

```
cau_oss_python_03 — -zsh — 98x36
(base) ~/Documents/PycharmProjects/cau_oss_python_03(main)$ git branch -l
* main
(base) ~/Documents/PycharmProjects/cau_oss_python_03(main)$ git branch hw5
(base) ~/Documents/PycharmProjects/cau_oss_python_03(main)$ git branch -l
hw5
* main
(base) ~/Documents/PycharmProjects/cau_oss_python_03(main)$ git switch hw5
[Switched to branch 'hw5']
(base) ~/Documents/PycharmProjects/cau_oss_python_03(hw5)$ git status
On branch hw5
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   figure.py
        modified:   figure_main.py

no changes added to commit (use "git add" and/or "git commit -a")
(base) ~/Documents/PycharmProjects/cau_oss_python_03(hw5)$ git add figure.py figure_main.py
(base) ~/Documents/PycharmProjects/cau_oss_python_03(hw5)$ git commit -m "implement hw5"
[hw5 1108bc6] implement hw5
 2 files changed, 60 insertions(+), 40 deletions(-)
(base) ~/Documents/PycharmProjects/cau_oss_python_03(hw5)$ git branch -l
* hw5
  main
(base) ~/Documents/PycharmProjects/cau_oss_python_03(hw5)$ git switch main
fatal: invalid reference: main
(base) ~/Documents/PycharmProjects/cau_oss_python_03(hw5)$ git switch main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
(base) ~/Documents/PycharmProjects/cau_oss_python_03(main)$ git merge hw5
Updating 64700e0..1108bc6
Fast-forward
 figure.py | 69 ++++++-----
 figure_main.py | 31 ++++++-----
 2 files changed, 60 insertions(+), 40 deletions(-)
(base) ~/Documents/PycharmProjects/cau_oss_python_03(main)$
```

figure_main.py ×

figure.py

figure_main.py > ...

```
1  import figure
2
3  myline = figure.line(10, 20)
4  width, height = myline.get_length()
5  try:
6      rectangle = figure.area_rectangle(width, height)
7      print(rectangle)
8  except ValueError:
9      print("width and height should be positive number")
10
11 myline.set_length(20, 30)
12 width, height = myline.get_length()
13 try:
14     triangle = figure.area_right_triangle(width, height)
15     print(triangle)
16 except ValueError:
17     print("width and height should be positive number")
18
19 myline.set_length(30, 40)
20 width, height = myline.get_length()
21 try:
22     ellipse = figure.area_ellipse(width, height)
23     print(ellipse)
24 except ValueError:
25     print("width and height should be positive number")
```

figure_main.py

figure.py ×

figure.py > ...

```
1  '''
2  figure 모듈은 도형에 관련된 함수 및 클래스를 제공하는 모듈입니다.
3  line 클래스를 통해 선의 길이에 대해 수정 및 접근하여
4  직사각형, 타원, 직각삼각형의 넓이를 구할 수 있습니다.
5  '''
6
7  import math
8
9  class line:
10     '''
11     line 클래스는 다양한 선의 길이를 저장하는 클래스입니다.
12     외부에서 접근 불가능한 __width와 __height가 있습니다.
13     해당 변수들에 접근하고 수정하기 위해 set_length 메소드와 get_length 메소드를 제공합니다.
14     '''
15     __width = 0
16     __height = 0
17     def __init__(self, width, height):
18         ''' 생성자를 통해 초기 width와 height값을 설정합니다.
19         Args:
20             width (int or float): 초기 가로 길이
21             height (int or float): 초기 세로 길이
22         '''
23         self.__width = width
24         self.__height = height
25
26     def set_length(self, width, height):
27         ''' 선의 길이를 수정합니다.
28         Args:
29             width (int or float): 수정하고자 하는 가로 길이
30             height (int or float): 수정하고자 하는 세로 길이
31         '''
32         self.__width = width
33         self.__height = height
34
35     def get_length(self):
36         ''' 객체가 가지고 있는 선의 길이 값을 반환합니다.
37         Returns:
38             (int or float): 저장하고 있는 선의 길이
39         '''
40         return self.__width, self.__height
41
42     def area_rectangle(width, height):
43         ''' 선의 길이를 입력받아 직사각형의 넓이를 구하는 함수입니다.
44         Args:
45             width (int or float): 직사각형 가로의 길이
46             height (int or float): 직사각형 세로의 길이
47         Returns:
48             (int or float): 직사각형의 넓이
49         '''
50         if width <= 0 or height <=0: raise ValueError
51         return width * height
52
53     def area_ellipse(width, height):
54         ''' 선의 길이를 입력받아 타원의 넓이를 구하는 함수입니다.
55         Args:
56             width (int or float): 타원 장축의 길이
57             height (int or float): 타원 단축의 길이
58         Returns:
59             (int or float): 타원의 넓이
60         '''
61         if width <= 0 or height <=0: raise ValueError
62         return width * height * math.pi
63
64     def area_right_triangle(width, height):
65         ''' 선의 길이를 입력받아 직각삼각형의 넓이를 구하는 함수입니다.
66         Args:
67             width (int or float): 직각삼각형 밑변의 길이
68             height (int or float): 직각삼각형 높이의 길이
69         Returns:
70             (int or float): 직각삼각형의 넓이
71         '''
72         if width <= 0 or height <=0: raise ValueError
73         return (width * height) / 2
```