

Pymensura, simpla modulo de Pitono (Python) por sistemo Mensurasoft

Komplementeco de Pitono kaj peliloj de Mensurasoft

Pitono estas programad-lingvo tre uzata por lernado ; ĝi estas interpreta lingvo, tre fleksebla, kun ĉiuj datum-strukturoj kaj program-strukturoj modernaj. Ĝi ebligas fari nur per malmultaj linioj programon sufiĉe potencon, kaj dank'al ĝia legebleco, tiuj programoj estas (iom) facile flegeblaj kaj plibonigeblaj.

La peliloj por mezur-aparatoj de la sistemo Mensurasoft estas malgrandaj dosieroj, kiuj havas plenumoblajn komandojn ; ili estas dinamikaj bibliotekoj (.dll en Windows, .so en Linux). Ĉiu dosiero havas la komandojn por fari mezurojn el iu tipo de mezuriloj. Ili devige estas faritaj per tradukebla program-lingvo (C/C++, Pascal, iuj Basic), ke ne ĉiam estas facile fareblaj.

La kunigo de tiuj du programadoj (per tradukebla program-lingvo kaj per Pitono) ebligas uzi la potencon de programoj en Pitono, sed sen la malfaciloj de programado por iu specifa aparato.

La modulo Pymensura (pymensura.py) estas metenda en dosierujo kiun povas uzi Pitono. Oni povas meti ĝin :

- ĉu en la kuranta dosierujo, kie estas la ĉefa programo en Pitono kaj la peliloj
- ĉu en la dosierujo /Lib/site-packages por Windows (ekzemple c:\Python33\Lib\site-packages)

Enhavo de la modulo pymensura

Ĝi havas nur la klason mensura, kiu estas por unu mezurilo. Kiam oni deziras uzi la pelilon por tiu aparato, oni kreu dinamikan ekzempleron (unu objekto) de tiu klaso per la komando :

```
mia_ilo=mensura("bibdyn_syst_pb_ANSI.dll")  
(se bibdyn_syst_pb_ANSI.dll estas la pelilo)
```

Poste, oni povas rekte uzi la metodojn kaj datumajn atribuojn de tiu objekto.

`codage` : estas la kodigo de la literoj (defaŭlte "windows-1252", sed oni povas uzi "utf-8", aŭ aliaj.

`correct` : indikas ĉu la pelilo estas bone ŝargita ; por tio, necesas unue ke la dosiero demandita ekzistus, ke ĝi estus dinamika biblioteko, kaj fine ke ĝi enhavu la funkcion `cdetail` (la kontrolo de aliaj funkcios ne estas farata dum tio).

La sekvantaj metodoj estas uzeblaj. Se la kreado de la objekto ne estas bone farita, la funkcioj kiuj donas ĉenojn donas malplenan ĉenon, krom `detail()` kaj `titre()`, kiuj donas mesaĝeton de eraro, kaj la pornombraj funkcioj, kiuj donas valoron -999.

`titre()` : mallonga titolo, kun la nomo de la mezurilo.

`detail()` : ĉeno pli longa ol `titre`, kun ofte la nomo de la aŭtoro, kaj la dato de verkado de tiu pelilo.

`ead(n)` estas la n-a analoga enigo (komenco en nulo), kaj `nead(n)` la nomo de tiu analoga enigo.

`sad(n, valoro)` donas valoro-n al la n-a analoga eligo (komenco en nulo), kaj `nsad(n)` estas la nomo de tiu analoga eligo.

`eb(n)` estas la n-a duuma enigo (komenco en nulo), kaj `neb(n)` estas la nomo de tiu duuma enigo.

`sb(n, valoro)` donas valoron al la n-a analoga eligo (komenco en nulo), kaj `nsb(n)` estas la nomo de tiu analoga eligo.

`stop()` haltigas la uzon de tiu aparato, kaj tiu aparato estas ekde nun uzebla de alia programo.

Ekzemplo de programo por uzi tiujn funkciojn

Ni supozu ke la nomo de la pelilo estas « `bibdyn_syst_pb_ANSI.dll` ».

Tiu programo ŝarĝas la pelilon, kaj montras la titolon (`titre()`), la detalon (`detail()`), la analogan enigon 0, la nomon de tiu analoga enigo 0, la duuman enigon 0, la nomon de tiu duuma enigo 0, fiksas la analogan eligon 0 en la valoro 1, montras la nomon de tiu analoga eligo 0, fiksas la duuman eligon 0 en la valoro 1 (= « vero »), donas la nomon de la duuma eligo 0, kaj fine haltigas la uzon de la pelilo (`stop()`).

```
from pymensura import *
mia_ilo=mensura("bibdyn_syst_pb_ANSI.dll")
print(mia_ilo.titre())
print(mia_ilo.detail())
print(mia_ilo.ead(0))
print(mia_ilo.nead(0))
print(mia_ilo.eb(0))
print(mia_ilo.neb(0))
print(mia_ilo.sad(0,1))
print(mia_ilo.nsad(0))
print(mia_ilo.sb(0,1))
print(mia_ilo.nsb(0))
mia_ilo.stop()
```

Por kiuj mezuriloj ?

1) Por ĉiuj kies peliloj jam estas faritaj, kaj ili estas multnombraj. En la retejo <http://sciencexp.free.fr>, oni povas trovi multajn pelilojn :

- ĝeneralaj aparatoj por perkomputilaj eksperimentoj : Jeulin, Orphy, Pierron, Eurosmart, ExpEyes...
- specifaj mezur-aparatoj : pHmetroj, konduktimetroj, pesiloj, multimetroj, kolorimetroj, spektrofotometroj, termometroj, lum-mezuriloj...
- elektronikaĵoj por hobiistoj : Arduino, MicroHope, Velleman...
- hejm-konstruaĵoj por diversaj konektiloj (lud-konektilo, laŭseria konektilo, paralela konektilo, k.a.).

2) Kaj ankaŭ por ĉiuj kies vi povas fari pelilon per kutimaj program-lingvoj (Delphi, FreePascal, C ou C++, PureBasic aŭ FreeBasic...). Legu la informojn kaj fonto-programoj en <http://sciencexp.free.fr>