

main (+ setup) + readme + globs

BlueOrigin

PACKAGES	<p>pretraitement</p> <p>: filtrage et mise au propre des données</p> <p>--> CSV_new</p>	<p>analyse_donnees</p> <p>: calculs et résultats à partir des données traitées</p>	<p>affichage</p> <p>: affiche les résultats (plan de vol, trajectoires...)</p> <p>--> plot, print</p>	<p>conditions_entrees</p> <p>: contient des fonctions/classes à configurer par l'utilisateur et à mettre en entrées de propriétés (profil vent, masse payload)</p>	<p>interface_utilisateur</p>
CLASSES	<p>FiltrageDonnees()</p> <p>: quelle entités récupérer</p> <p>def __init__()</p> <p>ConversionTemps()</p> <p>: data synchro avec le même nombre et même valeur de temps</p> <p>def __init__()</p> <p>CreationNewFichier()</p> <p>def __init__()</p>	<p>PhysiqueVol()</p> <p>g = 9,81</p> <p>isp =</p> <p>masse_viderocket =</p> <p>def __init__(masse_payload,)</p> <p>def calcul_vitesse()</p> <p>def calcul_masse()</p> <p>def calcul_poussee()</p> <p>def calcul_altitude()</p> <p>def calcul_centre_gravite()</p> <p>Trajectoire(profil_vent = 0, masse_payload = 0)</p> <p>g = 9,81</p> <p>isp =</p> <p>masse_viderocket =</p> <p>def __init__()</p> <p>PlanVol(profil_vent = 0, masse_payload = 0)</p> <p>g = 9,81</p> <p>isp =</p> <p>masse_viderocket =</p> <p>def __init__()</p> <p>ConeLimite()</p> <p>: definition du cone (altitude/rayon à deploy break)</p> <p>g = 9,81</p> <p>isp =</p> <p>masse_viderocket =</p>	<p>Affichage()</p> <p>def __init__()</p> <p>def affichage_trajectoire()</p> <p>def affichage_plan_vol()</p> <p>def affichage_physique()</p>	<p>ProfilVent ()</p> <p>def __init__(</p> <p>vitesse_min =18</p> <p>Vitesse_max =</p> <p>hauteur_min =</p> <p>hauteur_max =</p> <p>Rugosite =)</p> <p>def calcul_vent()</p> <div>$v_2 = v_1 \frac{\ln \left(\frac{h_2}{z_0} \right)}{\ln \left(\frac{h_1}{z_0} \right)}$</div>	<p>Profil_vent_jour1 =</p> <p>profil_vent(vitesse_min = 12...)</p> <p>Affichage_traj_ini =</p> <p>Affichage_new_traj =</p>

```
def __init__()
```

```
GoNogo()
```

```
def __init__()
```