

# Tkinter manager

## Uživatelská dokumentace

### Požadavky

Aplikace je napsaná v pythonu 3, je spustitelná na windows a linuxu.

Seznam závislostí:

- tkinter
- datetime
- json
- platform
- os
- (math)

Všechny jsou součástí python standard library. Některé linuxové distribuce mohou postrádat tkinter a je třeba ho doinstalovat.

### Spuštění

Ke spuštění aplikace spusťte soubor program.py. Tento soubor z původní složky nepřesouvejte. Pokud chcete mít možnost spustit aplikaci z jiného umístění (např. z plochy), spusťte soubor create\_alias.py, jenž vytvoří script, jenž aplikaci spouští přímo. Ten můžete přesunout na libovolné, vámi určené místo.

### O aplikaci

Tkinter manager je jednoduchá aplikace určená k uchovávání přehledu o nadcházejících úkolech a událostech. Pokud chcete přidat připomínku, klikněte na tlačítko New v levém horním rohu. To vytvoří novou událost, která je propojená s kalendářem, zobrazeným vpravo. Změněním deadlinu nebo urgency se změní i zvýraznění odpovídajícího dne v kalendáři. Kliknutím na událost k ní můžete přidat nebo změnit poznámky. Změny se automaticky uloží při zavření aplikace. Můžete je také uložit manuálně zmáčknutím Ctrl+S. V případě většího množství událostí se mezi nimi lze orientovat pomocí seřazovacích možností vpravo nahoře.

V pravém dolním rohu se také nachází příkazová řádka pro překladač pythonu. Její možnosti jsou však poněkud omezené, jejím primárním zamýšleným účelem je použití jako kalkulačka.

## Programátorská dokumentace

### Struktura

Hlavní část aplikace tvoří instance tříd Register (vlevo) a Calendar (vpravo). Register si drží seznam všech událostí (třída Task) ve svém atributu `register.tasks`, tlačítko New jednu přidá, smazání události ji odebere.

Funkce `on_closing`, která je volaná při zavření aplikace, převede hlavní atributy všech instancí třídy Task do obvyčejného seznamu (json si neporadí s custom třídami) a seznam těchto seznamů je následně pomocí modulu json uložen do souboru `tasks.json`. Při opětovném spuštění jsou instance za pomoci tohoto seznamu znovu vytvořeny a aplikace se obnoví do stavu, ve kterém byla před zavřením.

Komunikace mezi Registerem a Calendarem probíhá oběma směry; Calendar využívá metodu třídy Register `update()`, která při změně zobrazených měsíců v kalendáři zvýrazní na základě seznamu událostí všechny odpovídající dny a Register má naopak přístup ke globálnímu seznamu Buttons, díky čemuž může sám zvýrazňovat jednotlivé dny, když se změní některá z událostí.

### Python CLI

Pod kalendářem je také umístěná příkazová řádka, která je řešená pomocí funkce `eval()`. Přestože nemůže importovat žádné externí moduly, je u této funkce vyžadovaná určitá obezřetnost a pokud bude aplikaci používat někdo nedůvěryhodný, může být záhodno tento widget zakázat nebo úplně odstranit.

V základě má přístup k modulu `math` a lze jím provádět matematické operace. Jeho možnosti jdou dále rozšířit importem dalších modulů do souboru `modules/cli.py`.