

Programozás Alapjai 1 NHF felhasználói dokumentáció

Péter Bertalan Zoltán

2017. december 4.

drunt

1 Alapvető kezelés

Legegyszerűbben a program a parancssorból indítható. Sajnos, úgy tűnik, Windows-ra nem sikerül lefordítani, ezért ott egyelőre nem elérhető — ez a GTK miatt van.

Tehát indítsuk el a programot az alábbi két lehetőség közül valamelyikkel:

```
./drunt -i  
./drunt -g
```

Az első interaktív szöveges módba, a második GUI módba dob minket. Fontos tudni, hogy a GUI mód csak események megtekintésére használható jelenleg. Nem is igényel további leírást, a parancssori mód azonban igen, ezt az alábbiakban írom le.

2 Interaktív mód

Indításkor rögtön közli a program, hogy betölti az alapértelmezett naptárfájlunkat, ami a `calendar.ics` fájl a program mappájában. Amennyiben ez nem sikerül, a program tájékoztat minket a további teendőkről (manuálisan kell betölteni valamit, különben kvázi csak játszunk a programmal, semmi sem végleges). A program felkínálja a `help` parancsot, amit mindeképpen érdemes megnézni, de most itt is leírom, mi mit csinál.

2.1 help

Mint mondtam, ez a segítség, a puska. Ha csak egyszerűen beírjuk és enter-t ütünk, kiadja a lehetséges parancsok listáját. Ha egy parancs nevét írjuk utána, például `help help`, akkor egy másik puskát kapunk, az adott parancshoz tartozót. Érdemes sokat használni, hiszen gyorsabb mint ezt a könyvet olvasgatni.

2.2 exit

Kilép a naptárunkból. Ekkor és csakis ekkor mentődnek módosításaink a fájlba (kivéve azt az esetet, amikor újabb fájlt töltünk be...). Tehát amíg fut a program, a fájl változatlan marad. Ha nem szeretnénk, hogy bármilyen módosítás érvényre jusson, adjuk meg a `-d` opciót: ekkor biztonsági okokból visszakérdez a program, biztos kilépünk-e. Ha igen, akkor a program leáll és a módítások elvesznek. Ha mégis a nem-et választjuk, visszakerülünk a parancssorba.

2.3 load

Ha valamiért szeretnénk egy egyéni fájlt betölteni és azt kezelni, akkor ezzel a paraccsal megtehetjük. A `drunt` csak `.ics` kiterjesztésű (formátumú) fájlokat tud kezelni. Egyszerűen üssük be: `load file/elereis/ut/enFajlom.ics` és üssünk enter-t. Ekkor a program megkísérli betölteni a fájlt. Ilyenkor előtte elmenti az aktuálisan szerkesztettet az ahhoz tartozó fájlba és csak aztán tölti be az újat. Azt is csinálhatjuk, hogy csak annyit írunk: `load`: a program megkérdezi, hol a fájl.

2.4 create

Ez a legbonyolultabb parancs. Na persze ez sem olyan bonyolult. Események létrehozását teszi lehetővé. Alapvetően két lehetőségünk van: Vagy elindítjuk a varázslót és úgy adjuk meg az esemény paramétereit, vagy egysoros parancsban tesszük meg ugyanezt (haladóknak!). A varázsló indításához csak írjuk be: **create**. Ekkor a következő sorrendben érkeznek majd a kérdések

1. Kezdő dátum
2. Befejező dátum
3. Rövid leírás
4. Hely
5. Hosszabb leírás
6. Prioritás

Ezek közül egyébként csak az első három fontos, a többi nem lényeges kitöltenünk, de persze érdemes kihasználni a lehetőségeket. Fontos, hogy a dátumokat idővel együtt kell érteni, a következőképpen:

YYYY MM DD HH MM

Ahol az első négy karakter az év (így, négy karakterrel, nem csak az utolsó kettővel!), a második kettő a hónap, majd a nap, óra, perc. A prioritás értéke 0 és 9 között mozog, ahol a 0 jelenti a legfontosabbat, a 9 a legkevésbé fontosat.

Ha mégsem varázslózunk, akkor nézzük meg a **help create**-t. Ott látjuk az opciókat, hogy hogyan lehet egy parancssal elintézni egy egész eseményt. Két fontos irányelv: Ilyenkor a dátum-időt a következőképpen kell beütni: YYYYMMDDTHHMM00Z. Minden ugyanaz, mint fenn, de nincsen szóköz, illetve kötelező egy T-betűt tenni a dátum és idő közé. Kötelező a végére a két 0 és a Z betű is. Ha ezeket rosszul írjuk, problémákhoz vezethet. A másik irányelv: bármilyen szöveges információt (szóval igazából minden mást) aposztrófok közé írjunk. Ez nagyon fontos.

2.5 modify

Ezzel a parancssal módosítható egy már létező esemény. Listázás után érdemes használni, ha már tudjuk, hogy mit akarunk módosítani. Azt kell tudnunk, hogy mikor van az esemény. Ha tudjuk, akkor csak írjuk be így:

modify YYYY MM DD

és nyomjunk entert. Ekkor a program megnézi, hogy az adott napon van-e több esemény is. Ha nincs egy sem, akkor nem történik semmi. Ha csak egy van, akkor kiírja az eseményt és megkérdezi, valóban módosítani akarja-e. Ha több van, akkor kilistázza őket és egy számmal megadható, melyiket akarjuk módosítani. Ha megvan, egy menüből ki kell választani, hogy melyik paramétert szeretnénk módosítani. Csak írjuk be a számot, a varázsló vezet. Ezután vagy azt mondjuk a programnak (rákérdez), hogy kész vagyunk, vagy átírunk valami mást is. Ha kész, akkor még egy utolsó biztonsági kérdés után a program felülírja az eseményt.

2.6 delete

Óvatosan! Ezzel eseményeket törölhetünk. Ugyanúgy kell használni teljesen, mint a fenti parancsot, csak itt nem fogunk menüt kapni. Ha ki van választva az esemény, egy elfogadás után törölhetjük. Az esemény ugyan nem hozható vissza, de ha fent megismert `-d` opciót használjuk kilépésnél, nem fognak mentésre kerülni a változások, tehát tulajdonképpen ép marad az esemény.

2.7 list

Talán a leghasznosabb, leggyakoribb parancs. 4 féle variációja van:

2.7.1 list year

Egy egész évet kilistáz. Vagy adunk neki egy argumentumot így:

```
list year -d N YYYY
```

ahol N egy tetszőleges szélességet jelölő szám 1 és 4 között, vagy csak így futtatjuk:

```
list year YYYY
```

ahol (és feljebb is persze) YYYY a listázni kívánt év, négy számjeggyel. Ez az egész arra jó, hogy lássuk, milyen hónapok, hogyan alakulnak az évben

2.7.2 list month

Egy fokkal hasznosabb, ha egy adott hónapot nézhetünk meg, de úgy, hogy lássuk, vannak-e események a napokon. Csak üssük be a következőt:

```
list month YYYY MM
```

ahol a betűk korábbról ismertek. Egy táblázatot kapunk. A kis jelölés a nap alatt jelzi, hogy ott esemény van. Ha kíváncsiak vagyunk az esemény részleteire, jól jöhet az alábbi parancs

2.7.3 list day

Egy napot listáz. Azaz az aznapi eseményeket. Ha nincs semmi, csak egy üres fejlécut kapunk. Így kell:

```
list day YYYY MM DD
```

2.7.4 list agenda

Ennek adhatunk argumentumot, vagy nem adunk. Ha beütünk utána egy számot, akkor az annyiszor egy napot fog kiírni a program, azaz annyi nap eseményeit. Alapértelmezés szerint, ha nem adunk meg számot, akkor a következő 7 napot írja ki.

3 find

Keresésre ad lehetőséget. Viszonylag egyszerű a szintaxis: csak írjuk be a parancsot és utána a keresendőt: ha csak egy szó, aposztrófok nélkül, de ha több, akkor aposztróffal kell indítani és zárni is. Ezután a program kiírja azokat az eseményeket, amik megfelelnek az elvárásnak. Sajnos annyiszor ír ki egy eseményt, ahányszor találatként szerepelt (bár ennek pár sor lenne javítása), így egy esemény többször is megjelenhet.