour of reflected rays under direct white light.

If this is zero then there are no reflections.

3.4.2.4 specularColour

`material.Material.specularColour`

Colour of Material's specular highlights.

If this is zero then there are no highlights.

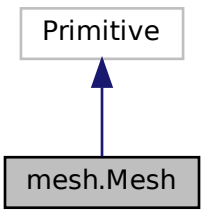
The documentation for this class was generated from the following file:

- material.py

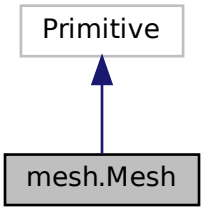
3.5 mesh.Mesh Class Reference

Documentation for a class Mesh.

Inheritance diagram for mesh.Mesh:



Collaboration diagram for mesh.Mesh:



Public Member Functions

- def __init__ (self, obj_file, position=Vec3(), material=None)

  Constructor.
- def get_detailed_intersections (self, ray)

  Checks if ray intersects with mesh and returns list of hits.
- def get_detailed_intersection (self, ray)

  Checks if ray intersects with mesh and returns hit closest to ray origin.
- def get_intersection (self, ray)

  Function returning intersection point.
- def **get_normal** (self, point)
- def **get_texture_color** (self, coords)

Public Attributes

- **triangles**

3.5.1 Detailed Description

Documentation for a class Mesh.

The documentation for this class was generated from the following file:

- mesh.py

3.6 image.MyImage Class Reference

Documentation for a class MyImage.

Public Member Functions

- def __init__ (self, width=500, height=500)

  The constructor.
- def len (self)

  Function returning image length.
- def clear_color (self, rgb_color)

  Function setting background color.
- def fancy_background (self)

  Function setting background color.
- def set_pixel (self, i, j, value)

  Function changing pixel color.
- def get_pixel_color (self, i, j)

  Function getting pixel color.
- def save_image (self)

  Function saving image to png format.

Public Attributes

- **width**
- **height**
- **image_matrix**

3.6.1 Detailed Description

Documentation for a class MyImage.

The documentation for this class was generated from the following file:

- image.py

3.7 orthogonal_camera.OrthogonalCamera Class Reference

Class for othogonal camera.

Public Member Functions

- def __init__ (self, position=Vec3(0, 0, 0), view_direction=Vec3(0, 0, 1), width=512, height=512, pixel_↩ size=(0.01, 0.01))

  Constructor.
- def render_scene (self, primitives)

  Function rendering the scene.

Public Attributes

- position
  Position of the camera.
- view_direction
  Direction camera is facing.
- w
  Width in pixels.
- h
  Height in pixels.
- wh_ratio
  Width-height raio.
- hw_ratio
  Height-width ratio.
- arRay
  Array of rays.
- x_angle
  Angle between view direction vector and X axis.
- y_angle
  Angle between view direction vector and Y axis.
- z_angle
  Angle between view direction vector and Z axis.

3.7.1 Detailed Description

Class for othogonal camera.

The documentation for this class was generated from the following file:

- orthogonal_camera.py

3.8 perspective_camera.PerspectiveCamera Class Reference

Class for othogonal camera.

Public Member Functions

- def __init__ (self, position=Vec3(0, 0, 0), view_direction=Vec3(0, 0, 1), width=512, height=512, near=.1, far=1000, fov=60)
  Constructor.
- def render_scene (self, primitives, light_sources, antialiasing=True)
  Function rendering the scene.

Static Public Member Functions

- def adaptive_antialiasing (ray, A, B, C, D, E, depth, max_depth, horizontal, vertical, background_color, primitives, lights)
  Function calculating color of pixel using adaptive antialiasing.

Public Attributes

- position

  Position of the camera.
- view_direction

  Direction camera is facing.
- width

  Width in pixels.
- height

  Height in pixels.
- near

  Near clipping plane.
- far

  Far clipping plane.
- fov

  Field of View.
- up

  Vector direction aligned with the "up" direction of camera.

3.8.1 Detailed Description

Class for othogonal camera.

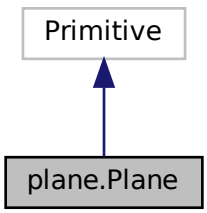
The documentation for this class was generated from the following file:

- perspective_camera.py

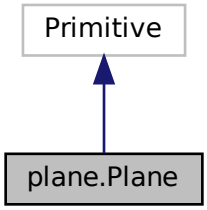
3.9 plane.Plane Class Reference

Documentation for a class Plane.

Inheritance diagram for plane.Plane:



Collaboration diagram for plane.Plane:



Public Member Functions

- def __init__ (self, normal_vector, d, color=[1, 0, 1], material=None)

  Constructor.
- def __str__ (self)

  Function returning object values in string format.
- def get_detailed_intersections (self, ray)

  Wrapper.
- def get_detailed_intersection (self, ray)

  Returns tuple with multiple data: point (None if no intersection), distance to point, color.
- def get_intersection (self, ray)

  Checks if plane and ray intersect witch each other and returns intersection point if they do, otherwise None.
- def **get_normal** (self, point)

Public Attributes

- normal_vector

  Vector perpendicular to plane.
- a

  Represents A in 'Ax + By + Cz D = 0' equation.
- b

  Represents B in 'Ax + By + Cz D = 0' equation.
- c

  Represents C in 'Ax + By + Cz D = 0' equation.
- d

  Represents D in 'Ax + By + Cz D = 0' equation.

3.9.1 Detailed Description

Documentation for a class Plane.

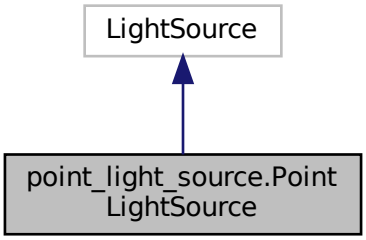
The documentation for this class was generated from the following file:

- plane.py

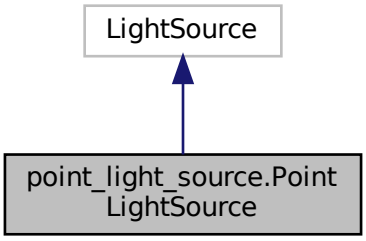
3.10 point_light_source.PointLightSource Class Reference

Documentation for a class PointLightSource.

Inheritance diagram for point_light_source.PointLightSource:



Collaboration diagram for point_light_source.PointLightSource:



Public Member Functions

- def **__init__** (self, position=[0, 0, 0], color=[1, 1, 1], intensity=1)

3.10.1 Detailed Description

Documentation for a class PointLightSource.

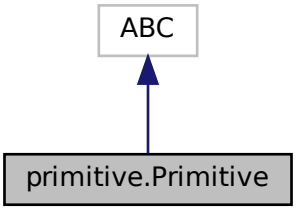
Light emitted from a Point.

The documentation for this class was generated from the following file:

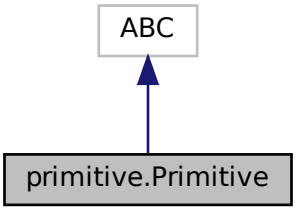
- point_light_source.py

3.11 primitive.Primitive Class Reference

Inheritance diagram for primitive.Primitive:



Collaboration diagram for primitive.Primitive:



Public Member Functions

- def **__init__** (self, color, material=None)
- def **get_intersection** (self, ray)
- def **get_detailed_intersection** (self, ray)
- def **get_detailed_intersections** (self, ray)
- def **get_normal** (self, point)
- def **get_texture_color** (self)

Public Attributes

- **color**
- **material**

The documentation for this class was generated from the following file:

- primitive.py

3.12 ray.Ray Class Reference

Documentation for a class Ray.

Public Member Functions

- def __init__ (self, origin=Vec3(0, 0, 0), direction=None, target=None, length=math.inf)

  Constructor.
- def __str__ (self)

  Function returning object values in string format.
- def is_point_on_ray (self, point)

  Check if point is on ray, returns true if yes, false otherwise.
- def set_direction (self, new_direction)

  Sets new direction vector and converts it to normalized vector.
- def set_target (self, new_target)

  Sets new target and updates direction vector.
- def get_plane_intersection (self, plane)

  Plane.get_intersection(ray) wrapper.
- def get_sphere_intersection (self, sphere)

  Sphere.get_intersection(ray) wrapper.
- def get_sphere_intersections (self, sphere)

  Sphere.get_ray_intersections(ray) wrapper.
- def get_pixel_hit (self, primitives)

  Iterates through list of primitives and returns hit.
- def **get_pixel_color** (self, primitives, lights)
- def **check_intersection** (self, primitives)

Public Attributes

- origin

  Origin vector of a given ray.
- direction

  Direction vector of a given ray.
- target

  Target point of a given ray.
- length

  Length of a given ray.

3.12.1 Detailed Description

Documentation for a class Ray.

3.12.2 Member Data Documentation

3.12.2.1 direction

`ray.Ray.direction`

Direction vector of a given ray.

Cannot be (0, 0, 0).

3.12.2.2 length

`ray.Ray.length`

Length of a given ray.

Default = Infinity

3.12.2.3 origin

`ray.Ray.origin`

Origin vector of a given ray.

Default = (0, 0, 0)

3.12.2.4 target

`ray.Ray.target`

Target point of a given ray.

Cannot be the same as origin.

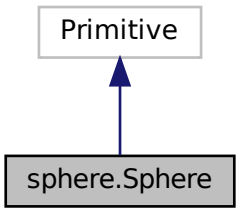
The documentation for this class was generated from the following file:

- ray.py

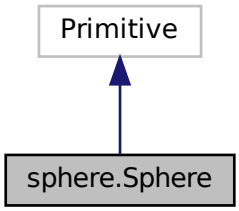
3.13 sphere.Sphere Class Reference

Documentation for a class Sphere.

Inheritance diagram for sphere.Sphere:



Collaboration diagram for sphere.Sphere:



Public Member Functions

- def __init__ (self, centre=Vec3(0, 0, 0), radius=1, color=[1, 0, 1], material=None)

  Constructor.
- def change_radius (self, new_radius)

  Sets radius and recalculates area and volume.
- def __str__ (self)

  Function returning object values in string format.
- def get_detailed_intersections (self, ray)

  Checks if ray intersects with sphere and returns list of hits.
- def get_detailed_intersection (self, ray)

  Function returning hit.
- def get_intersection (self, ray)

  Checks if ray intersects with sphere and returns point closest to ray origin.
- def get_normal (self, point)

  Gets normal for given point.
- def get_texture_color (self, coords)

  Gets pixel color from material texture.

Public Attributes

- centre
  Centre of the sphere.
- **color**
- radius
  Radius of the sphere.
- area
  Area of the sphere.
- volume
  Volume of the sphere.

3.13.1 Detailed Description

Documentation for a class Sphere.

3.13.2 Member Function Documentation

3.13.2.1 get_texture_color()

```
def sphere.Sphere.get_texture_color (
              self,
              coords )
```

Gets pixel color from material texture.

If texture or material is None than return privmitive color

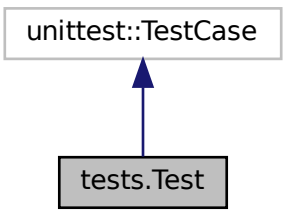
The documentation for this class was generated from the following file:

- sphere.py

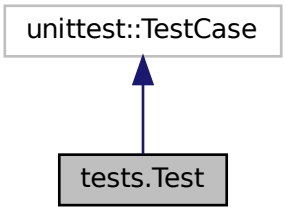
3.14 tests.Test Class Reference

Documentation for a class Test.

Inheritance diagram for tests.Test:



Collaboration diagram for tests.Test:



Public Member Functions

- def **setUp** (self)
- def test_add (self)

  Vector tests.
- def test_sub (self)

  Vector tests.
- def test_pos (self)

  Vector tests.
- def test_neg (self)

  Vector tests.
- def test_length (self)

  Vector tests.
- def test___truediv__ (self)

  Vector tests.
- def test_mul (self)

  Vector tests.
- def test_cross (self)

  Vector tests.
- def test_point_on_line (self)

  Ray tests.
- def test_plane_intersection (self)

  Plane tests.
- def test_get_centre (self)

  Sphere tests.
- def test_get_radius (self)

  Sphere tests.
- def test_surface_area (self)

  Sphere tests.
- def test_get_volume (self)

  Sphere tests.
- def test_get_sphere_intersection (self)

  Sphere tests.
- def test_clamp_0_255 (self)

  Light intensity tests.

Public Attributes

- **v1**
- **v2**
- **v3**
- **v4**
- **v5**
- **v6**
- **v7**
- **r1**
- **r2**
- **r3**
- **r4**
- **p1**
- **p2**
- **s1**
- **s2**
- **li1**
- **li2**
- **li3**

3.14.1 Detailed Description

Documentation for a class Test.

Unit tests.

The documentation for this class was generated from the following file:

- tests.py

3.15 texture.Texture Class Reference

Documentation for a class Texture.

Public Member Functions

- None __init__ (self, file_name)

  Constructor.
- def rectangular_mapping (self, coords)

  Returns pixel color for primitive and ray intersection represented by coords for rectangulars.
- def spherical_mapping (self, coords, r)

  Returns pixel color for primitive and ray intersection represented by coords for spheres.
- str **__str__** (self)

Public Attributes

- **file_name**
- **img**
- **height**
- **width**

3.15.1 Detailed Description

Documentation for a class Texture.

3.15.2 Member Function Documentation

3.15.2.1 spherical_mapping()

```
def texture.Texture.spherical_mapping (
              self,
              coords,
              r )
```

Returns pixel color for primitive and ray intersection represented by coords for spheres.

Needs a spehere radius to scale properly

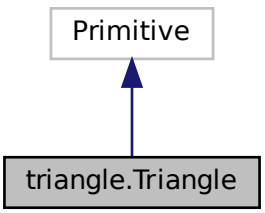
The documentation for this class was generated from the following file:

- texture.py

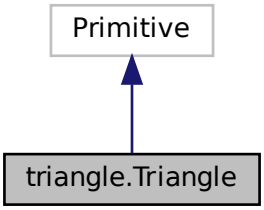
3.16 triangle.Triangle Class Reference

Documentation for a class Triangle.

Inheritance diagram for triangle.Triangle:



Collaboration diagram for triangle.Triangle:



Public Member Functions

- def __init__ (self, v1=Vec3(0, 0, 0), v2=Vec3(0, 0, 0), v3=Vec3(0, 0, 0), color=[1, 0, 1], material=None)

  Constructor.
- def __str__ (self)

  Function returning object values in string format.
- def get_detailed_intersections (self, ray)

  Checks if ray intersects with triangle and returns hit in form of a list.
- def get_detailed_intersection (self, ray)

  Checks if ray intercescts with triangle and returns hit.
- def get_intersection (self, ray)

  Checks if ray intersects with triangle and return intersection point.
- def **get_normal** (self, point)
- def **get_texture_color** (self, coords)

Public Attributes

- v1

  Triangle vertex.
- v2

  Triangle vertex.
- v3

  Triangle vertex.
- color

  Color of triangle.
- normal_vector

  Vector perpendicular to plane.

3.16.1 Detailed Description

Documentation for a class Triangle.

The documentation for this class was generated from the following file:

- triangle.py

3.17 vector.Vec2 Class Reference

Documentation for a class Vec2.

Public Member Functions

- def __init__ (self, x, y)

  The constructor.
- def __add__ (self, other)

  Function returning sum of two vectors or sum of a vector and a scalar.
- def __iadd__ (self, other)

  Function returning sum (In-place addition) of two vectors or sum of a vector and a scalar.
- def __sub__ (self, other)

  Function returning difference of two vectors or difference of a vector and a scalar.
- def __isub__ (self, other)

  Function returning difference (In-place Subtraction) of two vectors or difference of a vector and a scalar.
- def __eq__ (self, other)

  Function "equal".
- def __abs__ (self)

  Function returning absolute value of a given vector.
- def __ne__ (self, other)

  Function "not equal".
- def __neg__ (self)

  Function negating vector coordinates.
- def __pos__ (self)

  Function for positive vector coordinates.
- def __str__ (self)

  Function returning object values in string format.
- def length (self)

  Function returning vector length.
- def distance (self, other)

  Function returning the length of the displacement vector (distance between two points).
- def __truediv__ (self, other)

  Function returning quotient of two vectors or quotient of a vector and a scalar.
- def __itruediv__ (self, other)

  Function returning quotient (In-place Division) of two vectors or quotient of a vector and a scalar.
- def __mul__ (self, other)

  Function returning dot product of two vectors or dot product of a vector and a scalar.
- def __imul__ (self, other)

  Function returning dot product (In-place multiplication) of two vectors or dot product of a vector and a scalar.
- def __rmul__ (self, other)

  Function returning cross product of two vectors.

Public Attributes

- x

  A class variable.
- y

  A class variable.

3.17.1 Detailed Description

Documentation for a class Vec2.

3.17.2 Member Data Documentation

3.17.2.1 x

`vector.Vec2.x`

A class variable.

Coordinate x of a given vector.

3.17.2.2 y

`vector.Vec2.y`

A class variable.

Coordinate y of a given vector.

The documentation for this class was generated from the following file:

- vector.py

3.18 vector.Vec3 Class Reference

Documentation for a class Vec3.

Public Member Functions

- def __init__ (self, x=0, y=0, z=0)

  The constructor.
- def **x** (self)
- def **x** (self, inp)
- def **x** (self)
- def **y** (self)
- def **y** (self, inp)
- def **y** (self)
- def **z** (self)
- def **z** (self, inp)
- def **z** (self)
- def __add__ (self, other)

  Function returning sum of two vectors or sum of a vector and a scalar.
- def __iadd__ (self, other)

  Function returning sum (In-place addition) of two vectors or sum of a vector and a scalar.
- def __sub__ (self, other)

  Function returning difference of two vectors or difference of a vector and a scalar.
- def __isub__ (self, other)

  Function returning difference (In-place Subtraction) of two vectors or difference of a vector and a scalar.
- def __eq__ (self, other)

  Function "equal".
- def __abs__ (self)

  Function returning absolute value of a given vector.
- def __ne__ (self, other)

  Function "not equal".
- def __neg__ (self)

  Function negating vector coordinates.
- def __pos__ (self)

  Function for positive vector coordinates.
- def __str__ (self)

  Function returning object values in string format.
- def length (self)

  Function returning vector length.
- def distance (self, other)

  Function returning the length of the displacement vector (distance between two points).
- def is_point_on_ray (self, ray)

  Is point on ray wrapper.
- def **normalized** (self)
- def __truediv__ (self, other)

  Function returning quotient of two vectors or quotient of a vector and a scalar.
- def __itruediv__ (self, other)

  Function returning quotient (In-place Division) of two vectors or quotient of a vector and a scalar.
- def __mul__ (self, other)

  Function returning dot product of two vectors or dot product of a vector and a scalar.
- def __imul__ (self, other)

  Function returning dot product (In-place multiplication) of two vectors or dot product of a vector and a scalar.
- def __rmul__ (self, other)

  Function returning dot product (Reverse multiplication).
- def cross (self, other)

  Function returning cross product of two vectors.

Public Attributes

- x
  A class variable.
- y
  A class variable.
- z
  A class variable.

Static Public Attributes

- def r = x
  Alias
- def g = y
  Alias.
- def b = z
  Alias.

3.18.1 Detailed Description

Documentation for a class Vec3.

3.18.2 Member Data Documentation

3.18.2.1 x

`vector.Vec3.x`

A class variable.

Coordinate x of a given vector.

3.18.2.2 y

`vector.Vec3.y`

A class variable.

Coordinate y of a given vector.

3.18.2.3 z

`vector.Vec3.z`

A class variable.

Coordinate z of a given vector.

The documentation for this class was generated from the following file:

- vector.py

Index

__init__
    light_source.LightSource, 7

ambientColour
    material.Material, 8

diffuseColour
    material.Material, 8
direction
    ray.Ray, 17

get_texture_color
    sphere.Sphere, 20

image.MyImage, 10

length
    ray.Ray, 18
light_intensity.LightIntensity, 5
light_source.LightSource, 6
    __init__, 7

material.Material, 8
    ambientColour, 8
    diffuseColour, 8
    reflectColour, 9
    specularColour, 9
mesh.Mesh, 9

origin
    ray.Ray, 18
orthogonal_camera.OrthogonalCamera, 11

perspective_camera.PerspectiveCamera, 12
plane.Plane, 13
point_light_source.PointLightSource, 15
primitive.Primitive, 16

ray.Hit, 5
ray.Ray, 17
    direction, 17
    length, 18
    origin, 18
    target, 18
reflectColour
    material.Material, 9

specularColour
    material.Material, 9
sphere.Sphere, 19
    get_texture_color, 20
spherical_mapping
    texture.Texture, 23

target
    ray.Ray, 18
tests.Test, 20
texture.Texture, 22
    spherical_mapping, 23
triangle.Triangle, 23

vector.Vec2, 25
    x, 26
    y, 26
vector.Vec3, 26
    x, 28
    y, 28
    z, 28

x
    vector.Vec2, 26
    vector.Vec3, 28

y
    vector.Vec2, 26
    vector.Vec3, 28

z
    vector.Vec3, 28