Getting Started with Knowledge Engineer

Installation

Create project directory named 'snake'

```
(venv) jerry@desktop:~$ knowledge-engineer --create=snake
[19:15:53.671]create_new_proc::Create new process snake
[19:15:53.676]create_new_proc::Created ExampleProcess in snake.
[19:15:53.677]create_new_proc::cd snake
[19:15:53.677]create_new_proc::Edit the ke_process_config.env, and insert your OPENAI_API_KEY.
(venv) jerry@desktop:~$
```

Finish setup of snake project

- `cd snake`
- `edit ke_process_config.env`
- Enter your Open API Key in the line: `OPENAI_API_KEY='<Your Open API Key>'`

```
(venv) jerry@desktop:~$ cd snake
(venv) jerry@desktop:~/snake$ gedit ke_process_config.env
(venv) jerry@desktop:~/snake$
```

Quick peek at the Example project

```
(venv) jerry@desktop:~/snake$ tree ../snake
../snake
├── ke_process_config.env
├── Prompts
│   ├── 1- Make Generate Code Prompt.kepf
│   └── 2- Make Snake Game.kepf
└── Requirements
    └── ApplicationDescription.md

2 directories, 4 files
(venv) jerry@desktop:~/snake$
```

As you can see, the project consists of:

- the snake directory
- 2 subdirectories
- and 4 files.

The Configuration file ke_process_config.env

In the snake directory there is a config file called ke_process_config.env.

```
(venv) jerry@desktop:~/snake$ more ke_process_config.env
KE_PROC_DIR_PROMPTS='Prompts'
KE_PROC_DIR_LOGS='Logs'
OPENAI_API_KEY='<Your Open API Key>'

(venv) jerry@desktop:~/snake$
```

This config tells Knowledge Engineer 4 things:

1. It identifies the directory as a Knowledge Engineer Project.
2. It tells Knowledge Engineer that the Prompts defining the process steps are in the subdirectory 'Prompts/'
3. It tells Knowledge Engineer to place logs in the subdirectory 'Logs/'
4. And lastly it contains your OpenAI APi Key

The file Requirements/ApplicationDescription.md

This directory is arbitrarily named. This file is arbitrarily named. But the file is referenced in the first step of the process. In this example this file is used to describe the program to be generated. Changing the contents of this file will change what the LLM attempts to generate.

```
You are writing a python version of the snake game using pygame.

# The Game
```

```
Implement the standard "snake" game using arrow keys and 'q' to quit.
All rules and interface are to be as generally expected.

#### Architecture:
python 3, with pygame

#### The game_board
game_board size is 480x480.
use different colors for snake and foods

#### Game Play:
The game is implemented as steps 10 per second.
```

"Prompts/*" The Steps in the Process

We can list the steps in the process by executing: `knowledge_engineer --list`

```
(venv) jerry@desktop:~/snake$ knowledge-engineer --list
[20:04:21.772]          ke::List of all Steps in Process snake:
    1- Make Generate Code Prompt
    2- Make Snake Game
(venv) jerry@desktop:~/snake$
```

> Note that the steps are ordered alphabetically.
> This ordering is for listing but also for execution!

> When you communicate with LLM's you carry on Conversations. (Here called **Steps**).
> Conversations consist of **Interactions** (Instances of messages sent back and forth to the LLM).

Let's take a peek at the steps in the process:

Prompts/1- Make Generate Code Prompt.kepf

```
.llm "llm_name": "OpenAI", "model": "gpt-3.5-turbo-0125", "max_tokens": 50000
```

The **.llm** line identifies the LLM to be called and allows for its configuration.

```
    .clear "Code/*", "Planning/*", "Logs/*"
```

the **.clear** line tells Knowledge Engineer to delete matching files before executing the Step

```
.system
You are a Knowledge Engineer creating a GPT4 Turbo prompt.
The prompt will instruct GPT3.5 to create a Python 3 Application
Do not explain yourself.
Do not apologize.
Complete the tasks given in a way that is optimized for Chat GPTs easy comprehension while not leaving anything out.
Your answers will be in MarkDown
```

the **.system** line tells Knowledge Engineer to build a system message to be sent to the LLM

```
.user
read the description of the program: Requirements/ApplicationDescription.md
create a prompt that will instruct GPT 3.5 Turbo to:
 - create a complete running program as described.
 - contain initialize, process, and terminate phases
 - implement all the requirements in the prompts
Write the prompt to 'Planning/Gen_Code_Prompt.md' using function 'write_file'.
```

the **.user** line tells Knowledge Engineer to build a user message to be sent to the LLM

```
.exec
```

The **.exec** line tells Knowledge Engineer to start an **Iteration** with the LLM

Okay, lets run this Step

```
(venv) jerry@desktop:~/snake$ knowledge-engineer --step=1*
[20:25:35.655]          ke::Found ['Prompts/1- Make Generate Code Prompt.kepf']
[20:25:35.664]          ke::Clearing ['Code/*', 'Planning/*', 'Logs/*']
[20:25:35.665]         Step::┌ Step: snake:Prompts/1- Make Generate Code Prompt
```

```
[20:25:35.666]          Step::| Model: gpt-3.5-turbo-0125, Temperature: 0, Max Tokens: 50,000
[20:25:35.741]          AI::| ┌──────────────────────────────────────────────────────────────
[20:25:35.742]          AI::| │ system message ['You are a Knowledge Engineer creating a GPT4 Turbo prompt.\nThe prompt will
instruct G…
[20:25:35.743]          AI::| │   user message ["read the description of the program:
Requirements/ApplicationDescription.md\ncreate a …
[20:25:36.566]          AI::| │       read_file (Requirements/ApplicationDescription.md)
[20:25:36.569]          AI::| │            rtn ['You are writing a python version of the snake game using pygame.\n\n# The
Game\n\nImpl…
[20:25:39.832]          AI::| │      write_file (Planning/Gen_Code_Prompt.md, ...)
['{"name":"Planning/Gen_Code_Prompt.md","contents":"Cr…
[20:25:39.838]          AI::| │            rtn ['Done.']
[20:25:40.950]          AI::| │     AI message ["The prompt to instruct GPT 3.5 Turbo to create a Python program implementing
the snake…
[20:25:40.953]          AI::| └──────────────────────────────────────────────────────────────
[20:25:40.955]          Step::| Writing log Logs/1- Make Generate Code Prompt log.md
[20:25:40.957]          Step::| Elapsed: 0m 5.29s Token Usage: Total: 1,725 (Prompt: 1,450, Completion: 275)
[20:25:40.957]                │ Costs:: Total: $0.00 (Prompt: $0.0007, Completion: $0.0004)
[20:25:40.957]                └──────────────────────────────────────────────────────────────
```

Let us break it down

```
    (venv) jerry@desktop:~/snake$ knowledge-engineer --step=1*
```

You will note that I used the '*' in the step name because I did not want to type out the entire step name.

```
[20:25:35.655]           ke::Found ['Prompts/1- Make Generate Code Prompt.kepf']
```

KE writes the step found for execution

```
[20:25:35.664]           ke::Clearing ['Code/*', 'Planning/*', 'Logs/*']
```

KE deletes files matching the given globs

```
[20:25:35.665]          Step::┌ Step: snake:Prompts/1- Make Generate Code Prompt
[20:25:35.666]          Step::| Model: gpt-3.5-turbo-0125, Temperature: 0, Max Tokens: 50,000
```

KE Initializes the step to communicate with LLM

```
[20:25:35.741]          AI::| ┌──────────────────────────────────────────────────────────────
[20:25:35.742]          AI::| │ system message ['You are a Knowledge Engineer creating a GPT4 Turbo prompt.\nThe prompt will
instruct G…
[20:25:35.743]          AI::| │   user message ["read the description of the program:
Requirements/ApplicationDescription.md\ncreate a …
[20:25:36.566]          AI::| │       read_file (Requirements/ApplicationDescription.md)
[20:25:36.569]          AI::| │            rtn ['You are writing a python version of the snake game using pygame.\n\n# The
Game\n\nImpl…
[20:25:39.832]          AI::| │      write_file (Planning/Gen_Code_Prompt.md, ...)
['{"name":"Planning/Gen_Code_Prompt.md","contents":"Cr…
[20:25:39.838]          AI::| │            rtn ['Done.']
[20:25:40.950]          AI::| │     AI message ["The prompt to instruct GPT 3.5 Turbo to create a Python program implementing
the snake…
[20:25:40.953]          AI::| └──────────────────────────────────────────────────────────────
```

KE Iterates with the LLM (AI). You will see:

1. KE sent system message sent to the AI
2. KE sent user message sent to the AI
3. AI requested to read_file "Requirements/ApplicationDescription.md"
4. KE sent rtn with the contents of the file to AI
5. 3 seconds later AI asked KE to write_file(Planning/Gen_Code_Prompt.md, …)
6. KE responded with rtn 'Done.'
7. AI Terminated the Iteration with: AI message ["The prompt to instruct GPT 3.5 Turbo to create a Python program
   implementing the snake…

```
[20:25:40.955]          Step::| Writing log Logs/1- Make Generate Code Prompt log.md
[20:25:40.957]          Step::| Elapsed: 0m 5.29s Token Usage: Total: 1,725 (Prompt: 1,450, Completion: 275)
[20:25:40.957]                │ Costs:: Total: $0.00 (Prompt: $0.0007, Completion: $0.0004)
[20:25:40.957]                └──────────────────────────────────────────────────────────────
```

KE closes the Step printing stats

Prompts/2- Make Snake Game.kepf

```
.llm "llm_name": "OpenAI", "model": "gpt-4-0125-preview", "max_tokens": 50000
.system
You are an IT Engineer, programming a Python 3 Application
Do not explain yourself.
```

```
Do not apologize.
Check all code for completeness, correctness, and make sure it is executable.
write all python code via the 'write_file' function to the 'Code/' directory.

.user
.include Planning/Gen_Code_Prompt.md
write a complete executable program to 'Code/snake.py' with the 'write_file' function

.exec
```

> **Note that:**
> The LLM being used is **GPT4** not **3.5** as the previous step.
> The **.include** line which is used to insert the prompt generated by the first step into this prompt

Running the entire process

Instead of running the second step, lets run the entire process!

```
(venv) jerry@desktop:~/snake$ knowledge-engineer --execute
[22:14:40.143]        ke::Begin Execution of Process snake
[22:14:40.148]        ke::Execute snake(1): 1- Make Generate Code Prompt.kepf
[22:14:40.153]        ke::Clearing ['Code/*', 'Planning/*', 'Logs/*']
[22:14:40.154]     Step::┌ Step: snake:Prompts/1- Make Generate Code Prompt
[22:14:40.155]     Step::│ Model: gpt-3.5-turbo-0125, Temperature: 0, Max Tokens: 50,000
[22:14:40.229]       AI::│ ┌
[22:14:40.230]       AI::│ │ system message ['You are a Knowledge Engineer creating a GPT4 Turbo prompt.\nThe prompt will
instruct G…
[22:14:40.231]       AI::│ │   user message ["read the description of the program:
Requirements/ApplicationDescription.md\ncreate a …
[22:14:40.882]       AI::│ │      read_file (Requirements/ApplicationDescription.md)
[22:14:40.885]       AI::│ │          rtn ['You are writing a python version of the snake game using pygame.\n\n# The
Game\n\nImpl…
[22:14:43.997]       AI::│ │      write_file (Planning/Gen_Code_Prompt.md, ...)
['{"name":"Planning/Gen_Code_Prompt.md","contents":"Cr…
[22:14:44.003]       AI::│ │          rtn ['Done.']
[22:14:45.298]       AI::│ │      AI message ["The prompt to instruct GPT 3.5 Turbo to create the Python program for the
snake game u…
[22:14:45.301]       AI::│ └
[22:14:45.304]     Step::│ Writing log Logs/1- Make Generate Code Prompt log.md
[22:14:45.306]     Step::│ Elapsed: 0m 5.15s Token Usage: Total: 1,735 (Prompt: 1,449, Completion: 286)
[22:14:45.306]          │ Costs:: Total: $0.00 (Prompt: $0.0007, Completion: $0.0004)
[22:14:45.306]          └
[22:14:45.311]        ke::Execute snake(2): 2- Make Snake Game.kepf
[22:14:45.324]     Step::┌ Step: snake:Prompts/2- Make Snake Game
[22:14:45.326]     Step::│ Model: gpt-4-0125-preview, Temperature: 0, Max Tokens: 50,000
[22:14:45.328]       AI::│ ┌
[22:14:45.330]       AI::│ │ system message ["You are an IT Engineer, programming a Python 3 Application\nDo not explain
yourself.\n…
[22:14:45.333]       AI::│ │   user message ["Create a Python program that implements the classic snake game using
Pygame.\n\n### Ph…
[22:15:29.589]       AI::│ │      write_file (Code/snake.py, ...)['{"name":"Code/snake.py","contents":"import pygame\\nimport
sys\\ni…
[22:15:29.599]       AI::│ │          rtn ['Done.']
[22:15:32.331]       AI::│ │      AI message ["The Python program for the classic snake game using Pygame has been created
and saved …
[22:15:32.334]       AI::│ └
[22:15:32.336]     Step::│ Writing log Logs/2- Make Snake Game log.md
[22:15:32.338]     Step::│ Elapsed: 0m 47.01s Token Usage: Total: 2,282 (Prompt: 1,569, Completion: 713)
[22:15:32.338]          │ Costs:: Total: $0.04 (Prompt: $0.0157, Completion: $0.0214)
[22:15:32.338]          └
[22:15:32.343]        ke::Elapsed: 0m 52.19s Token Usage: Total: 4,017.0 (Prompt: 3,018, Completion: 999)
[22:15:32.343]          Costs:: Total: $0.04 (Prompt: $0.0164, Completion: $0.0218)
(venv) jerry@desktop:~/snake$
```

Entire process took 52.19 seconds, and cost me 4 cents

But the result is not consistent enough for me