



데이터 저장하기

Enhancement Coding

1. 변수 s1에 “#”, s2에 “@” 문자열을 대입한 후, 다음 형태로 출력해보자.

```
###
@@@
#@#@#@#@
```

- ❶ 변수에 대입된 문자열 값은 * 연산자를 이용하여 반복하여 출력하고, (s1 + s2) * 4 형태로 괄호 내에서 문자열 변수들을 + 연산자로 연결한 후 * 연산자로 경우 괄호 부분을 반복하여 출력한다.

```
>>> s1 = "#"
>>> s2 = "@"
>>> print(s1 * 3)
###
>>> print(s2 * 3)
@@@
>>> print((s1 + s2) * 4)
#@#@#@#@
```

2. 반복 횟수에 해당하는 정숫값을 입력받아 변수 rept에 대입하고, 해당 값에 따라 “#” 문자열을 반복하여 출력해보자.

```
반복 : 3
###
반복 : 5
#####
```

- ❶ “#” * rept 형태로 “#” 문자열을 반복하여 출력한다.

```
>>> rept = int(input("반복 : "))
반복 : 3
>>> print("#" * rept)
###
>>> rept = int(input("반복 : "))
반복 : 5
>>> print("#" * rept)
#####
```

3. 현재의 연도에 해당하는 정숫값을 입력받아 변수 year에 대입하고, 2000년으로부터 몇 년이 지났는지 구하여 출력해보자.

```
현재 연도 : 2020
경과 연도 : 20
```

- ❶ 지난 연수는 $\text{year} - 2000$ 으로 계산한다.

```
>>> year = int(input("현재 연도 : "))
현재 연도 : 2020
>>> print("경과 연도 =", year - 2000)
경과 연도 = 20
```

4. 가로와 세로에 해당하는 정숫값을 입력받아 변수 horizontal, vertical에 각각 대입하고, 두 변수를 이용하여 다음과 같이 직사각형의 면적을 구하여 출력해보자.

```
가로 : 5
세로 : 4
면적 = 20
```

- ❶ 직사각형의 면적은 가로 * 세로로 계산한다.

```
>>> horizontal = int(input("가로 : "))
가로 : 5
>>> vertical = int(input("세로 : "))
세로 : 4
>>> print("면적 =", horizontal * vertical)
면적 = 20
```

5. 반지름에 해당하는 정숫값을 입력받아 변수 radius에 대입하고, 해당 값을 이용하여 원의 넓이를 구하여 출력해보자.

```
반지름 : 5
원의 넓이 = 78.5398
```

- ❶ 원의 넓이는 반지름 * 반지름 * 3.141592로 계산한다.

```
>>> radius = int(input("반지름 : "))
반지름 : 5
>>> print("원의 넓이 =", radius * radius * 3.141592)
원의 넓이 = 78.5398
```

6. 정수 3개를 입력받아 변수 a, b, c에 각각 대입하고, 합계를 구하여 변수 sum에 대입하고 평균을 구하여 변수 avg에 대입해보자. 그리고 다음과 같이 모든 변수의 값을 출력해보자.

```
정수1 : 3
정수2 : 5
정수3 : 4
정수1 : 3 정수2 : 5 정수3 : 4
합 : 12 평균 : 4.0
```

- ❶ 합계는 $a + b + c$ 로 계산하고, 평균은 $\text{sum} / 3$ 으로 계산한다.

```
>>> a = int(input("정수1 : "))
```

```

정수1 : 3
>>> b = int(input("정수2 : "))
정수2 : 5
>>> c = int(input("정수3 : "))
정수3 : 4
>>> sum = a + b + c
>>> avg = sum / 3
>>> print("정수1 :", a, "정수2 :", b, "정수3 :", c)
정수1 : 3 정수2 : 5 정수3 : 4
>>> print("합 :", sum, "평균 :", avg)
합 : 12 평균 : 4.0

```

7. 좋아하는 과목 이름과 희망하는 점수를 다음과 같이 입력받아 변수 `subject`와 변수 `score`에 각각 대입하고, 두 변수의 값을 다음 형태로 출력해보자.

```

과목 : 수학
점수 : 95
선택 과목 : 수학 , 희망 점수 : 95

```

- ❶ 문자열 입력은 `input()` 함수를 이용하고, 정수는 `int(input())` 함수 형태로 사용하여 입력한다. `print()` 함수에서 출력 형식대로 문자열을 작성하고 이어서 변수를 지정한다.

```

>>> subject = input("과목 : ")
과목 : 수학
>>> score = int(input("점수 : "))
점수 : 95
>>> print("선택 과목 :", subject, ", 희망 점수 :", score)
선택 과목 : 수학 , 희망 점수 : 95


```

8. x 좌표와 y 좌표에 해당하는 정수를 입력받아 변수 `x`, `y`에 각각 대입하고, 해당 좌표 위치에 거북이의 모양을 도장 찍듯이 남겨보자.(프로그램 문장 작성은 파이썬 에디터를 사용하여 작성하고 실행해보자.)

```

x 좌표 : 0
y 좌표 : 0
x 좌표 : 50
y 좌표 : -50
x 좌표 : 100
y 좌표 : 50

```



- ❶ 파이썬 셸의 [File] 메뉴 > [New File]을 실행하여 파이썬 에디터를 열고 프로그램 문장을 작성한다. 작성이 끝난 후 파이썬 에디터의 [File] 메뉴 > [Save] 또는 [Save As]를 실행하여 작성한 프로그램을 저장하고, [Run] 메뉴 > [Run Module]을 실행하여 작성한 프로그램 문장을 실행한다. `turtle.goto(x, y)` 함수로 거북이를 이동하고, 이동하는 과정에 선을 그리지 않으려면 `turtle.up()` 함수를 사용한다. 거북이의 모양을 도장 찍듯이 남기려면 `turtle.stamp()` 함수를 사용한다.

```
import turtle
```

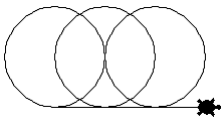
```

turtle.shape("turtle")
turtle.up()
x = int(input("x 좌표 : "))
y = int(input("y 좌표 : "))
turtle.goto(x, y)
turtle.stamp()
x = int(input("x 좌표 : "))
y = int(input("y 좌표 : "))
turtle.goto(x, y)
turtle.stamp()
x = int(input("x 좌표 : "))
y = int(input("y 좌표 : "))
turtle.goto(x, y)
turtle.stamp()

```

9. 반지름에 해당하는 정수를 입력받아 변수 radius에 대입하고, 길이에 해당하는 정수를 입력받아 변수 length에 대입해보자. 그리고 터틀 그래픽에서 반지름이 변수 radius의 값에 해당하는 원을 그리고, 변수 length 만큼 앞으로 이동한다. 원을 그리고 진행하는 동작을 세 번 반복하여 작성해보자. (프로그램 문장 작성은 파이썬 에디터를 사용하여 작성하고 실행해보자.)

반지름 : 100
진행 : 100



- ④ 반지름이 radius인 원을 그리려면 turtle.circle(radius) 함수를 사용하고, length 만큼 앞으로 진행하려면 turtle.forward(length) 함수를 사용한다.

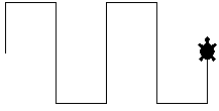
```

import turtle
radius = int(input("반지름 : "))
length = int(input("진행 : "))
turtle.shape("turtle")
turtle.circle(radius)
turtle.forward(length)
turtle.circle(radius)
turtle.forward(length)
turtle.circle(radius)
turtle.forward(length)

```

10. ‘2장의 Enhancement Coding 10번 문제’를 참고하여, 주기 값에 해당하는 정수를 입력받아 변수 t에 대입하고, 크기에 해당하는 정수를 입력받아 변수 p에 대입하자. 그리고 해당 변수들을 이용하여 다음과 같이 파형을 그려보자. (프로그램 문장 작성은 파이썬 에디터를 사용하여 작성하고 실행해보자.)

```
주기 : 100
크기 : 50
```



- 거북이의 방향을 바꾸려면 `turtle.left()`, `turtle.right()` 함수를 사용하고, 오른쪽으로의 이동은 변수 `t`의 $1/2$ 값만큼 각각 이동하고, 위/아래로의 이동은 변수 `p`의 값만큼 이동한다.

```
import turtle
turtle.shape("turtle")
turtle.left(90)
t = int(input("주기 : "))
p = int(input("크기 : "))
th = t / 2
```

```
turtle.forward(p)
turtle.right(90)
turtle.forward(th)
turtle.right(90)
turtle.forward(p*2)
turtle.left(90)
turtle.forward(th)
turtle.left(90)
turtle.forward(p)
```

```
turtle.forward(p)
turtle.right(90)
turtle.forward(th)
turtle.right(90)
turtle.forward(p*2)
turtle.left(90)
turtle.forward(th)
turtle.left(90)
turtle.forward(p)
```



데이터 연산하기

Enhancement Coding

1. 홍길동의 나이에 아버지의 나이를 더하면 38살이고 어머니의 나이는 34살이다. 홍길동의 나이가 현재 5살이면 아버지의 나이와 어머니의 나이는 몇 살인지 계산하여 출력해보자.

```
아버지의 나이 = 33
어머니의 나이 = 29
```

- ④ 아버지의 나이 = 38 - 홍길동의 나이, 어머니의 나이 = 34 - 홍길동의 나이로 계산한다.

```
>>> cage = 5
>>> fage = 38 - cage
>>> mage = 34 - cage
>>> print("아버지의 나이 =", fage)
아버지의 나이 = 33
>>> print("어머니의 나이 =", mage)
어머니의 나이 = 29
```

```
#=====
cage = 5
fage = 38 - cage
mage = 34 - cage
print("아버지의 나이 =", fage)
print("어머니의 나이 =", mage)
```

2. 원의 반지름을 입력받아 변수 radius에 대입하고, 원의 넓이와 둘레를 계산하여 출력해보자.

```
반지름 : 10
원의 넓이 : 314.1592
원의 둘레 : 62.83184
```

- ④ 원의 넓이는 반지름*반지름*3.141592로 계산하고, 원의 둘레는 2*반지름*3.141592로 계산한다. 3.141592 대신에 math 모듈을 포함한 후 math.pi를 사용할 수 있다.

```
>>> radius = int(input("반지름 : "))
반지름 : 10
>>> area = radius * radius * 3.141592
>>> circum = 2 * radius * 3.141592
>>> print("원의 넓이 :", area)
원의 넓이 : 314.1592
>>> print("원의 둘레 :", circum)
원의 둘레 : 62.83184
```

4. 데이터 연산하기

Enhancement Coding

```
#=====

radius = int(input("반지름 : "))
area = radius * radius * 3.141592
circum = 2 * radius * 3.141592
print("원의 넓이 :", area)
print("원의 둘레 :", circum)

#=====

>>> import math
>>> area = radius * radius * math.pi
>>> circum = 2 * radius * math.pi
>>> print("원의 넓이 :", area)
원의 넓이 : 314.1592
>>> print("원의 둘레 :", circum)
원의 둘레 : 62.83184
>>> print(math.pi)
3.141592653589793

#=====

import math
area = radius * radius * math.pi
circum = 2 * radius * math.pi
print("원의 넓이 :", area)
print("원의 둘레 :", circum)
print(math.pi)
```

3. 윗변, 밑변, 높이를 입력받아 변수 top, bottom, height에 각각 대입하고, 사다리꼴의 넓이를 계산하여 출력해보자.

```
윗변 : 6
밑변 : 8
높이 : 4
사다리꼴의 넓이 : 28.0
```

- ④ 사다리꼴의 넓이는 (윗변 + 밑변) * 높이 / 2로 계산한다.

```
>>> top = int(input("윗변 : "))
윗변 : 6
>>> bottom = int(input("밑변 : "))
밑변 : 8
>>> height = int(input("높이 : "))
높이 : 4
>>> area = (top + bottom) * height / 2
>>> print("사다리꼴의 넓이 :", area)
사다리꼴의 넓이 : 28.0

#=====

top = int(input("윗변 : "))
```

4. 데이터 연산하기

Enhancement Coding

```
bottom = int(input("밑변 : "))
height = int(input("높이 : "))
area = (top + bottom) * height / 2
print("사다리꼴의 넓이 :", area)
```

4. 시간의 분(minute)을 입력받아 변수 minute에 대입하고, 초(second)를 계산하여 출력해보자.

```
분 : 12
720 초
```

④ 분에서 초로 변환하려면 60을 곱하여 계산한다.

```
>>> minute = int(input("분 : "))
분 : 12
>>> second = minute * 60
>>> print(second, "초")
720 초
```

#=====

```
minute = int(input("분 : "))
second = minute * 60
print(second, "초")
```

5. 시간의 초(second)를 입력받아 변수 sec에 대입하고, 분(minute)과 초(second)를 계산하여 출력해보자.

```
초 : 345
345 초 = 5 분 45 초
```

④ 초에서 분을 구하기 위해서는 60으로 정수 나눗셈(//)을 하여 몫 값을 구하고, 60으로 나머지 나눗셈(%)을 하여 나머지 값을 구하면 분을 제외한 초를 구할 수 있다.

```
>>> sec = int(input("초 : "))
초 : 345
>>> minute = sec // 60
>>> second = sec % 60
>>> print(sec, "초 = ", minute, "분", second, "초")
345 초 = 5 분 45 초
```

#=====

```
sec = int(input("초 : "))
minute = sec // 60
second = sec % 60
print(sec, "초 = ", minute, "분", second, "초")
```

6. 1초에 3.4m 움직이는 자동차가 있을 때, 시간의 초(second)를 입력받아 해당 초 시간 동안 움직이는 거리를 계산하여 출력해보자.


```
초 : 10
34 m
```

- ❶ 초 * 3.4로 움직이는 거리를 계산한다.

```
>>> sec = int(input("초 : "))
초 : 10
>>> distance = sec * 3.4
>>> print(distance, "m")
34.0 m
```

```
#=====
```

```
sec = int(input("초 : "))
minute = sec // 60
second = sec % 60
print(sec, "초 = ", minute, "분", second, "초")
```

7. 1초당 움직이는 거리를 입력받아 1시간당 움직이는 시속을 계산하여 출력해보자.

```
1초당 움직이는 거리 : 3.4
시속 : 12240.0 m/h
시속 : 12.24 km/h
```

- ❶ 1시간은 60분이고, 1분은 60초이므로, 1시간에 움직이는 거리는 1초당 움직이는 거리 * (60 * 60)으로 계산하고, m/h 값 / 1000으로 km/h 값을 계산한다.

```
>>> distance = float(input("1초당 움직이는 거리 : "))
1초당 움직이는 거리 : 3.4
>>> perhour = distance * (60 * 60)
>>> print("시속 :", perhour, "m/h")
시속 : 12240.0 m/h
>>> print("시속 :", perhour / 1000, "km/h")
시속 : 12.24 km/h
```

```
#=====
```

```
distance = float(input("1초당 움직이는 거리 : "))
perhour = distance * (60 * 60)
print("시속 :", perhour, "m/h")
print("시속 :", perhour / 1000, "km/h")
```

8. 섭씨온도를 입력받아 1초에 움직이는 음속을 계산하여 출력해보자.

```
섭씨온도 : 10
음속 : 12240.0 m/s
```

- ❶ 음속의 공식은 $v = 331 + 0.6 * t$ (v : 음속(m/s), t : 섭씨온도(°C))이므로, $331 + 0.6 * t$ 로 계산한다.

```
>>> t = int(input("섭씨온도 : "))
섭씨온도 : 10
>>> v = 331 + 0.6 * t
```

4. 데이터 연산하기

Enhancement Coding

```
>>> print("음속 :", v, "m/s")
음속 : 337.0 m/s
```

```
#=====
```

```
t = int(input("섭씨온도 : "))
v = 331 + 0.6 * t
print("음속 :", v, "m/s")
```

9. 인치(in)를 입력받아 센티미터(cm)로 변환하여 출력해보자.

```
인치 : 10
10 인치 = 25.4 센티미터
```

④ 1인치는 2.54센티미터이므로, $\text{inch} * 2.54$ 로 계산한다.

```
>>> inch = int(input("인치 : "))
인치 : 10
>>> cm = inch * 2.54
>>> print(inch, "인치 =", cm, "센티미터")
10 인치 = 25.4 센티미터
```

```
#=====
```

```
inch = int(input("인치 : "))
cm = inch * 2.54
print(inch, "인치 =", cm, "센티미터")
```

10. 센티미터(cm)를 입력받아 인치(in)로 변환하여 출력해보자.

```
센티미터 : 30
10 센티미터 = 11.811 인치
```

④ 1센티미터는 0.39370인치이므로, $\text{cm} * 0.39370$ 으로 계산한다.

```
>>> cm = int(input("센티미터 : "))
센티미터 : 30
>>> inch = cm * 0.39370
>>> print(cm, "센티미터 =", inch, "인치")
30 센티미터 = 11.811 인치
```

```
#=====
```

```
cm = int(input("센티미터 : "))
inch = cm * 0.39370
print(cm, "센티미터 =", inch, "인치")
```

11. 닭, 토끼, 돼지의 수를 입력받아 모든 동물의 다리의 수를 계산하여 출력해보자.

```

닭 : 12
토끼 : 8
돼지 : 14
다리 합계 : 112

```

❶ 닭의 다리는 2개, 토끼의 다리는 4개, 돼지의 다리는 4개이다.

```

>>> chicken = int(input("닭 : "))
닭 : 12
>>> rabbit = int(input("토끼 : "))
토끼 : 8
>>> pig = int(input("돼지 : "))
돼지 : 14
>>> leg = chicken * 2 + rabbit * 4 + pig * 4
>>> print("다리 합계 :", leg)
다리 합계 : 112

```

#=====

```

chicken = int(input("닭 : "))
rabbit = int(input("토끼 : "))
pig = int(input("돼지 : "))
leg = chicken * 2 + rabbit * 4 + pig * 4
print("다리 합계 :", leg)

```

12. 하루에 한 갑의 담배를 피울 때 1년 동안의 총지출 금액을 계산하고, 경과 연수를 입력받아 경과 연수에 해당하는 기간의 총지출 금액을 계산하여 출력해보자.(단, 담배 가격은 4500원)

```

경과 연수 : 30
1 년 총지출 = 1642500 원
30 년 총지출 = 49275000 원

```

❶ 1년은 365일이므로 1년 동안의 총지출 금액은 $365 * 4500$ 으로 계산하고, 1년 동안의 총지출 금액 * 경과 연수로 입력한 경과 연수에 해당하는 총지출 금액을 계산한다.

```

>>> y = int(input("경과 연수 : "))
경과 연수 : 12
>>> oneytotal = 365 * 4500
>>> ytotal = oneytotal * y
>>> print("1 년 총지출 =", oneytotal, "원")
1 년 총지출 = 1642500 원
>>> print(y, "년 총지출 =", ytotal, "원")
30 년 총지출 = 49275000 원

```

#=====

```

y = int(input("경과 연수 : "))
oneytotal = 365 * 4500
ytotal = oneytotal * y
print("1 년 총지출 =", oneytotal, "원")
print(y, "년 총지출 =", ytotal, "원")

```

13. 예를 들면 $325 * 7$ 과 같이 세 자릿수와 한 자릿수의 곱셈은 다음과 같이 계산할 수 있다. 세 자릿수 opnd1과 한 자릿수 opnd2를 입력하여 다음과 같이 곱셈하는 과정을 출력해보자.

```

피연산자1 : 325
피연산자2 : 7
325 * 7
= ( 3 + 2 + 5 ) * 7
= 3 * 7 + 2 * 7 + 5 * 7
= 21 + 14 + 35
= 2275

```

- ④ 100 자릿수의 값은 opnd1 // 100으로, 10 자릿수의 값은 opnd1 // 10 % 10으로, 1 자릿수의 값은 opnd1 % 10으로 계산한다.

```

>>> opnd1 = int(input("피연산자1 : "))
피연산자1 : 325
>>> opnd2 = int(input("피연산자2 : "))
피연산자2 : 7
>>> v3 = opnd1 // 100
>>> v2 = opnd1 // 10 % 10
>>> v1 = opnd1 % 10
>>> print(opnd1, "*", opnd2)
325 * 7
>>> print("=", "(", v3, "+", v2, "+", v1, ")", "*", opnd2)
= ( 3 + 2 + 5 ) * 7
>>> print("=", v3, "*", opnd2, "+", v2, "*", opnd2, "+", v1, "*", opnd2)
= 3 * 7 + 2 * 7 + 5 * 7
>>> print("=", v3*opnd2, "+", v2*opnd2, "+", v1*opnd2)
= 21 + 14 + 35
>>> print("=", opnd1*opnd2)
= 2275

```

#=====

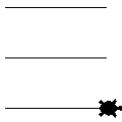
```

opnd1 = int(input("피연산자1 : "))
opnd2 = int(input("피연산자2 : "))
v3 = opnd1 // 100
v2 = opnd1 // 10 % 10
v1 = opnd1 % 10
print(opnd1, "*", opnd2)
print("=", "(", v3, "+", v2, "+", v1, ")", "*", opnd2)
print("=", v3, "*", opnd2, "+", v2, "*", opnd2, "+", v1, "*", opnd2)
print("=", v3*opnd2, "+", v2*opnd2, "+", v1*opnd2)
print("=", opnd1*opnd2)

```

14. ‘2장의 Enhancement Coding 11번 문제’를 참고하여, 길이와 간격을 입력받아 변수 length와 distance에 각각 대입하고 오른쪽으로 길이가 length만큼인 직선들을 distance 간격으로 그려보자.

```
길이 : 100
간격 : 50
```



- ④ 직선을 그릴 때는 `turtle.forward(length)`, 간격을 띄울 때 `turtle.goto(0, -(distance*1))`, `turtle.goto(0, -(distance*2))`로 그린다.

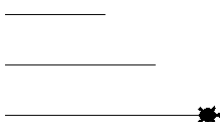
```
>>> length = int(input("길이 : "))
>>> distance = int(input("간격 : "))
>>> import turtle
>>> turtle.shape("turtle")
>>> turtle.forward(length)
>>> turtle.up()
>>> turtle.goto(0, -(distance*1))
>>> turtle.down()
>>> turtle.forward(length)
>>> turtle.up()
>>> turtle.goto(0, -(distance*2))
>>> turtle.down()
>>> turtle.forward(length)
```

```
#=====
```

```
length = int(input("길이 : "))
distance = int(input("간격 : "))
import turtle
turtle.shape("turtle")
turtle.forward(length)
turtle.up()
turtle.goto(0, -(distance*1))
turtle.down()
turtle.forward(length)
turtle.up()
turtle.goto(0, -(distance*2))
turtle.down()
turtle.forward(length)
```

15. ‘Enhancement Coding 14번 문제’를 참고하여, 길이와 간격을 입력받아 변수 `length`와 `distance`에 각각 대입하고 직선들을 `distance` 간격으로 그려보자. 단, 직선은 오른쪽으로 길이가 `length+distance*0`, `length+distance*1`, `length+distance*2`가 되도록 해보자.

```
길이 : 100
간격 : 50
```



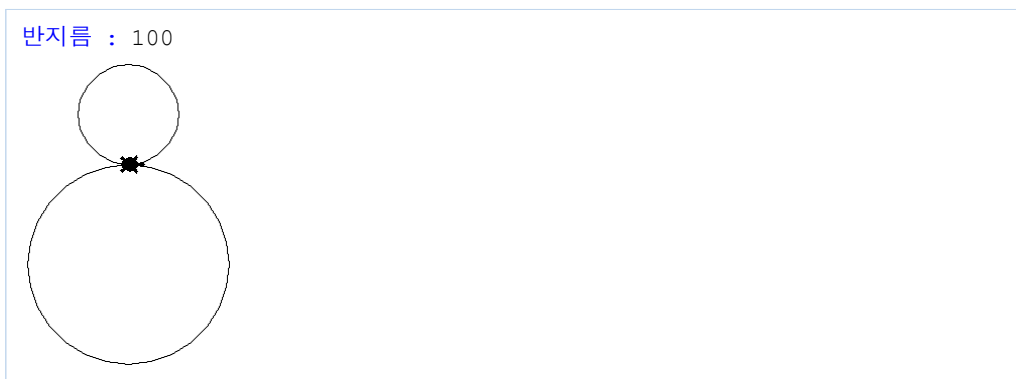
- ❶ 직선을 그릴 때는 `turtle.forward(length+distance*0)`, ..., `turtle.forward(length+distance*2)`, 간격을 띄울 때 `turtle.goto(0, -(distance*1))`, `turtle.goto(0, -(distance*2))`로 그린다.

```
>>> length = int(input("길이 : "))
>>> distance = int(input("간격 : "))
>>> import turtle
>>> turtle.shape("turtle")
>>> turtle.forward(length+distance*0)
>>> turtle.up()
>>> turtle.goto(0, -(distance*1))
>>> turtle.down()
>>> turtle.forward(length+distance*1)
>>> turtle.up()
>>> turtle.goto(0, -(distance*2))
>>> turtle.down()
>>> turtle.forward(length+distance*2)
```

#=====

```
length = int(input("길이 : "))
distance = int(input("간격 : "))
import turtle
turtle.shape("turtle")
turtle.forward(length+distance*0)
turtle.up()
turtle.goto(0, -(distance*1))
turtle.down()
turtle.forward(length+distance*1)
turtle.up()
turtle.goto(0, -(distance*2))
turtle.down()
turtle.forward(length+distance*2)
```

16. 반지름에 해당하는 정수를 입력받아 변수 `radius`에 대입하여 원을 그려보자. 그리고 원에 인접한 위쪽에 해당 반지름의 1/2 크기 반지름으로 원을 그려보자.



- ❶ `turtle.circle(radius)`로 아래쪽 원을 그리고, 위쪽으로 `turtle.goto(0, radius*2)` 만큼 이동한다. 그리고 `turtle.circle(radius/2)`로 위쪽 원을 그린다.

```
>>> radius = int(input("반지름 : "))
>>> import turtle
```

```
>>> turtle.shape("turtle")
>>> turtle.circle(radius)
>>> turtle.up()
>>> turtle.goto(0, radius*2)
>>> turtle.down()
>>> turtle.circle(radius/2)

#=====

radius = int(input("반지름 : "))
import turtle
turtle.shape("turtle")
turtle.circle(radius)
turtle.up()
turtle.goto(0, radius*2)
turtle.down()
turtle.circle(radius/2)
```