

UsingVehicleClassComplete 4.5.1

De Wiki

Aller à : [navigation](#), [rechercher](#)

[UsingVehicleClassComplete 4.5.1](#)

```
public class UsingVehicleClassComplete {

    private enum TYPE { EMPTY, COMPLETE };

    public static void main(String[] args) throws PatriusException {

        // Dry mass
        final double dryMass = 1000.;
        final MassProperty dryMassProperty = new MassProperty(dryMass);

        // Shape
        final double lref = 1.0;
        final Sphere sphere = new Sphere(lref);
        //final RightParallelepiped solarPanels = new RightParallelepiped(1.,
0.1, 0.1); // To be replaced by addSolarPanels ?
        //VehicleSurfaceModel vehicleRefSurfaceWithSolarpanels = new
VehicleSurfaceModel(sphere, solarPanels);
        VehicleSurfaceModel vehicleRefSurfacewithoutSolarPanels = new
VehicleSurfaceModel(sphere, null);
        VehicleSurfaceModel vehicleRefSurface =
vehicleRefSurfacewithoutSolarPanels;

        // Aerodynamic properties
        final double cd = 2.;
        final double cl = 2.;
        final AerodynamicProperties aerodynamicProperties = new
AerodynamicProperties(vehicleRefSurface, cd, cl);

        // Radiative properties
        final double ka = 1.0;
        final double ks = 0.0;
        final double kd = 0.0;
        final double kaIr = 1.0;
        final double ksIr = 0.0;
        final double kdIr = 0.0;
        final RadiativeProperty radiativeProperty = new RadiativeProperty(ka,
ks, kd);
        final RadiativeIRProperty radiativeIRProperty =
            new RadiativeIRProperty(kaIr, ksIr, kdIr);
        final RadiativeProperties radiativeProperties =
            new RadiativeProperties(radiativeProperty,
radiativeIRProperty, vehicleRefSurface);

        // Propulsive properties
```

```

// Tanks
final double merg1 = 100.;
final TankProperty tank1 = new TankProperty(merg1);
tank1.setPartName("TANK1");
final double merg2 = 200.;
final TankProperty tank2 = new TankProperty(merg2);
tank2.setPartName("TANK2");
final ArrayList<TankProperty> tanksList = new
ArrayList<TankProperty>();
tanksList.add(tank1);
tanksList.add(tank2);

// Engine
final double thrust = 400.;
final double isp = 320.;
final PropulsiveProperty engine = new PropulsiveProperty(thrust,
isp);
engine.setPartName("PROP");
final ArrayList<PropulsiveProperty> enginesList = new
ArrayList<PropulsiveProperty>();
enginesList.add(engine);

for ( TYPE type : TYPE.values() ) {

    Vehicle veh = null;

    if ( type == TYPE.EMPTY ) {

        // Case with an initial empty constructor
        System.out.println("\nCASE OF EMPTY CONSTRUCTOR");
        veh = new Vehicle();
        veh.setDryMass(dryMass);
        veh.setMainShape(sphere);
        veh.addSolarPanel(Vector3D.PLUS_I, 1.);
        veh.setAerodynamicsProperties(cd, cl);
        veh.setRadiativeProperties(ka, ks, kd, kaIr, ksIr, kdIr);
        veh.addTank(tank1.getPartName(), tank1);
        veh.addTank(tank2.getPartName(), tank2);
        veh.addEngine(engine.getPartName(), engine);

    } else {

        // Case with a complete constructor
        System.out.println("\nCASE OF COMPLETE CONSTRUCTOR");
        veh = new Vehicle(vehicleRefSurface, null, dryMassProperty,
aerodynamicProperties, radiativeProperties, enginesList, tanksList);
        //veh.setMainShape(sphere);

    }

}

```

```

        // Getting the corresponding assembly
        final Assembly assembly =
veh.createAssembly(FramesFactory.getCIRF());

        // Getting the corresponding mass model (useful for propagation,
maneuvres, ...)
        final MassProvider mm = new MassModel(assembly);

        // Getting main characteristics
        System.out.println("\nMAIN PROPERTIES");
        System.out.println("Name of the main part: " +
assembly.getMainPart().getName());
        System.out.println("Total mass: " + mm.getTotalMass());

        // Getting propulsive characteristics
        System.out.println("\nPROPULSIVE PROPERTIES");
        for (int i = 0; i < veh.getEnginesList().size(); i++) {

System.out.println(veh.getEnginesList().get(i).getPartName());
                System.out.println("Thrust =
"+veh.getEnginesList().get(i).getThrust(null)+" N");
                System.out.println("Isp =
"+veh.getEnginesList().get(i).getIsp(null)+" s");
        }
        for (int i = 0; i < veh.getTanksList().size(); i++) {
                System.out.println(veh.getTanksList().get(i).getPartName()+":
"+veh.getTanksList().get(i).getMass()+" kg");
        }

        // Getting aerodynamics characteristics
        System.out.println("\nAERODYNAMIC PROPERTIES");
        final Vector3D xDir = new Vector3D(1., 0., 0.);
        final Vector3D yDir = new Vector3D(0., 1., 0.);
        final Vector3D zDir = new Vector3D(0., 0., 1.);
        System.out.println("SX =
"+veh.getMainShape().getCrossSection(xDir)+" m2");
        System.out.println("SX =
"+veh.getAerodynamicProperties().getVehicleSurfaceModel().getMainPartShape().
getCrossSection(xDir)+" m2");
        System.out.println("SY =
"+veh.getAerodynamicProperties().getVehicleSurfaceModel().getMainPartShape().
getCrossSection(yDir)+" m2");
        System.out.println("SZ =
"+veh.getAerodynamicProperties().getVehicleSurfaceModel().getMainPartShape().
getCrossSection(zDir)+" m2");
        System.out.println("CD =
"+veh.getAerodynamicProperties().getConstantDragCoef());

        System.out.println("\nRADIATIVE PROPERTIES");
        System.out.println("SX =
"+veh.getRadiativeProperties().getVehicleSurfaceModel().getMainPartShape().

```

```

getCrossSection(xDir)+" m2");
        System.out.println("SY =
"+veh.getRadiativeProperties().getVehicleSurfaceModel().getMainPartShape().
getCrossSection(yDir)+" m2");
        System.out.println("SZ =
"+veh.getRadiativeProperties().getVehicleSurfaceModel().getMainPartShape().
getCrossSection(zDir)+" m2");
        System.out.println("KA =
"+veh.getRadiativeProperties().getRadiativeProperty().getAbsorptionRatio().
getValue());
        System.out.println("KD =
"+veh.getRadiativeProperties().getRadiativeProperty().
getDiffuseReflectionRatio().getValue());
        System.out.println("KS =
"+veh.getRadiativeProperties().getRadiativeProperty().
getSpecularReflectionRatio().getValue());
        System.out.println("KAI =
"+veh.getRadiativeProperties().getRadiativeIRProperty().getAbsorptionCoef().
getValue());
        System.out.println("KDI =
"+veh.getRadiativeProperties().getRadiativeIRProperty().
getDiffuseReflectionCoef().getValue());
        System.out.println("KSI =
"+veh.getRadiativeProperties().getRadiativeIRProperty().
getSpecularReflectionCoef().getValue());

    }

}

}

```

Récupérée de

« http://patrius.cnes.fr/index.php?title=UsingVehicleClassComplete_4.5.1&oldid=2719 »

Menu de navigation

Outils personnels

- [18.118.119.129](#)
- [Discussion avec cette adresse IP](#)
- [Créer un compte](#)
- [Se connecter](#)

Espaces de noms

- [Page](#)
- [Discussion](#)

Variantes

Affichages

- [Lire](#)
- [Voir le texte source](#)
- [Historique](#)
- [Exporter en PDF](#)

Plus

Rechercher

PATRIUS

- [Welcome](#)

Evolutions

- [Main differences between V4.15 and V4.14](#)
- [Main differences between V4.14 and V4.13](#)
- [Main differences between V4.13 and V4.12](#)
- [Main differences between V4.12 and V4.11](#)
- [Main differences between V4.11 and V4.10](#)
- [Main differences between V4.10 and V4.9](#)
- [Main differences between V4.9 and V4.8](#)
- [Main differences between V4.8 and V4.7](#)
- [Main differences between V4.7 and V4.6.1](#)
- [Main differences between V4.6.1 and V4.5.1](#)
- [Main differences between V4.5.1 and V4.4](#)
- [Main differences between V4.4 and V4.3](#)
- [Main differences between V4.3 and V4.2](#)
- [Main differences between V4.2 and V4.1.1](#)
- [Main differences between V4.1.1 and V4.1](#)
- [Main differences between V4.1 and V4.0](#)
- [Main differences between V4.0 and V3.4.1](#)

User Manual

- [User Manual 4.15](#)

- [User Manual 4.14](#)
- [User Manual 4.13](#)
- [User Manual 4.12](#)
- [User Manual 4.11](#)
- [User Manual 4.10](#)
- [User Manual 4.9](#)
- [User Manual 4.8](#)
- [User Manual 4.7](#)
- [User Manual 4.6.1](#)
- [User Manual 4.5.1](#)
- [User Manual 4.4](#)
- [User Manual 4.3](#)
- [User Manual 4.2](#)
- [User Manual 4.1](#)
- [User Manual 4.0](#)
- [User Manual 3.4.1](#)
- [User Manual 3.3](#)

Tutorials

- [Tutorials 4.15](#)
- [Tutorials 4.14](#)
- [Tutorials 4.13.5](#)
- [Tutorials 4.12.1](#)
- [Tutorials 4.8.1](#)
- [Tutorials 4.5.1](#)
- [Tutorials 4.4](#)
- [Tutorials 4.1](#)
- [Tutorials 4.0](#)

Links

- [CNES freeware server](#)

Navigation

- [Accueil](#)
- [Modifications récentes](#)
- [Page au hasard](#)
- [Aide](#)

Outils

- [Pages liées](#)
- [Suivi des pages liées](#)
- [Pages spéciales](#)
- [Adresse de cette version](#)

- [Information sur la page](#)
- [Citer cette page](#)

- Dernière modification de cette page le 17 août 2020 à 09:17.

- [Politique de confidentialité](#)
- [À propos de Wiki](#)
- [Avertissements](#)

- 