

ບົດທີ 04

ຕົວປ່ຽນ ແລະ ການຮັບຄ່າ (Variables and Input)

● Course Overview

- ✓ ເພື່ອໃຫ້ເຂົ້າໃຈກ່ຽວກັບຕົວປ່ຽນ ແລະ ສາມາດສະແດງຄ່າຂອງຕົວປ່ຽນ, ຄ່າຄົງທີ່.
- ✓ ເພື່ອໃຫ້ເຂົ້າໃຈປະເພດຂໍ້ມູນຂອງຕົວປ່ຽນ, ນຳໃຊ້ຕົວດຳເນີນການກຳນົດຄ່າ ແລະ ການນຳໃຊ້ຕົວປ່ຽນ.
- ✓ ເພື່ອໃຫ້ເຂົ້າໃຈກ່ຽວກັບຕົວດຳເນີນການແບບປະສົມ ແລະ ນຳໃຊ້ໃນການຊອກຫາ ຜົນບວກສະສົມຂອງຕົວປ່ຽນ.
- ✓ ເພື່ອໃຫ້ສາມາດກຳນົດ ແລະ ຄຳນວນຄ່າຂອງຕົວປ່ຽນດ້ວຍຕົວດຳເນີນການປະເພດ ຕ່າງໆ.

Course Overview

- ✓ ຮຽນຮູ້ກ່ຽວກັບຕົວປ່ຽນ ແລະ ຕົວດຳເນີນການຕ່າງໆ.
- ✓ ຮຽນຮູ້ກ່ຽວກັບຊະນິດຂໍ້ມູນຂອງຕົວປ່ຽນ ແລະ ຊຸດຂໍ້ຄວາມ (String).
- ✓ ຮຽນຮູ້ກ່ຽວກັບຕົວດຳເນີນການທາງຄະນິດສາດ ແລະ ການດຳເນີນການພື້ນຖານກັບຕົວປ່ຽນ.
- ✓ ຮຽນຮູ້ການຊອກຫາຄ່າຜົນລວມສະສົມຂອງຂໍ້ມູນດ້ວຍການນຳໃຊ້ຕົວດຳເນີນການແບບປະສົມ.
- **Concepts you will need to know from previous units**
 - ✓ ການໃຊ້ຟັງຊັນ `print()` ໃນການສະແດງຜົນຂໍ້ມູນທາງຈໍພາບ ແລະ ການໃຊ້ຟັງຊັນ `input()` ໃນການຮັບຂໍ້ມູນຈາກຜູ້ໃຊ້.
 - ✓ ການໃຊ້ຟັງຊັນ `str()` ໃນການປ່ຽນຂໍ້ມູນຊະນິດຕົວເລກເປັນຊະນິດຂໍ້ຄວາມ.
 - ✓ ການໃຊ້ຕົວດຳເນີນການພື້ນຖານທາງຄະນິດສາດ.

Keywords

ຕົວປ່ຽນ (Variable)

ຄໍາສະຫງວນ
(Reserved Key words)

ຕົວດໍາເນີນການກຳນົດຄ່າ
(Assignment Operator)

ຕົວດໍາເນີນການແບບປະສົມ
(Compound Assignment Operator)

ຕົວດໍາເນີນການທາງຄະນິດສາດ
(Arithmetic Operator)

ບົດທີ 04 ຕົວປຸງນ ແລະ ການຮັບຮັບຄ່າ

| Mission

1. Real world problem

1.1. ບັນຫາກ່ຽວກັບພະຍາດໂລກອ່ວນ

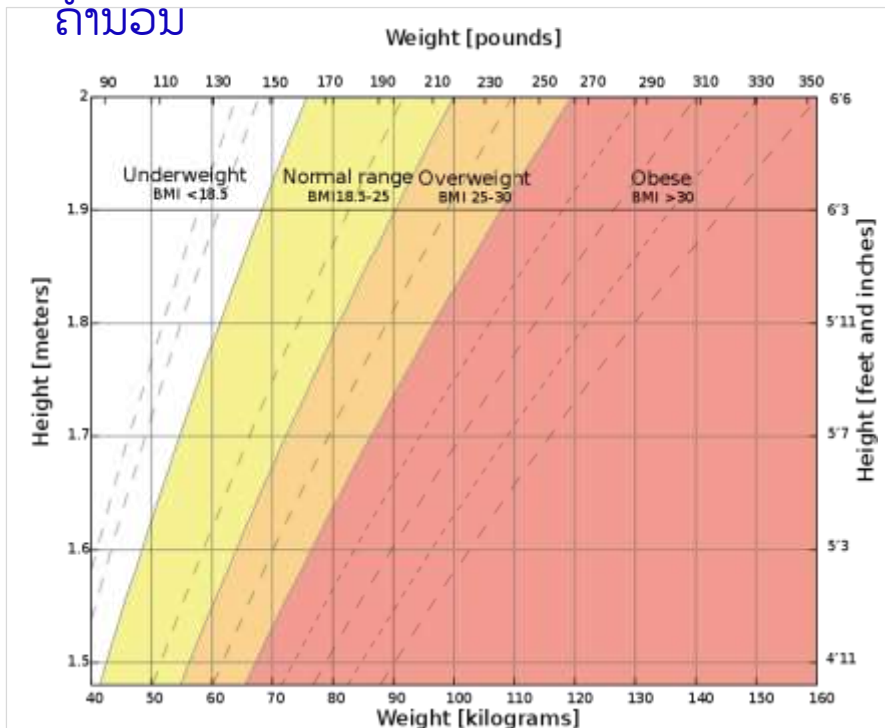


- ອີງຕາມສະຖິຕິຈາກອົງກອນ WHO EURO, 50% ມີລາຍງານວ່າຄົນເອີຣົບມີນ້ຳໜັກເກີນ ຫຼື ເປັນໂລກອ່ວນ. ຈາກການລາຍງານໃນປີ 2019 ມີ 1 ໃນ 8 ຄົນ ຂອງປະຊາກອນໂລກທີ່ເປັນໂລກອ່ວນ.
- ໂລກອ່ວນກຳລັງກາຍເປັນບັນຫາທີ່ສຳຄັນໃນທົ່ວໂລກ
- ມາດຕະຖານຂອງຄວາມອ່ວນໄດ້ວັດແທກຈາກດັດສະນີມວນສານນ້ຳໜັກຮ່າງກາຍ (BMI: Body Mass Index)
- BMI ຖືກວັດແທກຈາກການຄຳນວນຂອງລວງສູງ ແລະ ນ້ຳ

- ▶ ແນວໃດກໍຕາມ ຄວາມຖືກຕ້ອງຍັງບໍ່ຊັດເຈນເນື່ອງຈາກເປົ້າໝາຍຂອງໄຂມັນໃນຮ່າງກາຍວັດແທກໄດ້ໂດຍບໍ່ຕ້ອງໃຊ້ອຸປະກອນແບບມີອາຊີບ.
- ▶ ຍິ່ງໄປກວ່ານັ້ນ ມາດຕະຖານ BMI ຍັງຖືກກຳນົດໂດຍອຸດສາຫະກຳປະກັນໄພ ເຊິ່ງບໍ່ແມ່ນຮັບປະກັນຈາກແພດ ຫຼື ນັກວິທະຍາສາດ.
- ▶ ຕົ້ນກຳນົດຂອງ BMI ແມ່ນໃນປີ 1895 ໂດຍ Metropolitan ປະຈຸບັນແມ່ນ MetLife ດ້ວຍການສ້າງແຜນສະແດງກຣາບເພື່ອບັນທຶກນ້ຳໜັກ ແລະ ລວງສູງຂອງລູກຄ້າ.

1. Real world problem
 - 1.2. BMI ແມ່ນຫຍັງ?

BMI ແມ່ນຄ່າວັດແທກດັດສະນີມວນສານຂອງຮ່າງກາຍຄົນ ຈາກສູດທີ່ມີການນຳເອົາລວງສູງ ແລະ ນ້ຳໜັກມາຄຳນວນ



https://commons.wikimedia.org/wiki/File:Body_mass_index_chart.svg

- ຟັງຊັນສຳລັບການຄິດໄລ່ສຳລັບຊອກຫາຄ່າ $BMI(kg/m^2) = (weight)/(height)^2$. ຫົວໜ່ວຍ ເປັນ kg ສຳລັບນ້ຳໜັກ ແລະ m ສຳລັບລວງສູງ.
- ຕົວຢ່າງ:** BMI ຂອງຄົນທີ່ມີນ້ຳໜັກ 55 kg ແລະ ລວງສູງ 168 cm ຈະເທົ່າກັບ $55/(1.68)^2 = 19.4$
- ມາດຕະຖານຂອງຄວາມອ້ວນແມ່ນອີງໃສ່ຄ່າ BMI ຂອງຄົນເຂດຕາເວັນຕົກແມ່ນຕ່າງກັບຄົນເຂດອາຊີຕາເວັນອອກສ່ຽງໃຕ້

1. Real world problem

1.3. ມາດຕະຖານ WHO BMI

ມາດຕະຖານ WHO ສໍາລັບຄ່າປົກກະຕິຂອງ BMI ແມ່ນຢູ່ໃນຊ່ວງ 18.5 – 24.9, ແລະ ຈະແຕກຕ່າງ ກັນໂດຍອີງຕາມປະຊາກອນໃນແຕ່ລະປະເທດ, ຂົງເຂດ ແລະ ອາຍຸ.

BMI	ສະຖານະພາບໂພສະນາການ
ຕໍ່າກວ່າ 18.5	ນ້ຳໜັກຫຼຸດມາດຕະຖານ
18.5 – 24.9	ນ້ຳໜັກປົກກະຕິ
25.0 – 29.9	ອ້ວນ
30.0 – 34.9	ອ້ວນລະດັບ 1
35.0 – 39.9	ອ້ວນລະດັບ 2
ຫຼາຍກວ່າ 40	ອ້ວນລະດັບ 3

BMI Calculator

US Units Metric Units Other Units

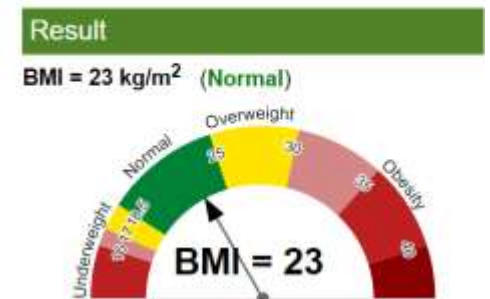
Age: 55 ages: 2 - 120

Gender: ☒ Male ☐ Female

Height: 160 cm

Weight: 59 kg

Calculate Clear



<https://www.euro.who.int/en/health-topics/disease-prevention/nutrition/a-healthy-lifestyle/body-mass-index-bmi>

2. Mission

2.1. ການຄິດໄລ່ຫາ BMI ແມ່ນຫຍັງ?

ເຮົາສາມາດຄິດໄລ່ຫາຄ່າ BMI ຈາກເວັບໄຊ [Calculate Your BMI - Standard BMI Calculator \(nih.gov\)](#) ສະໜັບສະໜູນໂດຍ NIH (ຈະຂ້ອນຂ້າງຕ່າງຈາກມາດຕະຖານຂອງ WHO)

BMI Categories:
Underweight = <18.5
Normal weight = 18.5–24.9
Overweight = 25–29.9
Obesity = BMI of 30 or greater

What Next? Take Action Towards Better Health:

Maintain a Healthy Weight

- Maintaining a healthy weight is important for your heart health.
- Learn more about [overweight and obesity](#)

Increase Physical Activity

- Moving more can lower your risk factors for heart disease.

Eat a Heart-Healthy Diet

- Eating a healthy diet is the key to heart disease prevention.

Know and Control Your Heart Health Numbers

- Tracking your heart health stats can help you meet your heart health goals.

Download the BMI calculator app today

- ກຸ່ມ BMI ຂອງການຈັດລະດັບຄວາມອ້ອນແມ່ນໄດ້ຄິດໄລ່ໂດຍການນຳເອົານ້ຳໜັກຂອງບຸກຄົນຫານດ້ວຍລວງສູງຂຶ້ນກຳລັງສອງ ສະແດງຟັງຊັນຄື: kg/m^2
 - ອີງຕາມເວັບໄຊຂ້າງຊ້າຍນີ້, ຄ່າ BMI ນ້ອຍກວ່າ 18.5 ຫຼື ເທົ່າກັບ 18.5 ຖືວ່າມີນ້ຳໜັກຫຼຸດມາດຕະຖານ; ຖ້າຢູ່ໃນລະຫວ່າງ 18.5 – 24.9 ແມ່ນປົກກະຕິ; ຢູ່ໃນລະຫວ່າງ 25.0 – 29.9 ຖືວ່າ ນ້ຳໜັກເກີນມາດຕະຖານ ແລະ ຫຼາຍກວ່າ 30 ຂຶ້ນໄປຖືວ່າເປັນພະຍາດໂລກອ້ອນ.
- ຕົວຢ່າງ:** ຖ້າທ່ານມີລວງສູງ 170 cm ແລະ ນ້ຳໜັກ 73 kg , ອີງຕາມສູດຄິດໄລ່ ກໍ່ຈະໄດ້: $73/(1.7 \times 1.7) = 25.26$ ສະແດງວ່າຈັດຢູ່ໃນກຸ່ມທີ່ມີນ້ຳໜັກເກີນມາດຕະຖານ.

2. Mission

2.2. ວິທີການຄິດໄລ່ຫາຄ່າ BMI

```
1 name = input("Enter name: ")
2 height_cm = int(input("Enter height(cm): "))
3 height_m = height_cm / 100
4 weight = int(input("Enter weight(kg): "))
5 bmi = weight / (height_m ** 2)
6 print(name, "s BMI is", bmi.)
```

Enter name" David
Enter height(cm): 170
Enter weight(kg): 70
David's BMI is 24.221453287197235.

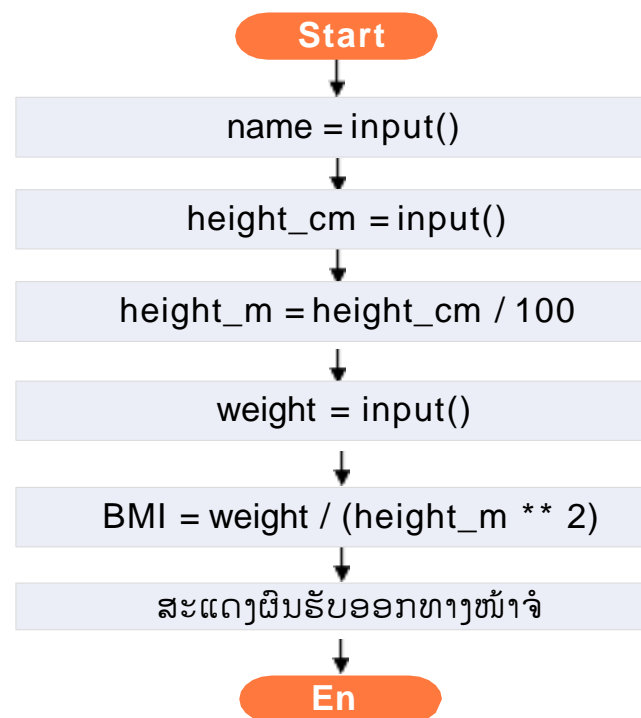
2. Mission

2.3. ແຜນຜັງຂັ້ນຕອນການຂຽນໂປຣແກຣມ

ລະຫັດຈຳລອງ (Pseudocode)

- 1 ເລີ່ມຕົ້ນ
- 2 ປ້ອນຄ່າທີ່ເປັນຊື່
- 3 ປ້ອນຄ່າລວງສູງ (cm)
- 4 ປ່ຽນຫົວໜ່ວຍລວງສູງຈາກ ຊັງຕີແມັດ ເປັນ ແມັດ
- 5 ປ້ອນຄ່ານ້ຳໜັກ (kg)
- 6 ຄິດໄລ່ຄ່າ BMI
- 7 ສະແດງຜົນຮັບອອກທາງໜ້າຈໍ
- 8 ຈົບໂປຣແກຣມ

ແຜນຜັງ (Flowchart)



2. Mission

2.4. ໂປຣແກຣມສຳລັບການຄິດໄລ່ຫາຄ່າ BMI

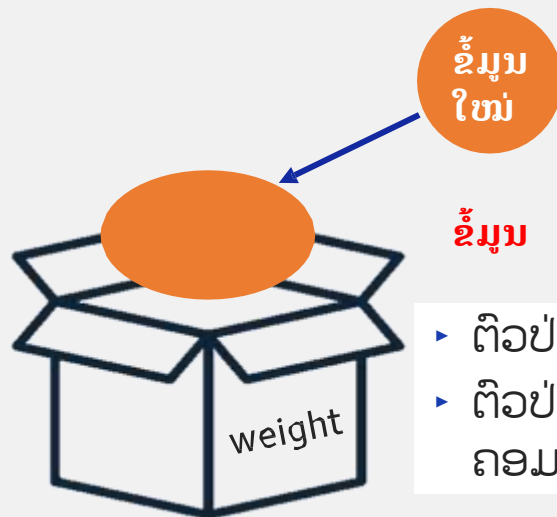
```
1 name = input("Enter name: ")
2 height_cm = int(input("Enter height(cm): "))
3 height_m = height_cm / 100
4 weight = int(input("Enter weight(kg): "))
5 bmi = weight / (height_m ** 2)
6 print(name, "s BMI is", bmi.)
```

| Key concept

1. ການປະກາດຕົວປ່ຽນ

1.1. ຕົວປ່ຽນແມ່ນຫຍັງ?

ໃນລະຫວ່າງການຂຽນໂປຣແກຣມ, ບັນຫາໜຶ່ງທີ່ເກີດຂຶ້ນໃນເວລາປ່ຽນແປງຄ່າຂໍ້ມູນຫຼາຍໆຄ່າອາດສົ່ງຜົນໃຫ້ໄດ້ຜິດຮັບຜິດພາດ. ຈາກບັນຫາດັ່ງກ່າວ ການໃຊ້ຕົວປ່ຽນຈະຊ່ວຍໃຫ້ຂະບວນການນີ້ງ່າຍຂຶ້ນ.



- ▶ ຕົວປ່ຽນ ແມ່ນອົງປະກອບຫຼັກໃນການຂຽນໂປຣແກຣມ.
- ▶ ຕົວປ່ຽນຖືກໃຊ້ໃນການເກັບຄ່າຂໍ້ມູນບາງຢ່າງພາຍໃນຄອມພິວເຕີ

ຄ່າທີ່ຢູ່ໃນກ່ອງສາມາດປ່ຽນແປງຄ່າໄດ້ຕະຫຼອດເວລາ. ເພື່ອໃຫ້ເຂົ້າໃຈງ່າຍທີ່ສຸດກ່ຽວກັບຄ່ານີ້ອາດຈະເອີ້ນວ່າ ຂໍ້ມູນ. ຂໍ້ມູນສາມາດເກັບເປັນຄ່າດຽວໄດ້ ເຊັ່ນ: ນ້ຳໜັກ (weight) = 78.7 ແລະ ຍັງສາມາດເກັບໄດ້ຫຼາຍໆຄ່າໃນຕົວປ່ຽນດຽວ ເຊັ່ນ: person = ('David', 78.7).

1. ການປະກາດຕົວປ່ຽນ

1.2. ໄວຍະກອນຂອງການປະກາດຕົວປ່ຽນ

ສໍາລັບຕົວທີ່ໃຊ້ໃນການເກັບຄ່າຂໍ້ມູນຈະເອີ້ນວ່າ: **ຕົວປ່ຽນ**. ສໍາລັບພາສາ Python ແມ່ນບໍ່ຈໍາເປັນຕ້ອງກໍານົດຊະນິດຂໍ້ມູນຂອງຕົວປ່ຽນ, ເຊິ່ງຈະຕ່າງຈາກພາສາໂປຣແກຣມອື່ນໆ ເຊັ່ນ: ພາສາ C, Java ທີ່ຈໍາເປັນຕ້ອງກໍານົດ. ຕົວຢ່າງ: ສົມມຸດວ່າ ເຮົາເກັບຂໍ້ມູນນໍ້າໜັກດ້ວຍຕົວປ່ຽນຊື່ weight. ນອກຈາກນີ້ເຮົາຍັງສາມາດເກັບຄ່ານໍ້າໜັກ ແລະ ຊື່ ໄວ້ໃນຕົວປ່ຽນດຽວກັນເຊັ່ນ: person ໂດຍໃຊ້ເຄື່ອງໝາຍ () ສັງເກດດັ່ງລຸ່ມນີ້:

```
1 weight = 78.7
2 person = ('David', 78.7)
```

ແຖວ 1, 2

- ໃນເວລາປະມວນຜົນໂປຣແກຣມ ຕົວປ່ຽນຊື່ weight ຈະຖືກສ້າງໃນໜ່ວຍຄວາມຈໍາ ແລະ ເກັບຄ່າ 78.7 .
- ໃນຕົວປ່ຽນຊື່ person ກໍ່ຈະເກັບຄ່າ David, 78.7 ເຊິ່ງເປັນຕົວປ່ຽນຊະນິດ Tuple.
- ເຄື່ອງໝາຍ = ແມ່ນຕົວດໍາເນີນການກໍານົດຄ່າ ໂດຍມີຄວາມໝາຍວ່າ ຄ່າຂໍ້ມູນທີ່ຢູ່ທາງເບື້ອງຂວາຈະຖືກສັ່ງໃຫ້ເກັບໄວ້ໃນຕົວປ່ຽນຢູ່ທາງດ້ານຊ້າຍ

1. ການປະກາດຕົວປ່ຽນ

1.2. ໄວຍະກອນຂອງການປະກາດຕົວປ່ຽນ

 **Focus** ການກຳນົດ ແລະ ເອີ້ນໃຊ້ງານຄ່າຂໍ້ມູນໃນຕົວປ່ຽນ ຈະຊ່ວຍໃຫ້ຫຼຸດຜ່ອນຂໍ້ຜິດພາດ ແລະ ແກ້ໄຂໂປຣແກຣມໄດ້ງ່າຍຂຶ້ນ.

```
1 radius = 4.0
2 print('Radius of Circle', radius)
3 print('Area of Circle', 3.14 * radius * radius)
4 print('Circumference of Circle', 2.0 * 3.14 * radius)
```

Radius of Circle 4.0 Area of Circle 50.24
Circumference of Circle 25.12

1. ການປະກາດຕົວປ່ຽນ

1.2. ໄວຍະກອນຂອງການປະກາດຕົວປ່ຽນ

ຊື່ຕົວປ່ຽນ: ພວກເຮົາຈະຮຽນຮູ້ການເຮັດວຽກກັບຂໍ້ມູນຕ່າງໆ, ແຕ່ເພື່ອຈະເຮັດແນວນັ້ນໄດ້ພວກເຮົາຕ້ອງການຊື່ຕົວປ່ຽນທີ່ແຕກຕ່າງກັນໃນການເກັບຂໍ້ມູນເຊັ່ນ: ຊື່ຄົນ Jame ແລະ Michael, ລະບຸຕົວປ່ຽນຕ່າງກັນ.



- ▶ ຊື່ຕົວປ່ຽນ ຕ້ອງຂຽນຕາມກົດເກນດັ່ງລຸ່ມນີ້:
 - ຊື່ຕົວປ່ຽນ ສາມາດເປັນຕົວອັກສອນ, ຕົວເລກ ຫຼື ເຄື່ອງໝາຍ _ ປະສົມກັນກັນ. ຫ້າມໃຊ້ສັນຍະລັກພິເສດ.
 - ຊື່ຕົວປ່ຽນບໍ່ສາມາດຂຶ້ນຕົ້ນດ້ວຍຕົວເລກ.
 - ພາຍໃນຊື່ຕົວປ່ຽນຫ້າມມີການຍະຫວ່າງ ຫຼື ແທ້ບ.
 - ຊື່ຕົວປ່ຽນຕົວພິມນ້ອຍ ແລະ ຕົວພິມໃຫຍ່ຖືກວ່າເປັນຕົວປ່ຽນຕ່າງກັນ
 - ຊື່ຂອງຕົວປ່ຽນຫ້າມຊ້ຳກັບຄຳສະຫງວນ

1. ການປະກາດຕົວປ່ຽນ

1.3. ການປະກາດຕົວປ່ຽນ ແລະ ຊື່ຕົວປ່ຽນ

ຕົວຢ່າງຂອງຊື່ຕົວປ່ຽນ: ການຄິດໄລ່ເນື້ອທີ່ຂອງຮູບສີ່ແຈສາກ ເມື່ອລວງກວ້າງເທົ່າກັບ 10 ແລະ ລວງສູງເທົ່າກັບ 5 ເຮົາສາມາດຕັ້ງຊື່ຕົວປ່ຽນ width ເກັບລວງຍາວ ແລະ height ໃຊ້ເກັບລວງສູງ, ຈາກນັ້ນທຸກການຄິດໄລ່ໃນທີ່ນີ້ຈະເຫັນວ່າການຕັ້ງຊື່ຕົວປ່ຽນຂ້ອນຂ້າງງ່າຍໆ

```
1 width = 10
2 height = 5
3 rectangle_area = width * height
4 print('rectangle_area =', rectangle_area)
```

```
rectangle_area = 50
```



ແຖວ 1, 2

- ເປັນການຕັ້ງຊື່ຕົວປ່ຽນໂດຍໃຊ້ຄຳວ່າ width ແລະ height ເຮົາສາມາດປ່ຽນຄ່າຂອງ width ແລະ height ທີ່ແທນຄ່າລວງຍາວ ແລະ ລວງສູງຂອງຮູບສີ່ແຈດ້ວຍຄ່າອື່ນໆໄດ້.

1. ການປະກາດຕົວປ່ຽນ

1.4. ຄໍາສະຫງວນໃນພາສາ Python

ໃນຄະນະທີ່ພວກເຮົາສາມາດຕັ້ງຊື່ຕົວປ່ຽນໄດ້ຢ່າງອິດສະຫຼະ, ແຕ່ພວກເຮົາບໍ່ສາມາດນໍາໃຊ້ຄໍາສະຫງວນໃນພາສາ Python ເພື່ອຕັ້ງເປັນຊື່ຂອງຕົວປ່ຽນ, ເຊິ່ງຄໍາສະຫງວນສະແດງດັ່ງຕາຕະລາງລຸ່ມນີ້:

ຄໍາສະຫງວນໃນພາສາ Python ທີ່ເຮົາບໍ່ສາມາດນໍາມາຕັ້ງເປັນຊື່ຂອງຕົວປ່ຽນ

False	class	return	is	finally	None
if	for	lambda	continue	True	def
from	while	nonlocal	and	del	global
not	with	as	elif	try	or
yield	assert	else	import	pass	break
except	in	raise			

1. ການປະກາດຕົວປ່ຽນ

1.4. ຄໍາສະຫງວນໃນພາສາ Python

ການໃຊ້ຄໍາສະຫງວນໃນການຕັ້ງຊື່ຕົວປ່ຽນຈະເກີດຜົນຮັບທີ່ຜິດພາດ ສະແດງດັ່ງລຸ່ມນີ້:

SyntaxError

```
1 global = 500
```

```
File "<ipython-input-6-6cbcd63281ae>", line 1
```

```
global = 500
      ^
```

```
SyntaxError: invalid syntax
```

1. ການປະກາດຕົວປຸງນ

1.5. ແນະນຳກ່ຽວກັບ Python Keywords



ວິທີຄົ້ນຫາ Python Keywords:

Ex

```
1 import keyword
2
3 # Search for Python Keywords
4 print("Python keyword list...")
5 print(keyword.kwlist)
```

Python keyword list...

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continu
e', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import',
'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'wit
h', 'yield']
```

ເພີ່ມເຕີມ

➤ ເຈົ້າຕັ້ງຊື່ໃຫ້ກັບຕົວປ່ຽນແນວໃດ?

- ໃນເວລາຕັ້ງຊື່ຕົວປ່ຽນ ເປັນເລື່ອງສໍາຄັນໃນການຕັ້ງຊື່ຕົວປ່ຽນທີ່ຈະອະທິບາຍ ແລະ ສື່ຄວາມໝາຍກັບຂໍ້ມູນທີ່ເກັບ
- ຊື່ຕົວປ່ຽນທີ່ດີຈະອ່ານງ່າຍໃນການຂຽນໂປຣແກຣມ

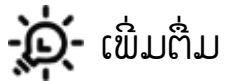
Ex ການໃຊ້ຕັ້ງຊື່ຕົວປ່ຽນໂດຍໃຊ້ຄຳວ່າ width ແລະ height ໃນການເກັບຄ່າລວງຍາວ ແລະ ລວງສູງ ຈະເຂົ້າໃຈງ່າຍກວ່າການຕັ້ງຊື່ເປັນ x1, x2

```
1 weight = 78.2
2 height = 180.0
```

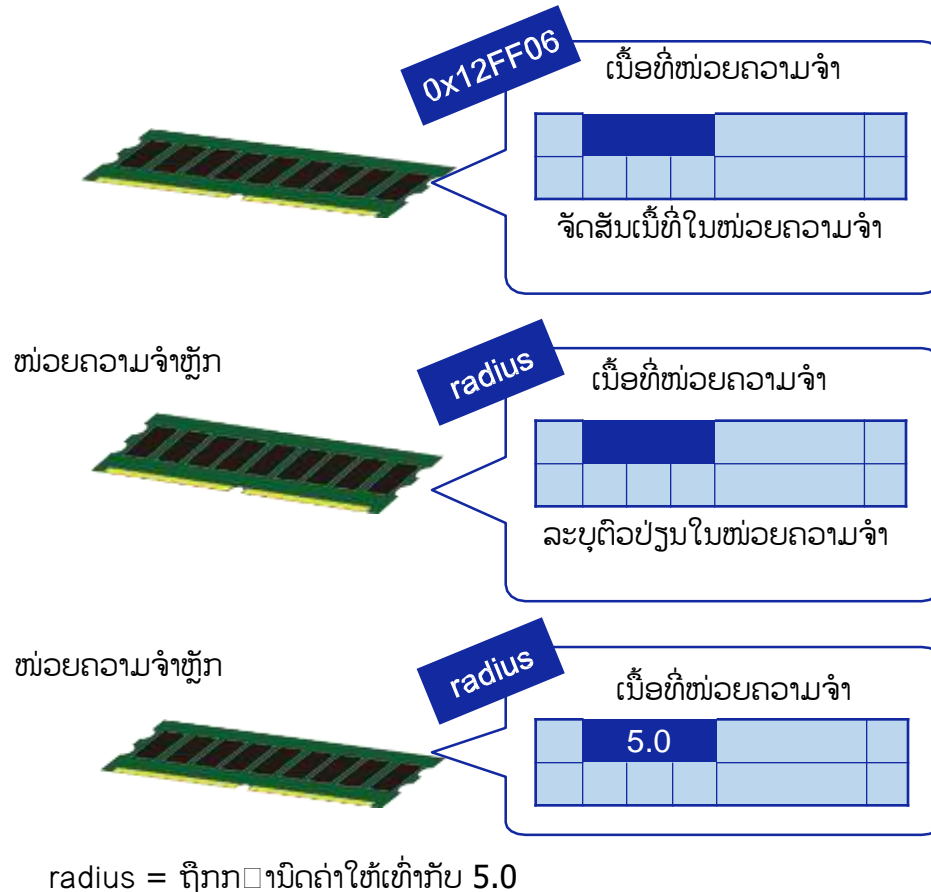


```
1 x1 = 78.2
2 x2 = 100.0
```





ເພີ່ມຕື່ມ



- **ໜ່ວຍຄວາມຈຳຫຼັກ:** ໜ່ວຍຄວາມຈຳແມ່ນບ່ອນທີ່ຄອມພິວເຕີໃຊ້ສຳລັບເກັບ, ອ່ານ, ຂຽນ ແລະ ຂຽນທັບຂໍ້ມູນ.
- **ໝາຍເລກທີ່ຢູ່ໃນໜ່ວຍຄວາມຈຳ:** ແມ່ນ ໝາຍເລກທີ່ຢູ່ບ່ອນເກັບຂໍ້ມູນໃນໜ່ວຍຄວາມຈຳ ເພື່ອຈະອ່ານ/ຂຽນຂໍ້ມູນເປັນສິ່ງຈຳເປັນທີ່ຈະຕ້ອງຮູ້ວ່າຂໍ້ມູນຖືກເກັບໄວ້ໃນຕຳແໜ່ງໃດໃນໜ່ວຍຄວາມຈຳ, ປົກກະຕິຕົວເລກທີ່ຢູ່ຈະເປັນເລກຖານ 16.
- ມັນສະດວກຕໍ່ການໃຊ້ຊື່ຕົວປ່ຽນ ເຊັ່ນ: radius ຕ່າງຈາກໜ່ວຍຄວາມຈຳ.

1. ການປະກາດຕົວປ່ຽນ

1.6. String

ທີ່ຜ່ານມານັ້ນ ພວກເຮົາໄດ້ເກັບຂໍ້ມູນຊະນິດຕົວເລກ **ຈຳນວນຖ້ວນ ແລະ ຈຳນວນຈິງ** ໃນຕົວປ່ຽນ. ເຖິງຢ່າງໃດກໍຕາມ **ຍັງສາມາດເກັບຂໍ້ມູນຊະນິດຂໍ້ຄວາມ**. ໃນການຂຽນໂປຣແກຣມ, ຂໍ້ມູນທີ່ເປັນຂໍ້ຄວາມ ຈະເອີ້ນວ່າ: **String**.

ການສ້າງຂໍ້ຄວາມຈະຖືກກຳນົດໄວ້ໃນເຄື່ອງໝາຍວົງຢືມດຽວ ຫຼື ຄູ່ ກໍໄດ້ເຊັ່ນ: **“Hello World”** ຫຼື **‘Hello World’**

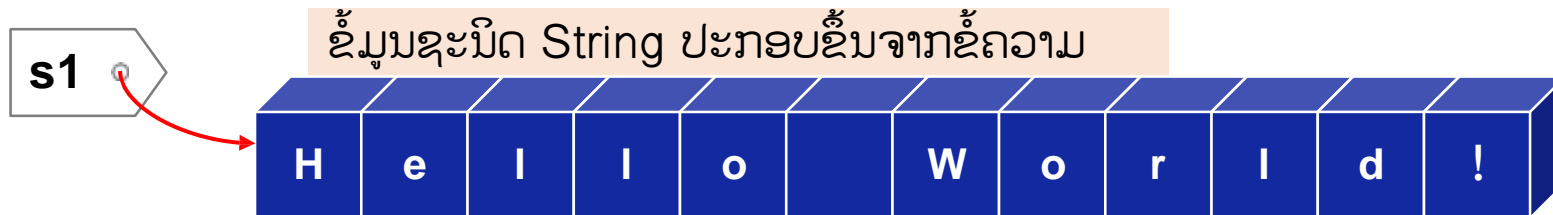
```
1 s1 = 'Hello World'
2 print(s1)
```

Hello World

1. ການປະກາດຕົວປ່ຽນ

1.6. String

- ໃນພາສາ Python, ຂໍ້ມູນຊະນິດຂໍ້ຄວາມ ສະແດງດັ່ງຕົວຢ່າງຂ້າງເທິງເອີ້ນວ່າ: Object
- ໃນຕົວຢ່າງຕໍ່ໄປນີ້ ຕົວປ່ຽນ s1 ຄວນຈະມີແທັກ ເພື່ອອ້າງອີງເປັນ Object



1. ການປະກາດຕົວປ່ຽນ

1.6. String

String ສາມາດປະຕິບັດການເພີ່ມຕົວດໍາເນີນການ ໂດຍການເຊື່ອມຕໍ່ 2 String ເຂົ້າດ້ວຍກັນດ້ວຍເຄື່ອງໝາຍ + ສະແດງດັ່ງຕົວຢ່າງລຸ່ມນີ້:

```
1 s1 = "Hello"  
2 s2 = "World!"  
3 print(s1 + s2)
```

HelloWorld!

1. ການປະກາດຕົວປ່ຽນ

1.7. ຊະນິດຂໍ້ມູນໃນພາສາ Python

ຊະນິດຂໍ້ມູນ: ການນິຍາມ ແລະ ກວດເຊັກຊະນິດຂໍ້ມູນ

ຊະນິດຂໍ້ມູນ

- ຊະນິດຂໍ້ມູນ ແມ່ນປະເພດຂອງຂໍ້ມູນທີ່ສາມາດປະມວນຜົນໃນພາສາໂປຣແກຣມ.
- ຊະນິດຂໍ້ມູນພື້ນຖານ ປະກອບມີ: Boolean, ຕົວເລກ (ເລກຈຳນວນຖ້ວນ, ຈຳນວນຈິງ, ຈຳນວນສົນ), Strings, Lists, Tuple, Sets ແລະ Dictionaries.
- ຟັງຊັນ `type()` ອະນຸຍາດກວດເບິ່ງຊະນິດຂໍ້ມູນທີ່ໃຫ້ຜົນຮັບອອກມາເປັນ Object.

1. ການປະກາດຕົວປ່ຽນ

1.7. ຊະນິດຂໍ້ມູນໃນພາສາ Python

ການປະກາດຕົວປ່ຽນໂດຍບໍ່ມີການກຳນົດຊະນິດຂໍ້ມູນຂອງຕົວປ່ຽນ, ສາມາດນຳໃຊ້ກັບຕົວປ່ຽນທີ່ເກັບຄ່າຂໍ້ມູນຊະນິດຕ່າງກັນໄດ້, ສະແດງດັ່ງຕົວຢ່າງລຸ່ມນີ້:

```
1 num = 85
2 print(type(num))
3
4 pi = 3.14159
5 print(type(pi))
6
7 message = "Good morning"
8 print(type(message))
```

```
<class 'int'>
<class 'float'>
<class 'str'>
```

ເພີ່ມເຕີມ

- ການປະກາດຕົວປ່ຽນທີ່ບໍ່ມີການກຳນົດຊະນິດຂໍ້ມູນ: ຈະເກີດຂຶ້ນໃນຄະນະທີ່ກຳລັງຮັບໂປຣແກຣມ.
- ການປະກາດຕົວປ່ຽນທີ່ມີການນຳນິດຊະນິດຂໍ້ມູນ: ຈະເກີດຂຶ້ນລ່ວງໜ້າກ່ອນການຮັບໂປຣແກຣມ.
- ພາສາ Python ນຳໃຊ້ການປະກາດຕົວປ່ຽນແບບບໍ່ມີການກຳນົດຊະນິດຂໍ້ມູນ ໝາຍຄວາມວ່າ: ການຂຽນໂປຣແກຣມຈະມີຄວາມຢືດຢຸນ.
- ພາສາໂປຣແກຣມທີ່ໃຊ້ການປະກາດຕົວປ່ຽນແບບມີການກຳນົດຊະນິດຂໍ້ມູນ ເຊັ່ນ: ພາສາ C ອາດຈະມີການກັ່ນຕອງການທຳງານທີ່ມີການກຳນົດຄ່າຜິດຊະນິດຂໍ້ມູນ ຫຼື ພະຍາຍາມປະມວນຜົນທີ່ບໍ່ມີຂໍ້ມູນທີ່ສາມາດດຳເນີນການໄດ້, ໂດຍຜ່ານການວິເຄາະໂຄດໂປຣແກຣມກ່ອນການຮັບໂປຣແກຣມ.

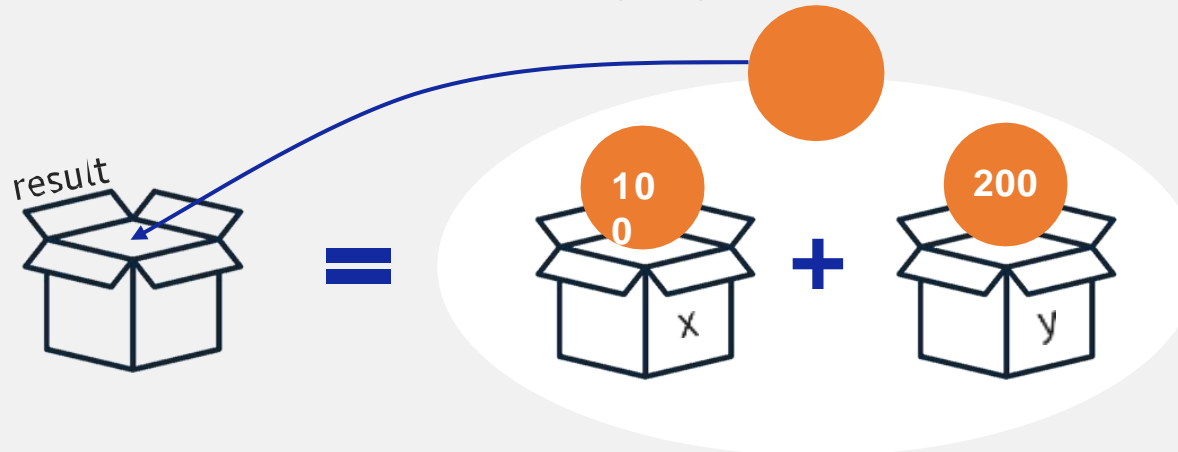
ການປະກາດຕົວປ່ຽນທີ່ມີການກຳນົດຊະນິດຂໍ້ມູນ	ການປະກາດຕົວປ່ຽນທີ່ບໍ່ມີການກຳນົດຊະນິດຂໍ້ມູນ
<ul style="list-style-type: none">- ກວດສອບຊະນິດຂໍ້ມູນກ່ອນການຄອມໄຟລ- ຈຳເປັນຕ້ອງມີການລະບຸຊະນິດຂໍ້ມູນແຕ່ລະປະເພດ- ໄດ້ແກ່ພາສາ: C, C++, C#, JAVA, Object-C, Pascal, ອື່ນໆ	<ul style="list-style-type: none">- ຂໍ້ມູນແຕ່ລະປະເພດບໍ່ຈຳເປັນຕ້ອງມີຂໍ້ກຳນົດ ສິ່ງຜິດໃຫ້ໄດ້ Code ໂປຣແກຣມງ່າຍໆ- ອາດຈະສິ່ງຜິດໃຫ້ເກີດຂໍ້ຜິດພາດປະເພດຂໍ້ມູນໃນລະຫວ່າງການປະມວນຜົນ- ໄດ້ແກ່ພາສາ: Python, Basic, Ruby, PHP, JavaScript...

1. ການປະກາດຕົວປ່ຽນ

1.8. ຕົວດຳເນີນການກຳນົດຄ່າໃນພາສາ Python

ໃຫ້ສັງເກດສຳນວນຕໍ່ໄປນີ້ $\text{result} = 100 + 200$.

ໃນສຳນວນຂ້າງເທິງນີ້ ໝາຍຄວາມວ່າ: ພວກເຮົາກຳນົດຄ່າຂໍ້ມູນທີ່ຢູ່ທາງເບື້ອງຂວາໃຫ້ເກັບໄວ້ໃນຕົວປ່ຽນທີ່ຢູ່ທາງເບື້ອງຊ້າຍ



ເຄື່ອງໝາຍ '=' ໃນ $\text{result} = 100 + 200$ ບໍ່ແມ່ນເຄື່ອງເທົ່າກັບ, ມັນແທນການເກັບຄ່າຂໍ້ມູນທີ່ຢູ່ທາງຂວາຂອງເຄື່ອງໝາຍດຳເນີນການ = ໄວ້ໃນຕົວປ່ຽນ result ທີ່ຢູ່ທາງເບື້ອງຊ້າຍຂອງເຄື່ອງໝາຍ =
ເຄື່ອງໝາຍ = ນີ້ເອີ້ນວ່າ: ຕົວດຳເນີນການກຳນົດຄ່າ ຫຼື assignment operator.

2. ຕົວດໍາເນີນການກຳນົດຄ່າ ແລະ ການອ້າງອິງ

2.1. ການປະກາດ ແລະ ກຳນົດຄ່າພ້ອມກັນໃຫ້ກັບຕົວປ່ຽນໃນພາສາ Python

 ສຸມໃສ່ ດັ່ງສະແດງໃນສໍານວນ `x, y = 100, 200` ເປັນການປະກາດ ແລະ ກຳນົດຄ່າຕົວປ່ຽນແບບພ້ອມກັນໃນແຖວດຽວກັນ ເຊິ່ງເອີ້ນວິທີນີ້ວ່າ ການປະກາດ ແລະ ກຳນົດຄ່າພ້ອມກັນໃຫ້ກັບຕົວປ່ຽນ.

```
1 x, y = 100, 200
2 result = x + y
3 print(result)
```

300

2. ຕົວດຳເນີນການກຳນົດຄ່າ ແລະ ການອ້າງອີງ

2.1. ການກຳນົດຄ່າພ້ອມກັນໃຫ້ກັບຕົວປ່ຽນໃນພາສາ Python

ດັ່ງທີ່ສະແດງໃຫ້ເຫັນ `num1 = num2 = num3 = 200` ເປັນການກຳນົດຄ່າ 200 ໃຫ້ກັບຕົວປ່ຽນຫຼາຍໆຕົວ

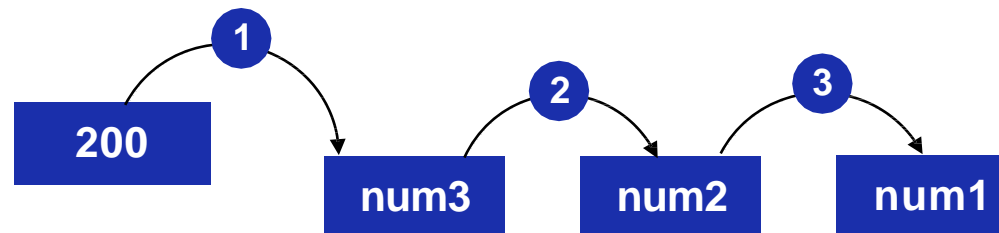
```
1 num1 = num2 = num3 = 200
2 print(num1, num2, num3)
```

200 200 200

File

ແຖວ 1

- 200 ໄດ້ຖືກກຳນົດໃຫ້ເກັບໄວ້ໃນຕົວປ່ຽນ `num1`, `num2` ແລະ `num3`. ລຳດັບການກຳນົດແມ່ນເລີ່ມຈາກຊ້າຍ ເຊິ່ງສະແດງດັ່ງລຸ່ມນີ້:



ລຳດັບການກຳນົດຄ່າໃຫ້ກັບຕົວປ່ຽນ

2. ຕົວດຳເນີນການກຳນົດຄ່າ ແລະ ການອ້າງອີງ

2.2. ເຄື່ອງໝາຍເທົ່າກັບໃນພາສາ Python

⚙️ ເຄື່ອງໝາຍເທົ່າກັບ ບໍ່ໄດ້ໝາຍຄວາມວ່າມີສອງຄ່າຂໍ້ມູນເທົ່າກັນ.

```
1 300 = 300
```

```
File "<ipython-input-8-b44a4a79e963>", line 1
```

```
300 = 300
    ^
```

SyntaxError: cannot assign to literal

- ເຄື່ອງໝາຍເທົ່າກັບ '=' ບໍ່ໄດ້ໝາຍຄວາມວ່າ ສອງຄ່າໃດໜຶ່ງເທົ່າກັນ, ແຕ່ເປັນການກຳນົດໃຫ້ຄ່າທີ່ຢູ່ເບື້ອງຂວາ ໃຫ້ເກັບໄວ້ໃນຕົວປ່ຽນທີ່ຢູ່ເບື້ອງຊ້າຍ, ຍ້ອນວ່າ 300 ບໍ່ສາມາດກຳນົດໃຫ້ເກັບໄວ້ໃນ 300 ດັ່ງນັ້ນຜົນການຮັບຈຶ່ງ ຜິດພາດ
- ເພື່ອຄວາມແນ່ໃຈ ແລະ ບໍ່ໃຫ້ສັບສົນກັບຕົວດຳເນີນການປຽບທຽບ ==

3. ຕົວດຳເນີນການກຳນົດຄ່າແບບປະສົມ

3.1. ຄວາມຫຼາກຫຼາຍຂອງຕົວດຳເນີນການກຳນົດຄ່າແບບປະສົມໃນພາສາ Python


- ໃນພາສາ Python ມີຕົວດຳເນີນການກຳນົດຄ່າແບບປະສົມຫຼາຍແບບ. ເຖິງວ່າຈະບໍ່ຄຸ້ນເຄີຍກັບຕົວດຳເນີນການແບບປະສົມກໍຕາມ ເຮົາຍັງສາມາດນຳໃຊ້ຕົວດຳເນີນການທາງຄະນິດສາດ ແລະ ຕົວດຳເນີນການກຳນົດຄ່າໂດຍປັດສະຈາກບັນຫາໃນການຂຽນ Code ໂປຣແກຣມ.
- ໃນກໍລະນີບໍ່ຮູ້ວ່າ ຕົວດຳເນີນການແບບປະສົມທີ່ນັກຂຽນໂປຣແກຣມມັກນຳໃຊ້ໃນເວລາຂຽນໂປຣແກຣມ, ເຖິງຢ່າງໃດກໍຕາມບັນຫາອາດຈະມີໃນເວລາອ່ານ Code ໂປຣແກຣມບາງຢ່າງ.

ຕົວດຳເນີນການແບບປະສົມ

ຕົວດຳເນີນການ	ຄຳອະທິບາຍ	ຕົວຢ່າງ
+=	ການລວມກັນຂອງຕົວດຳເນີນການເພີ່ມ ແລະ ຕົວດຳເນີນການກຳນົດຄ່າ	i+=10
-=	ການລວມກັນຂອງຕົວດຳເນີນການລົບ ແລະ ຕົວດຳເນີນການກຳນົດຄ່າ	i-=10
=	ການລວມກັນຂອງຕົວດຳເນີນການຄູນ ກັບຕົວດຳເນີນການກຳນົດຄ່າ	i=10
/=	ການລວມກັນຂອງຕົວດຳເນີນການຫານ ແລະ ຕົວດຳເນີນການກຳນົດຄ່າ	i/=10
=	ການລວມກັນຂອງຕົວດຳເນີນການຂຶ້ນກຳລັງ ແລະ ຕົວດຳເນີນການກຳນົດຄ່າ	i^=10
%=	ການລວມກັນຂອງຕົວດຳເນີນການຫານ(ເອົາຄ່າເສດ) ແລະ ຕົວດຳເນີນການກຳນົດຄ່າ	i%=10

3. ຕົວຕຳເນີນການກຳນົດຄ່າແບບປະສົມ

3.1. ຄວາມຫຼາກຫຼາຍຂອງຕົວຕຳເນີນການກຳນົດຄ່າແບບປະສົມໃນພາສາ Python

 ມັນຈຳເປັນທີ່ຕ້ອງຮູ້ຕົວຕຳເນີນການກຳນົດຄ່າແບບປະສົມ, ເຮົາອາດບໍ່ສາມາດອ່ານ ແລະ ເຂົ້າໃຈ Code ໂປຣແກຣມ ຖ້າເຮົາບໍ່ຮູ້ຕົວຕຳເນີນການປະເພດນີ້

```
1 num = 200
2 num += 100
3 print(num)
4 num -= 100
5 print(num)
6 num *= 20
7 print(num)
8 num /= 2
9 print(num)
```

```
300
200
4000
2000.0
```


4. ຕົວດຳເນີນການທາງຄະນິດສາດ

4.1. ຕົວດຳເນີນການທາງຄະນິດສາດໃນພາສາ Python

ຕົວດຳເນີນການທາງຄະນິດສາດ ໄດ້ແກ່: ບວກ, ລົບ, ຄູນ, ຫານ, ຫານເອົາຄ່າເສດ ແລະ ຕົວດຳເນີນການຂຶ້ນກຳລັງ.

ສຳນວນຕົວເລກທີ່ເໝາະສົມສາມາດເພີ່ມປະສິດທິພາບໃນການຄຳນວນ ແລະ ຫຼຸດຄວາມສັບຊ້ອນຂອງ Code ໂປຣແກຣມ.

ໃນສຳນວນ $a = 10 + 1$, $10 + 1$ ທີ່ຢູ່ທາງເບື້ອງຂວາຂອງເຄື່ອງໝາຍ $=$ ແມ່ນຄ່າປະເມີນ. ຄ່ານີ້ຖືກຄຳນວນ ແລະ ກຳນົດໂດຍຕົວດຳເນີນການທາງຄະນິດສາດ ແລະ ກຳນົດໃຫ້ເກັບໄວ້ໃນຕົວປ່ຽນ

$$\begin{array}{c} 11 \\ a \end{array} \quad a = 10+1 \quad = \boxed{10+1} = 11$$


4. ຕົວດຳເນີນການທາງຄະນິດສາດ

4.2. ວິທີການເຮັດວຽກຂອງຕົວດຳເນີນການທາງຄະນິດສາດ

ໃນສຳນວນ $a + b$, ເຄື່ອງໝາຍ $+$ ແມ່ນຕົວດຳເນີນການ, ສ່ວນ a, b ແມ່ນຕົວຖືກດຳເນີນການ. ຕົວດຳເນີນການໃນພາສາ Python ອະທິບາຍດັ່ງຕາຕະລາງລຸ່ມນີ້:

ຕົວດຳເນີນການ	ຄວາມໝາຍ	ການດຳເນີນການ
+	ບວກ	ບວກຄ່າຂອງຕົວຖືກດຳເນີນການທັງເບື້ອງຊ້າຍ ແລະ ຂວາ
-	ລົບ	ລົບຄ່າຂອງຕົວຖືກດຳເນີນການທັງເບື້ອງຊ້າຍ ແລະ ຂວາ
*	ຄູນ	ຄູນຄ່າຂອງຕົວຖືກດຳເນີນການທັງເບື້ອງຊ້າຍ ແລະ ຂວາ
/	ຫານ	ຫານຄ່າຂອງຕົວຖືກດຳເນີນການທັງເບື້ອງຊ້າຍ ແລະ ຂວາ ແລ້ວໄດ້ຜົນຮັບເປັນເລກຈຳນວນຈິງ
//	ຫານເອົາຄ່າຖ້ວນ	ຕ່າງຈາກການຫານແບບ $/$, ເຊິ່ງຈະເອົາຜົນຮັບທີ່ເປັນຈຳນວນຖ້ວນ
%	ຫານເອົາຄ່າເສດ	ເປັນການຫານທີ່ເອົາຜົນຮັບເປັນຈຳນວນເສດ
**	ຂຶ້ນກຳລັງ	ຕົວຖືກດຳເນີນການເບື້ອງຊ້າຍຂຶ້ນກຳລັງດ້ວຍຕົວຖືກດຳເນີນການເບື້ອງຂວາ

4. ຕົວດຳເນີນການທາງຄະນິດສາດ

4.2. ວິທີການເຮັດວຽກຂອງຕົວດຳເນີນການທາງຄະນິດສາດ

ຄຳແນະນຳ: ຮາກຂັ້ນສອງຂອງ 4 ສາມາດຂຽນເປັນ $4^{**}0.5$ ແລ້ວຈະໄດ້ຜົນຮັບເທົ່າກັບ 2.

ຕົວດຳເນີນການ	ສັນຍາລັກ	ຕົວຢ່າງ	ຜົນໄດ້ຮັບ
ບວກ	+	$7 + 4$	11
ລົບ	-	$7 - 4$	3
ຄູນ	*	$7 * 4$	28
ຫານຈຳນວນຈິງ	/	$7 / 4$	1.75
ຫານຈຳນວນຖ້ວນ	//	$7 // 4$	1
ຫານເອົາຄ່າເສດ	%	$7 \% 4$	3
ຂຶ້ນກຳລັງ	**	$7 ** 2$	49

4. ຕົວຕຳເນີນການທາງຄະນິດສາດ

4.2. ວິທີການເຮັດວຽກຂອງຕົວຕຳເນີນການທາງຄະນິດສາດ



ສັງເກດຊະນິດຂໍ້ມູນຂອງແຕ່ລະຕົວປ່ຽນທີ່ຮັບເຂົ້າມາ ແລະ ວິທີການໃຊ້ຕົວປ່ຽນໃນການຄຳນວນ.

ໂປຣແກຣມຕໍ່ໄປນີ້ ໄດ້ມີການແປງຄ່າຂໍ້ມູນທີ່ຮັບເຂົ້າມາໃຫ້ເປັນຊະນິດຈຳນວນຈິງ (float). ສ່ວນຟັງຊັນ `inf(input())` ແມ່ນໃຊ້ສຳລັບແປງຄ່າຂໍ້ມູນທີ່ຮັບເຂົ້າມາໃຫ້ເປັນຕົວເລກຊະນິດຈຳນວນຖ້ວນ.

```
1 weight = float(input("Enter weight in kg: "))
2 height = float(input("Enter height in m: "))
3
4 bmi = (weight / (height ** 2))
5 print("Your BMI=", bmi)
```

Enter weight in kg: 70 Enter height in m: 1.75
Your BMI= 22.857142857142858

4. ຕົວດຳເນີນການທາງຄະນິດສາດ

4.3. ຟັງຊັນ input() ຂອງ Python ແລະ ຕົວດຳເນີນການທາງຄະນິດສາດ

- ຟັງຊັນ `int(input(''))` ປົກກະຕິນຳໃຊ້ສຳລັບຮັບຄ່າຈາກຜູ້ໃຊ້ ແລະ ແປງຊະນິດຂໍ້ມູນທີ່ຮັບມາເປັນຊະນິດເລກຈຳນວນຖ້ວນ.
- ແກ້ໄຂບັນຫາການແລກປ່ຽນເງິນຕາໃນກໍລະນີເຮົາມີພຽງເງິນໃບ 500 ແລະ ໃບ 100 ວອນ.

ວອນແມ່ນສະກຸນເງິນຂອງເກົາຫຼີ

```
1 money = int(input("Money Paid : "))
2 price = int(input("Price of Item : "))
3
4 change = money-price
5 print("Amount of Change : ", change)
6 coin500s = change // 500
7 change = change % 500
8 coin100s = change // 100
9
10 print("Pieces of 500 won coins: ", coin500s)
11 print("Pieces of 100 won coins: ", coin100s)
```

Money Paid: 5000
Price of Item: 300
Amount of Change: 4700
Pieces of 500 won coins :9
Pieces of 100 won coins: 2

ແຖວ 1-2

- ຮັບຈຳນວນເງິນ ແລະ ລາຄາຈາກຜູ້ໃຊ້ ແລະ ແປງຂໍ້ມູນເປັນຕົວເລກຊະນິດຈຳນວນຖ້ວນ
- ຫຼັກຈາກນັ້ນ, ບັນຫາສາມາດຖືກແກ້ໄຂດ້ວຍຕົວດຳເນີນການທານເລກຈຳນວນຖ້ວນ.

4. ຕົວດຳເນີນການທາງຄະນິດສາດ

4.3. ຟັງຊັນ input() ຂອງ Python ແລະ ຕົວດຳເນີນການທາງຄະນິດສາດ

- ຫຼັງຈາກຮັບຄ່າທີ່ປ້ອນເຂົ້າມາ, ຜົນຮັບຕ່າງໆທີ່ຕ້ອງການສາມາດຊອກຫາໄດ້ໂດຍການນຳໃຊ້ຕົວດຳເນີນການ.

```
1 money = int(input("Money Paid : "))
2 price = int(input("Price of Item : "))
3
4 change = money-price
5 print("Amount of Change : ", change)
6 coin500s = change // 500
7 change = change % 500
8 coin100s = change // 100
9
10 print ("Pieces of 500 won coins: ", coin500s)
11 print ("Pieces of 100 won coins: ", coin100s)
```

Money Paid: 5000
Price of Item: 300
Amount of Change: 4700
Pieces of 500 won coins :9
Pieces of 100 won coins: 2

ແຖວ 4-8

- ຄ່າຂອງການປ່ຽນແມ່ນເອົາ $\text{money} - \text{price}$.
- ໃນຫຼຽນ 500 ວອນ ແມ່ນເອົາ $\text{change} \% 500$ ຈາກນັ້ນເກັບຄ່າໄວ້.

4. ຕົວດຳເນີນການທາງຄະນິດສາດ

4.3. ຟັງຊັນ input() ຂອງ Python ແລະ ຕົວດຳເນີນການທາງຄະນິດສາດ

- ຫຼັງຈາກຮັບຄ່າທີ່ປ້ອນເຂົ້າມາ, ຜົນຮັບຕ່າງໆທີ່ຕ້ອງການສາມາດຊອກຫາໄດ້ໂດຍການນຳໃຊ້ຕົວດຳເນີນການ.

```
1 money = int(input("Money Paid : "))
2 price = int(input("Price of Item : "))
3
4 change = money-price
5 print("Amount of Change : ", change)
6 coin500s = change // 500
7 change = change % 500
8 coin100s = change // 100
9
10 print("Pieces of 500 won coins: ", coin500s)
11 print("Pieces of 100 won coins: ", coin100s)
```

Money Paid: 5000
Price of Item: 300
Amount of Change: 4700
Pieces of 500 won coins :9
Pieces of 100 won coins: 2

ແຖວ 4-8

- ເກັບສ່ວນທີ່ເຫຼືອຈາກການຫານ change % 500 ໄວ້ໃນ change.
- ໃນຫຼຽນ 100, ແມ່ນເອົາ change % 100 ຈາກນັ້ນເກັບໄວ້.

4. ຕົວດຳເນີນການທາງຄະນິດສາດ

4.4. ຂໍ້ຄວນລະວັງໃນການຕົວດຳເນີນການທາງຄະນິດສາດ

 ການໃຊ້ສົມຜົນທີ່ຜິດ ກໍ່ຈະເຮັດໃຫ້ໄດ້ຜົນຮັບທີ່ຜິດພາດ.

```
1 num1 = 3
2 num2 = 0
3 print(num1/num2)
```

```
-----
ZeroDivisionError                                Traceback (most recent call last)
<ipython-input-18-b729e0caa109> in <module>
      1 num1 = 3
      2 num2 = 0
----> 3 print(num1/num2)
```

ZeroDivisionError: division by zero

- ໃນເວລາໃຊ້ຕົວດຳເນີນການທາງຄະນິດສາດ, ໃຫ້ກວດເບິ່ງໃຫ້ແນ່ໃຈກ່ອນວ່າມີການຜິດພາດທາງດ້ານສົມຜົນ ຫຼື ບໍ່.
- ຖ້າຄ່າຕົວເລກຖືກຫານໃຫ້ 0, ຄວາມຜິດພາດກໍ່ຈະເກີດຂຶ້ນ. ໃຫ້ລະວັງ **ZeroDivisionError**

ບົດທີ 04 ຕົວປຸງນ ແລະ ການຮັບຮັບຄ່າ

Paper Coding

- ພະຍາຍາມທຳຄວາມເຂົ້າໃຈພື້ນຖານແນວຄວາມຄິດກ່ອນຈະກ້າວໄປຍັງຂັ້ນຕອນຖັດໄປ.
- ຖ້າບໍ່ເຂົ້າໃຈພື້ນຖານແນວຄວາມຄິດກໍ່ຈະພາໃຫ້ມີບັນຫາໃນການຮຽນສຳລັບຫຼັກສູດນີ້ ອາດຈະເຮັດໃຫ້ເຮົາບໍ່ຜ່ານຫຼັກສູດນີ້.
- ມັນອາດຈະຂ້ອນຂ້າງຍາກໃນຕອນນີ້, ແຕ່ຜົນສຳເລັດຂອງຫຼັກສູດນີ້ພວກເຮົາຂໍແນະນຳໃຫ້ທ່ານເຂົ້າໃຈຢ່າງລະອຽດກ່ຽວກັບແນວຄວາມຄິດ ແລ້ວຈຶ່ງກ້າວໄປຂັ້ນຕອນຕໍ່ໄປ.

Q1.

ໃຫ້ກຳນົດຄ່າຕົວເລກ 30, 60 ໄວ້ໃນຕົວປ່ຽນ width ແລະ height, ຕາມລຳດັບ.
ໃຫ້ຂຽນໂປຣແກຣມໂດຍນຳໃຊ້ຕົວປ່ຽນສອງຕົວດັ່ງກ່າວນີ້ ເພື່ອຄິດໄລ່ຫນ້າທີ່ຂອງຮູບສີ່ແຈສາກ ເພື່ອໃຫ້
ສະແດງຜົນຮັບດັ່ງລຸ່ມນີ້:

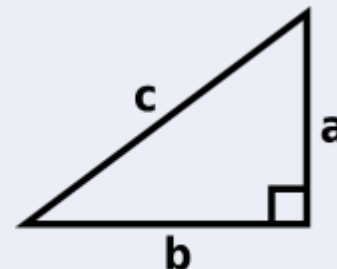
ເງື່ອນໄຂໃນການ ດຳເນີນການ	"Print "Area of Rectangle : 1800."
ໃຊ້ເວລາ	5 ນາທີ




ຂຽນໂຄດທັງໝົດ ແລະ ຜົນໄດ້ຮັບໄວ້ໃນປຶ້ມບັນທຶກ

Q2. ຫຼັກການປີຕາກຳກ່າວວ່າ: ຂ້າງເຊິ່ງໜ້າມູມສາກ c ເທົ່າກັບຜົນບວກຂຶ້ນກຳລັງສອງຂອງສອງຂ້າງຕິດມູມສາກ a ແລະ b . ໃຫ້ຂຽນໂປຣແກຣມຄິດໄລ່ຫາຂ້າງເຊິ່ງໜ້າ ໂດຍມີການຮັບຄ່າຟື້ນ ແລະ ລວງສູງ ເປັນຊະນິດຕົວເລກຈຳນວນຖ້ວນ.

ເງື່ອນໄຂການ ດຳເນີນການ	ປ້ອນຄ່າຂ້າງຟື້ນໃນແຖວທີ 1 ແລະ ຄ່າລວງສູງໃນແຖວທີ 2 . ໃນແຖວທີ 3 ໃຫ້ສະແດງຜົນຮັບຂອງການຊອກຫາຄ່າລວງຍາວຂອງຂ້າງເຊິ່ງໜ້າມູມສາກ
ເວລາ	5 ນາທີ
ຫຼັກການປີຕາກຳ	$c^2 = a^2 + b^2$



$$c^2 = a^2 + b^2$$

 ຂຽນໂຄດທັງໝົດ ແລະ ຜົນໄດ້ຮັບໄວ້ໃນປຶ້ມບັນທຶກ

| Let's Code

1. ຮັບຄ່າ String ຈາກຜູ້ໃຊ້ດ້ວຍຟັງຊັນ input()

1.1. ການນຳໃຊ້ຟັງຊັນ input()

ຊື່ຜູ້ໃຊ້ຖືກເກັບໄວ້ໃນຕົວປ່ຽນ name ຈາກນັ້ນຂຽນ Code ໂປຣແກຣມ ເພື່ອສະແດງຄ່າຕົວປ່ຽນດັ່ງຕົວຢ່າງລຸ່ມນີ້:

```
1 name = input("Enter Name: ")
2 print("Hello", name)
3 print("Welcome to the World of Python Programming.")
```


1. ຮັບຄ່າ String ຈາກຜູ້ໃຊ້ດ້ວຍຟັງຊັນ input()
 - 1.1. ການນຳໃຊ້ຟັງຊັນ input()

ກວດເບິ່ງຜົນໄດ້ຮັບ:

```
1 name = input ( " Enter Name: ")  
2 print ("Hello", name.)  
3 print ("Welcome to the World of Python Programming.")
```

Enter Name: David

Hello, David.

Welcome to the World of Python
Programming.

1. ຮັບຄ່າ String ຈາກຜູ້ໃຊ້ດ້ວຍຟັງຊັນ input()

1.2. ການຮັບຄ່າຕົວເລກຈຳນວນເຕັມຈາກຜູ້ໃຊ້

ການຂຽນໂປຣແກຣມທີ່ຮັບຕົວດຳເນີນການທາງຄະນິດສາດດ້ວຍການຮັບຄ່າຈາກຜູ້ໃຊ້

- ▶ ໃນການບວກ, ໂປຣແກຣມທີ່ໄດ້ຮຽນຜ່ານມາ ຜົນຮັບຂອງ 100+200 ແມ່ນຖືກສະແດງປົກກະຕິ.
- ▶ ມັນຈະເປັນປະໂຫຍດຫຼາຍກວ່າເມື່ອຂຽນໂປຣແກຣມທີ່ມີການຮັບຄ່າສອງຄ່າທີ່ເປັນຊະນິດຕົວເລກຈຳນວນຖ້ວນ, ຈາກນັ້ນສະແດງຜົນອອກມາ

```
1 x = int(input("Enter the first integer: "))
2 y = int(input("Enter the second integer: "))
3 s = x + y
4 print("The sum of," x, "and" y, "is", s)
```

Enter the first integer: 300

Enter the second integer: 400

The sum of 300 and 400 is 700

- ຟັງຊັນ input() ໃຊ້ສຳລັບຮັບຄ່າຕົວເລກຈຳນວນຖ້ວນ, ເຖິງຢ່າງໃດກໍຕາມ, ຟັງຊັນ input() ຈະໃຫ້ຄ່າມາເປັນຊະນິດຕົວອັກສອນ ຫຼື String, ດັ່ງນັ້ນ 300 ແມ່ນຊະນິດ String.
- ເພື່ອຈະໃຫ້ 300 ກາຍເປັນຊະນິດຕົວເລກ ເຮົາຈະຕ້ອງໃຊ້ຟັງຊັນ int() ເພື່ອແປງເປັນຕົວເລກຈຳນວນຖ້ວນ.

1. ຮັບຄ່າ String ຈາກຜູ້ໃຊ້ດ້ວຍຟັງຊັນ input()

1.2. ການຮັບຄ່າຕົວເລກຈຳນວນເຕັມຈາກຜູ້ໃຊ້

ຖ້າເຮົາປ້ອນຕົວອັກສອນ ຫຼື ຂໍ້ຄວາມ ເຊັ່ນ: hundred ແທນຕົວເລກ 300 ດັ່ງຕົວຢ່າງລຸ່ມນີ້, ຟັງຊັນ int() ບໍ່ສາມາດແປງໃຫ້ເປັນຊະນິດຕົວເລກຈຳນວນຖ້ວນໄດ້ ແລະ ເຮົາກໍຈະໄດ້ຜິດພາດຮັບທີ່ ຜິດພາດ.

❗ ລະອຽດຜິດພາດດ້ານຄ່າຂໍ້ມູນ

```
1 #If the characters "hundred" are entered, the int() function fails to convert it into an integer and will print an error.
2 x = int(input("      Enter the first integer: "))
3 y = int(input("      Enter the second integer: "))
4 s = x + y
5 print(x("The sum of," x, "and" y, "is", s)
```

Enter the first integer: hundred

```
-----
ValueError                                TrTraceback (most recent call last)
<i>ipython-input-2-611617428447> in <module>
      1 #If the characters "hundred" are entered, the int() function fails to convert it into an integer and will print an error.
----> 2 x = int(input("Enter the first integer: "))
      3 y = int(input("Enter the second integer: "))
      4 s = x + y
      5 pprint("The sum of," x, "and" y, "is", s)

ValueError: invalid literal for int() with base 10: 'hundred'
```

ເພີ່ມເຕີມ

- ພາສາ Python ສາມາດກວດເຊັກຊະນິດຂໍ້ມູນດ້ວຍຟັງຊັນ `type()` ເຊັ່ນຕົວຢ່າງ `x = int(x)` ເຊິ່ງຂໍ້ມູນໃນ `x` ຕ້ອງເປັນຕົວເລກຈຳນວນຖ້ວນເທົ່ານັ້ນຈຶ່ງເປັນໄປໄດ້. ນອກຈາກນີ້ເຮົາຍັງສາມາດກວດເຊັກຊະນິດຂໍ້ມູນຂອງ `x` ດ້ວຍຟັງຊັນ `isdigit()` ເຊິ່ງຈະໃຫ້ຄ່າເປັນຊະນິດ Boolean ເຊັ່ນ: `x.isdigit() == True`.

```
1 x = input("Enter the first integer: ") ('Input
2 printData Type: ', type(x))
```

Enter the first integer: 100 Input Data
Type: <class 'str'>

```
1 # Let's check whether x is only a number.
2 if x.isdigit() == True :
3     x = int(x)
4     print('Input Data Type: ', type(x))
5 else:
6     print(xis not a number type.)
```

Input Data Type: <class 'int'>

2. ພື້ນຖານຂອງຕົວປ່ຽນ

2.1. ການໃຊ້ຕົວປ່ຽນສໍາລັບການຂຽນໂປຣແກຣມທີ່ມີປະສິດທິພາບ

- ໃນເວລາເຮົາຂຽນໂປຣແກຣມ ຕົວປ່ຽນເປັນສິ່ງທີ່ຈຳເປັນຕ້ອງໃຊ້ສໍາລັບການເກັບຂໍ້ມູນຊະນິດຕ່າງໆ.



```
radius = 4.0  
name = input('input your name ')
```

- ການນຳໃຊ້ຕົວປ່ຽນ ຈະເຮັດໃຫ້ວຽກງານທີ່ມີຄວາມສັບຊ້ອນງ່າຍຂຶ້ນ ເພາະເຮົາສາມາດເກັບຄ່າຕ່າງໆໄວ້ໃນຕົວປ່ຽນ, ຈາກນັ້ນກໍເອີ້ນໃຊ້ງານຕົວປ່ຽນ.

2. ພື້ນຖານຂອງຕົວປ່ຽນ

2.1. ການໃຊ້ຕົວປ່ຽນສໍາລັບການຂຽນໂປຣແກຣມທີ່ມີປະສິດທິພາບ

- ການຮັບຄ່າຂໍ້ຄວາມ: ວິທີການເກັບຄ່າຂໍ້ມູນທີ່ເປັນຊະນິດຂໍ້ຄວາມ (String) ໄວ້ໃນຕົວປ່ຽນ.



2. ພື້ນຖານຂອງຕົວປ່ຽນ

2.1. ການໃຊ້ຕົວປ່ຽນສໍາລັບການຂຽນໂປຣແກຣມທີ່ມີປະສິດທິພາບ

ການຮັບຄ່າຂໍ້ຄວາມ: ວິທີການເກັບຄ່າຂໍ້ມູນທີ່ເປັນຊະນິດຂໍ້ຄວາມ (String) ໄວ້ໃນຕົວປ່ຽນ.

```
1 radius = 4.0
2 print('Radius of Circle', radius)
3 print('Area of Circle', 3.14 * radius * radius)
4 print('Circumference of Circle', 2.0 * 3.14 * radius)
```

Radius of Circle 4.0 Area of Circle 50.24
Circumference of Circle 25.12

```
1 name = input('input your name: ')
```

input your name:

```
1 name = input('input your name: ')
```

input your name: David

2. ພື້ນຖານຂອງຕົວປ່ຽນ

2.2. ຊະນິດຂໍ້ມູນຂອງຕົວປ່ຽນ

▪ ປະເພດຕົວປ່ຽນ

- ຕົວປ່ຽນແຕ່ລະຕົວຕ້ອງມີເນື້ອທີ່ພຽງພໍໃນໜ່ວຍຄວາມຈໍາສໍາລັບເກັບຄ່າຂອງມັນ.
- ໃນການກໍານົດພື້ນທີ່ຂອງໜ່ວຍຄວາມຈໍາທີ່ຕ້ອງການ ແລະ ບັນທຶກ/ອ່ານຄ່າ ຂໍ້ມູນໃນພື້ນທີ່ດັ່ງກ່າວຈໍາເປັນຈະຕ້ອງກໍານົດຊະນິດຂໍ້ມູນ.
- ໃນພາສາ Python ແມ່ນບໍ່ຈໍາເປັນຕ້ອງກໍານົດຊະນິດຂໍ້ມູນ.

```
1 name = 'David'
2 age = 27
3 print(type(name))
4 print(type(age))
```

```
<class 'str'>
<class 'int'>
```


2. ພື້ນຖານຂອງຕົວປ່ຽນ

2.3. ຂໍ້ຄວນລະວັງສໍາລັບ TypeError ເວລາໃຊ້ງານຕົວປ່ຽນ

TypeError ຈະເກີດຂຶ້ນໃນເວລາທີ່ຄ່າຂໍ້ມູນເປັນ String ເຊັ່ນ: ປະກາດຕົວປ່ຽນ my_height ເພື່ອບວກກັບເລກຈຳນວນຖ້ວນ ດັ່ງໂປຣແກຣມລຸ່ມນີ້:



TypeError

```
1 my_age = 22
2 my_height = '177'
3 my_age = my_age + 1
4 my_height = my_height + 1
5 print(my_age, my_height)
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-9-1e339f59d7ef> in <module>
      2 my_height = '177'
      3 my_age = my_age + 1
----> 4 my_height = my_height + 1
      5 print(my_age, my_height)
```

TypeError: can only concatenate str (not "int") to str

ບົດທີ 04 ຕົວປຸງນ ແລະ ການຮັບຮັບຄ່າ

| Pair Programming



ຝຶກການຂຽນໂປຣແກຣມແບບເປັນຄູ່

ການຂຽນໂປຣແກຣມແບບເປັນຄູ່ ເປັນການຂຽນໂປຣແກຣມທີ່ໃຫ້ສອງຄົນເຮັດວຽກໃນງານດຽວກັນ ພ້ອມກັນໂດຍໃຊ້ຄອມພິວເຕີໜ່ວຍດຽວ ແຕ່ໃຊ້ສະໜອງສອງຄົນຊ່ວຍກັນຄິດ ເຊິ່ງອາດຈະໃຫ້ຄົນທີ່ເກັ່ງກວ່າ ເປັນຄົນເບິ່ງ ຄົນທີ່ອ່ອນກວ່າ(ໃນກໍລະນີທີ່ຄວາມສາມາດຂອງສອງຄົນຕ່າງກັນ) ແຕ່ໃນກໍລະນີຄົນສອງຄົນ ຄວາມສາມາດໃກ້ຄຽງກັນ ກໍ່ອາດຈະໃຫ້ຄົນໃດຄົນໜຶ່ງເປັນຄົນຂຽນ ແລະ ອີກຄົນໃຫ້ເປັນຜູ້ຕິດຕາມ,ແນະນຳ ຖ້າມີຄຳຜິດກໍ່ບອກຄົນຂຽນເພື່ອແກ້ໄຂໃຫ້ໂປຣແກຣມທີ່ຂຽນນັ້ນຖືກຕ້ອງ.

- ແນວທາງການຂຽນໂປຣແກຣມແບບນີ້ໃຊ້ໄດ້ຫຼາຍກໍລະນີ ເຊັ່ນ:
 - ✓ ຖ້າຕ້ອງຂຽນໂປຣແກຣມຍາກໆທີ່ຄິດຄົ້ນດຽວບໍ່ອອກ ກໍ່ລອງຫາຄູ່ຂຽນໂປຣແກຣມທີ່ມີຄວາມສາມາດໃນລະດັບດຽວກັນ ຫຼື ໃຫ້ຄຽງກັນ ເພື່ອພັດປ່ຽນການຂຽນໂປຣແກຣມ ແລະ ທຳງານ.
 - ✓ ຕ້ອງການພັດທະນານັກຂຽນໂປຣແກຣມມືໃໝ່ ເພື່ອຮັບເຂົ້າທົມພັດທະນາ



ຝຶກການຂຽນໂປຣແກຣມແບບເປັນຄູ່

❑ ຂໍ້ດີຂອງການຂຽນໂປຣແກຣມແບບເປັນຄູ່

- ເຮົາຈະໄດ້ພັດທະນາທັກສະການເຮັດວຽກເປັນທີມ, ຝຶກການເວົ້າ, ອະທິບາຍໃຫ້ໝູ່ເຂົ້າໃຈວ່າກຳລັງຄົ້ນຄິດເລື່ອງຫຍັງ ແລະ ກຳລັງເຮັດຫຍັງຢູ່.
- ຊ່ວຍລຸດຄວາມຜິດພາດໃນການຂຽນໂຄດໂປຣແກຣມ ເພາະວ່າມີຄົນໜຶ່ງເປັນຜູ້ເບິ່ງ, ຕິດຕາມຄວາມຖືກຕ້ອງຂອງໂຄດທີ່ຂຽນ.
- ໄດ້ຮຽນຮູ້ເທັກນິກການຂຽນໂປຣແກຣມ ແລະ ການແກ້ໄຂປັນຫາຈາກຄູ່ ຫຼື ທີ່ມ ເຊິ່ງຈະຊ່ວຍໃຫ້ເຮົາຮຽນຮູ້ໄດ້ໄວຂຶ້ນ

Q1. ໃຫ້ຂຽນໂປຣແກຣມຮັບຄ່າລັດສະໝີຂອງຮູບວົງມົນຈາກຜູ້ໃຊ້ ແລະ ສະແດງຄ່າລວງຮອບ ແລະ ເນື້ອທີ່ຂອງຮູບວົງມົນ ທີ່ມີລັດສະໝີຕາມຄ່າດັ່ງລຸ່ມນີ້. ໃຊ້ຕົວປ່ຽນ $PI = 3.141592$.

ຜົນໄດ້ຮັບ

Enter the radius of a circle: 11

Circumference of a circle= 69.115024, Area of a circle= 280.132632

- ▶ ລວງຮອບຂອງຮູບວົງມົນຄິດໄລ່ຈາກສູດ: $2 * \text{radius} * PI$
- ▶ ເນື້ອທີ່ຂອງຮູບວົງມົນຄິດໄລ່ໄດ້ຈາກສູດ: $PI * \text{radius} * \text{radius}$

.

Q2. ໃຫ້ຂຽນໂປຣແກຣມສະແດງຄ່າຕົວເລກທີ່ຂຶ້ນກຳລັງຈາກ 2 ເຖິງ 6 ທີ່ສະແດງຜົນຮັບດັ່ງຮູບຜົນຮັບດັ່ງລຸ່ມນີ້. ໂດຍທີ່ a ສາມາດເພີ່ມຂຶ້ນຈາກ 2 ເຖິງ 6 ແລະ n ແມ່ນມີຄ່າເທົ່າກັບ 2. ຈາກນັ້ນໃຫ້ສະແດງຜົນຮັບຂອງ $a^{**}n$

ຜົນໄດ້ຮັບ	a	n	$a^{**}n$
	2	2	4
	3	2	9
	4	2	16
	5	2	25
	6	2	36