

[🏠](#) [■ Modules](#) [■ Chains](#) [■ Additional](#) [■ Program-aided language model \(PAL\) chain](#)

Program-aided language model (PAL) chain

Implements Program-Aided Language Models, as in <https://arxiv.org/pdf/2211.10435.pdf>.

```
from langchain.chains import PALChain
from langchain import OpenAI
```

```
llm = OpenAI(temperature=0, max_tokens=512)
```

Math Prompt

```
pal_chain = PALChain.from_math_prompt(llm, verbose=True)
```

```
question = "Jan has three times the number of pets as Marcia. Marcia has two more pets than Cindy. If Cindy has four pets, how many total pets do the three have?"
```

```
pal_chain.run(question)
```

```
> Entering new PALChain chain...
def solution():
    """Jan has three times the number of pets as Marcia. Marcia has two more pets than Cindy. If Cindy
has four pets, how many total pets do the three have?"""
    cindy_pets = 4
    marcia_pets = cindy_pets + 2
    jan_pets = marcia_pets * 3
    total_pets = cindy_pets + marcia_pets + jan_pets
    result = total_pets
    return result

> Finished chain.

'28'
```

Colored Objects

```
pal_chain = PALChain.from_colored_object_prompt(llm, verbose=True)
```

```
question = "On the desk, you see two blue booklets, two purple booklets, and two yellow pairs of sunglasses.
If I remove all the pairs of sunglasses from the desk, how many purple items remain on it?"
```

```
pal_chain.run(question)
```

```
> Entering new PALChain chain...
# Put objects into a list to record ordering
objects = []
objects += [('booklet', 'blue')] * 2
objects += [('booklet', 'purple')] * 2
objects += [('sunglasses', 'yellow')] * 2

# Remove all pairs of sunglasses
objects = [object for object in objects if object[0] != 'sunglasses']

# Count number of purple objects
num_purple = len([object for object in objects if object[1] == 'purple'])
answer = num_purple

> Finished PALChain chain.

'2'
```

Intermediate Steps

You can also use the intermediate steps flag to return the code executed that generates the answer.

```
pal_chain = PALChain.from_colored_object_prompt(  
    llm, verbose=True, return_intermediate_steps=True  
)
```

```
question = "On the desk, you see two blue booklets, two purple booklets, and two yellow pairs of sunglasses.  
If I remove all the pairs of sunglasses from the desk, how many purple items remain on it?"
```

```
result = pal_chain({"question": question})
```

```
> Entering new PALChain chain...  
# Put objects into a list to record ordering  
objects = []  
objects += [('booklet', 'blue')] * 2  
objects += [('booklet', 'purple')] * 2  
objects += [('sunglasses', 'yellow')] * 2  
  
# Remove all pairs of sunglasses  
objects = [object for object in objects if object[0] != 'sunglasses']  
  
# Count number of purple objects  
num_purple = len([object for object in objects if object[1] == 'purple'])  
answer = num_purple
```

```
> Finished chain.
```

```
result["intermediate_steps"]
```

```
"# Put objects into a list to record ordering\nobjects = []\nobjects += [('booklet', 'blue')] * 2\nobjects += [('booklet', 'purple')] * 2\nobjects += [('sunglasses', 'yellow')] * 2\n\n# Remove all pairs of sunglasses\nobjects = [object for object in objects if object[0] != 'sunglasses']\n\n# Count number of purple objects\nnum_purple = len([object for object in objects if object[1] == 'purple'])\nanswer = num_purple"
```