



Utiliser l'API Python pour automatiser les mises en données et analyse avec THERCAST®

Vous souhaitez accroître votre productivité ? Ayez dès à présent connaissance des outils mis à votre disposition pour effectuer de manière automatique les étapes de mises en données et d'analyse de vos résultats.

Le temps que vous consacrez à créer vos projets de simulations et à analyser les résultats de vos calculs est généralement très conséquent. Les opérations que vous effectuez sont souvent redondantes et parfois très chronophages.

Les scripts Python vont vous permettre de créer des projets, de lancer des calculs et d'analyser des résultats avec une automatisation maximale. Typiquement, vous pourrez créer votre procédé personnalisé, gérer vos objets, importer et générer des maillages, définir tous types de paramètres, générer automatiquement des variantes de calculs, afficher uniquement

les résultats dont vous avez besoin dans la configuration optimale, exporter vos résultats et bien d'autres choses encore. Cette nouvelle fonctionnalité offre de nombreux avantages: gain de temps, automatisation, sécurisation des projets, interconnexion avec vos autres outils numériques.

Que vous souhaitiez automatiser tout ou partie de vos opérations, définir des données constantes ou de manière dynamique ou encore appeler une application tierce depuis THERCAST®, tout est possible et imaginable. Cette formation est faite pour vous !

NIVEAU

Intermédiaire

PRÉREQUIS

Disposer d'une première expérience des logiciels TRANSVALOR. Vous devez connaître l'utilisation de l'interface NxT.

Disposer d'une première expérience en codage avec le langage PYTHON

OBJECTIFS

- Découvrir ce que l'API Python peut vous apporter en automatisation
- Tirer profit des nouveautés de l'interface pour accélérer la mise en données et l'analyse des résultats

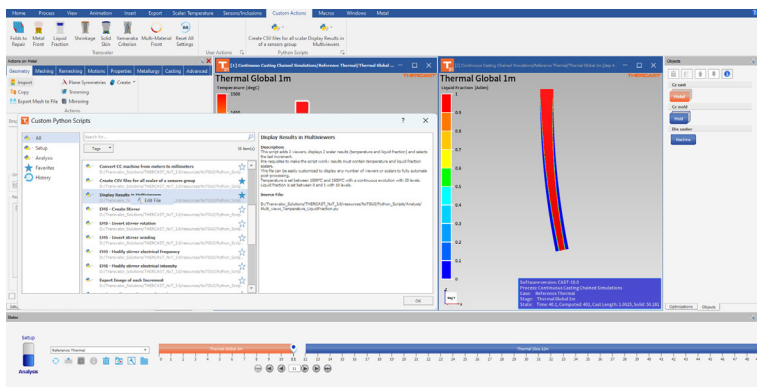
FORMATION	DURÉE	PRIX HT	PARTICIPANTS
Intra-entreprise	2 jours	2800 €/formation	1 à 3 personnes

JOUR 1 > 08h30 - 12h00 et 13h30 - 17h00

Introduction	- Présentation de Transvalor - Objectifs de la formation
Pourquoi cette API ?	- Contexte - Outils précédents mis à disposition - Pré requis - Limitations actuelles - Perspectives
Structure des scripts	- Fonctionnement de la console Python - Vocabulaire (notion de classes, fonctions et arguments) - Liens entre les différents objets, simulations, attributs, propriétés
Scripts de mise en données	- Comprendre les scripts existants - Travailler sur un script de mise en données complète d'étape - Coder son propre script de mise en données
Scripts d'analyse	- Comprendre les scripts existants - Comment les adapter à vos besoins ? - Coder son propre script d'analyse des résultats
Documentation	- Explication de la documentation mise à disposition pour coder vos propres scripts de mise en données et d'analyse - Python Help

JOUR 2 > 08h30 - 12h00 et 13h30 - 17h00

Mise en pratique sur automatisation de la mise en données	- Définition de la problématique et des étapes à automatiser - Réalisation du script d'automatisation
Mise en pratique sur analyse des résultats	- Description des étapes d'analyse - Réalisation des scripts d'automatisation
Perspectives	- Quelles possibilités pour aller plus loin et automatiser complètement la mise en données et l'analyse ? - Paramètres variables, interfaces personnalisées, exécution en lignes de commande
Conclusion	Questions diverses et évaluation de la formation



```

D:\Transvalor\Solution\THERCAST_Nxt_3.0\resources\NxtGUI\Python_Scripts\Analysis\Multi_views_Temperature_LiquidFraction.p...
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Thermocouple_1.txt | Transvalor_Parametric | Example_of_Analysis | Multi_views_Temperature_LiquidFraction.py
1 # *****
2 #
3 # * Copyright (C) Transvalor S.A., France. All rights reserved.
4 #
5 # *****
6 # Sample Python script that adds 2 viewers and displays 2 different results for the last increment
7 # Pre-requisites to make the script work: Results must contain temperature and liquid fraction sca
8 # temperature is set between 1000 and 1500 °C with a continuous evolution with 20 levels.
9 # liquid fraction is set between 0 and 1 with 10 levels.
10
11
12 from ts.vpy.ts.vanalysis import *
13
14 def multi_views(viewer, scalarname, rangemin, rangemax, level, value = ColorScheme.NORMAL, usec
15 states = viewer.states()
16 lastinc = states.count()-1
17 states.setCurrentIndex(lastinc)
18 scalars = viewer.scalarResults()
19 scalar = scalars.getByName(scalarname)
20 scalars.setFringes(scalar)
21 scalar_settings = scalar.settings()
22 scalar_settings.setAutoRangeMode(AutoRangeMode.OFF)
23 scalar_settings.setRange(rangemin, rangemax)
24 scalar_settings.setLegendLevel(level)
25 scalar_settings.setColorScheme(value)
26 scalar_settings.setUseContinuousLegendColoring(usecontinuous)
27
28 return scalar_settings
  
```