
lavalera97

Oct 24, 2021

1		3
2		5
3		7
4		13
Python Module Index		15
Index		17

•
•
•
•

1. :

•
•
•
•
•
•
•

2. :

•
•
•
•
•
•
•

•
•
•

CHAPTER
ONE

:

1. *main.py*
2. , ,
- 3.

**CHAPTER
TWO**

:

:

$$\bullet : S = r^2$$

$$\bullet : P = 2r$$

:

$$\bullet : S = a * 2$$

$$\bullet : P = 4 * a$$

:

$$\bullet : S = a * b$$

$$\bullet : P = 2(a + b)$$

:

• :

$$- : S = \frac{1}{2}a * h$$

$$- : S = \sqrt{p(p-a)(p-b)(p-c)}$$

$$\bullet : P = a + b + c$$

:

• :

$$- : S = \frac{a+b}{|a-b|} \sqrt{(p-a)(p-b)(p-a-c)(p-a-d)}$$

$$- : S = \frac{1}{2}(a+b) * h$$

$$\bullet : P = a + b + c + d$$

:

• :

$$- : S = a * h$$

$$- : S = \frac{1}{2}d_1 * d_2$$

$$\bullet : P = 4 * a$$

:

$$\bullet : S = 4R^2$$

$$\bullet : V = \frac{4}{3}R^3$$

:

• : $S = 6 * a^2$

• : $V = a^3$

:

• : $S = 2(a * b + a * h + b * h)$

• : $V = a * b * h$

:

• :

- :

1. **3** : $S = \frac{\sqrt{3}}{4}a^2 + \frac{3}{2}a\sqrt{b^2 - \frac{a^2}{4}}$

2. **4** : $S = a^2 + 2a\sqrt{b^2 - \frac{a^2}{4}}$

3. **6** : $S = 3a(\sqrt{b^2 - \frac{a^2}{4}} + \frac{\sqrt{3}}{2}a)$

- :

1. **3** : $S = S + 3S$

2. **4** : $S = S + 4S$

3. **6** : $S = S + 6S$

• : $V = \frac{4}{3}R^3$

:

• : $S = 2Rh$

• : $V = R^2h$

:

• : $S = Rl$

• : $V = \frac{1}{3}R^2h$

CHAPTER
THREE

```
class figures.Figure

    abstract area() → float

class figures.FlatFigure
    . Figure

    abstract area() → float

    abstract perimeter() → float

class figures.VolumetricFigure
    . Figure

    abstract area() → float

    abstract volume() → float

class figures.Circle(radius: float)
    . FlatFigure

        Parameters radius – (float)
        area() → float

            Returns (float)

        perimeter() → float

            Returns (float)

class figures.Square(a: float)
    . FlatFigure

        Parameters a – (float)
        area() → float
```

Returns (float)

perimeter() → float

Returns (float)

class figures.Rectangle(a: float, b: float)

- . Square

Parameters

- **a** – (float)
- **b** – (float)

area() → float

Returns

perimeter() → float

Returns (float)

class figures.Triangle(a: Optional[float] = None, b: Optional[float] = None, c: Optional[float] = None, h: Optional[float] = None, method: Optional[int] = None)

- . FlatFigure

Parameters

- **a** – (float)
- **b** – (float)
- **c** – (float)
- **h** – (float)
- **method** – (int)

area() → float

- : 1. 2.

Returns (float)

perimeter() → float

Returns (float)

class figures.Trapezium(a: Optional[float] = None, b: Optional[float] = None, c: Optional[float] = None, d: Optional[float] = None, h: Optional[float] = None, method: Optional[int] = None)

- . Triangle

Parameters

- **a** – (float)
- **b** – (float)

-
- **c** – (float)
 - **d** – (float)
 - **h** – (float)
 - **method** – (int)

area() → float

: 1. 2.

Returns (float)

perimeter() → float

Returns (float)

class figures.Diamond(*a*: Optional[float] = None, *h*: Optional[float] = None, *c*: Optional[float] = None, *d*:

Optional[float] = None, *method*: Optional[int] = None)

. Square

Parameters

- **a** – (float)
- **h** – (float)
- **c** – (float)
- **d** – (float)
- **method** – (float)

area() → float

: 1. 2.

Returns (float)

perimeter() → float

Returns (float)

class figures.Sphere(*r*: float)

. VolumetricFigure

Parameters **r** – (float)

area() → float

Returns (float)

volume() → float

Returns (float)

class figures.Cube(*a*: float)

. VolumetricFigure

Parameters `a` – (float)
`area()` → float

Returns (float)

`volume()` → float

Returns (float)

```
class figures.Parallelepiped(a: float, b: float, h: float)
    . Cube
```

Parameters

- `a` – (float)
- `b` – (float)
- `h` – (float)

`area()` → float

Returns (float)

`volume()` → float

Returns (float)

```
class figures.Cylinder(r: float, h: float)
    . Sphere
```

Parameters

- `r` – (float)
- `h` – (float)

`area()` → float

Returns (float)

`volume()` → float

Returns (float)

```
class figures.Cone(r: float, h: float)
    . Cylinder
```

Parameters

- `r` – (float)
- `h` – (float)

```
area() → float
```

Returns (float)

```
volume() → float
```

Returns (float)

```
class figures.Pyramid(a: Optional[float] = None, b: Optional[float] = None, h: Optional[float] = None,  
    area_main: Optional[float] = None, area_side: Optional[float] = None, angle_amount:  
    Optional[float] = None, method: Optional[int] = None)
```

```
    . VolumetricFigure
```

Parameters

- **a** – (float)
- **b** – (float)
- **h** – (float)
- **area_main** – (float)
- **area_side** – (float)
- **angle_amount** – (int)
- **method** – (int)

```
area() → float
```

```
.
```

```
1. 3
```

```
2. 4
```

```
3. 6
```

```
:
```

```
1.
```

```
2.
```

Returns (float)

```
volume() → float
```

Returns (float)

CHAPTER
FOUR

```
class tests.TestFigures(methodName='runTest')
```

Create an instance of the class that will use the named test method when executed. Raises a ValueError if the instance does not have a method with the specified name.

setUp()

test_circle()

test_square()

test_rectangle()

test_triangle()

test_trapezium()

test_diamond()

test_sphere()

test_cube()

test_parallellepiped()

test_cylinder()

test_cone()

test_pyramid()

- genindex

github: [lavalera97](#)

PYTHON MODULE INDEX

f

figures, 7

t

tests, 13

INDEX

A

area() (*figures.Circle method*), 7
area() (*figures.Cone method*), 10
area() (*figures.Cube method*), 10
area() (*figures.Cylinder method*), 10
area() (*figures.Diamond method*), 9
area() (*figures.Figure method*), 7
area() (*figures.FlatFigure method*), 7
area() (*figures.Parallelepiped method*), 10
area() (*figures.Pyramid method*), 11
area() (*figures.Rectangle method*), 8
area() (*figures.Sphere method*), 9
area() (*figures.Square method*), 7
area() (*figures.Trapezium method*), 9
area() (*figures.Triangle method*), 8
area() (*figures.VolumetricFigure method*), 7

C

Circle (*class in figures*), 7
Cone (*class in figures*), 10
Cube (*class in figures*), 9
Cylinder (*class in figures*), 10

D

Diamond (*class in figures*), 9

F

Figure (*class in figures*), 7
figures
 module, 7
FlatFigure (*class in figures*), 7

M

module
 figures, 7
 tests, 13

P

Parallelepiped (*class in figures*), 10
perimeter() (*figures.Circle method*), 7
perimeter() (*figures.Diamond method*), 9

perimeter() (*figures.FlatFigure method*), 7
perimeter() (*figures.Rectangle method*), 8
perimeter() (*figures.Square method*), 8
perimeter() (*figures.Trapezium method*), 9
perimeter() (*figures.Triangle method*), 8
Pyramid (*class in figures*), 11

R

Rectangle (*class in figures*), 8

S

setUp() (*tests.TestFigures method*), 13
Sphere (*class in figures*), 9
Square (*class in figures*), 7

T

test_circle() (*tests.TestFigures method*), 13
test_cone() (*tests.TestFigures method*), 13
test_cube() (*tests.TestFigures method*), 13
test_cylinder() (*tests.TestFigures method*), 13
test_diamond() (*tests.TestFigures method*), 13
test_paralleliped() (*tests.TestFigures method*),
 13

test_pyramid() (*tests.TestFigures method*), 13

test_rectangle() (*tests.TestFigures method*), 13

test_sphere() (*tests.TestFigures method*), 13

test_square() (*tests.TestFigures method*), 13

test_trapezium() (*tests.TestFigures method*), 13

test_triangle() (*tests.TestFigures method*), 13

TestFigures (*class in tests*), 13

tests

 module, 13

Trapezium (*class in figures*), 8

Triangle (*class in figures*), 8

V

volume() (*figures.Cone method*), 11
volume() (*figures.Cube method*), 10
volume() (*figures.Cylinder method*), 10
volume() (*figures.Parallelepiped method*), 10
volume() (*figures.Pyramid method*), 11
volume() (*figures.Sphere method*), 9

`volume()` (*figures.VolumetricFigure method*), [7](#)
`VolumetricFigure` (*class in figures*), [7](#)