Activity-Tracker

Develop an activity-tracker, that can monitor your process and help you visualize the time you spent on different apps

Code Tree

```
📁 activity_tracker/
├── 📁 core/
│   ├── 📄 __init__.py
│   │
│   ├── 📄 process_tracker.py    # 进程追踪功能
│   ├── 📁 windows/
│   │   ├── 📄 __init__.py
│   │   └── 📄 windows[core](core)_tracker.py
│   ├── 📁 macos/
│   │   ├── 📄 __init__.py
│   │   └── 📄 macos_tracker.py
│   ├── 📁 linux/[core](core)
│   │   ├── 📄 __init__.py
│   │   └── 📄 linux_tracker.py
│   │
│   ├── 📄 data_analysis.py    # 分析数据
│   └── 📄 database.py         # 数据库操作
├── 📄 config.py              # 配置文件
├── 📄 test.py                # run a small test
└── 📄 tracker.py             # 主程[core](core)序


To be added:
tray_app.py
```

process_tracker.py

responsible to track all the process info

- ☑ foreground_windows : to be completed in subclassed
- ☑ background_process

Subclasses:

1. **windows_tracker**: implement this first
2. macos_tracker
3. linux_tracker

database.py

responsible to store all the infomation from `tracker` in database

- ☐ init_database
- ☐ save_activity

data_analysis.py

Analyze all the collected data , for some apps might now run in the foreground, like some audio players.

test.py & main.py

run a simple instance on test.py

Developing Plan

1. Implement the `tracker`, *Windows* part first, then try mac/linux
2. Handle the data collected from `tracker`,
    1. init database
    2. analyze data in the base
3. Get the analysis **visualized**

> process_tracker -> windows_tracker -> database -> test.py -> visualize.py