Exercises 3

≔ Week	Semana 3

Drivers License

Input

- Age
- Letter for Indication: Y, N

Output

- Yes
- No

Code

1. Input age as an int and save it in a variable
2. Input yes or no and save it in a variable
3. If…
    a. Age is higher or equal to 18 and has ID, then is 'Yes'
    b. Age is higher or equal to 18 and has no ID , then is 'Yes'
    c. Age is lower to 18 and has ID, then is 'No'.
    d. Age is lower to 18 and has no ID, then is 'No'.

Test Cases / Terminal :

- 18 | Y = YES

- 19 | N = YES

- 17 | Y = NO

- 13 | N = N

- abc | abc = Wym (Invalid data)

```
Age: 18
Do you have an ID? yes
Yes
ximyer@Yeraldins-MacBook-Air week3 % /usr/local/bin/
python3.12 /Users/ximyer/Desktop/A-TEC/week3/main.py

Age: 19
Do you have an ID? no
Yes
ximyer@Yeraldins-MacBook-Air week3 % /usr/local/bin/
python3.12 /Users/ximyer/Desktop/A-TEC/week3/main.py

Age: 17
Do you have an ID? yes
No
ximyer@Yeraldins-MacBook-Air week3 % /usr/local/bin/
python3.12 /Users/ximyer/Desktop/A-TEC/week3/main.py
```

Triangle Type

Input

- Three integers and save them in 3 variables

Output

- Equilateral
- Isosceles
- Scalene
- Not a Triangle

Test Cases / Terminal :

1. 5, 5, 5 = Equilateral
2. 4, 4, 5 = Isosceles
3. 3, 4, 5 = Scalene
4. 1, 1, 5 = Not a Triangle

TERMINAL   >_ **Python** + ∨ ⊡ 🗑 ⋯

```
/usr/local/bin/python3.12 /Users/
ximyer/Desktop/A-TEC/week3/triang
le.py
ximyer@Yeraldins-MacBook-Air week
3 % /usr/local/bin/python3.12 /Us
ers/ximyer/Desktop/A-TEC/week3/tr
iangle.py
Size 1: 5
Size 2: 5
Size 3: 5
EQUILATERAL
ximyer@Yeraldins-MacBook-Air week
3 % /usr/local/bin/python3.12 /Us
ers/ximyer/Desktop/A-TEC/week3/tr
iangle.py
Size 1: 3
Size 2: 4
Size 3: 5
SCALENE
ximyer@Yeraldins-MacBook-Air week
3 % /usr/local/bin/python3.12 /Us
ers/ximyer/Desktop/A-TEC/week3/tr
iangle.py
Size 1: 3
Size 2: 3
Size 3: 5
ISOSCELES
ximyer@Yeraldins-MacBook-Air week
3 % /usr/local/bin/python3.12 /Us
ers/ximyer/Desktop/A-TEC/week3/tr
iangle.py
Size 1: 112
Size 2: 12
Size 3: 1
NOT A TRIANGLE
ximyer@Yeraldins-MacBook-Air week
3 %
```

Greater than 3:

Input:

- 3 integers in 3 variables and save them in a dict?
- IF:
    - a > B, a > c → the greatest number is: a
    - b > a, b > c → the greatest number is: b
    - c > a, c > c → the greatest number is: b
    - else: no idea

Test Cases / Terminal :

- 1, 2, 3 → 3
- 3, 4, 2→ 4
- 5, 3, 4 → 5
- 1, 1, 2 → 2
- 1, 1, 1 → Wrong Data → Wym?

```
/usr/local/bin/python3.12 /Users/ximyer
/Desktop/A—TEC/week3/a.py
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/a.py
  Give me a number: 1
  Give me a number: 2
  Give me a number: 3
  The greatest number is:  3
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/a.py
  Give me a number: 3
  Give me a number: 4
  Give me a number: 2
  The greatest number is:  4
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/a.py
  Give me a number: 5
  Give me a number: 3
  Give me a number: 4
  The greatest number is:  5
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/a.py
  Give me a number: 1
  Give me a number: 1
  Give me a number: 1
  Idk wym
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/a.py
  Give me a number: 1
  Give me a number: 1
  Give me a number: 2
  The greatest number is:  2
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/a.py
  Give me a number: 1
  Give me a number: 1
  Give me a number: 1
  Idk wym
○ ximyer@Yeraldins—MacBook—Air week3 % ▌
```

Order 3 Numbers

Input:

- 3 numbers in 3 variables

Output:

- The 3 numbers from least to greatest

Process:

1. Input 3 numbers and save them in 3 variables:
2. IF:
3. a < b and a < c → a

Test Cases / Terminal :

- 1, 2, 3 → 1, 2, 3
- 1, 3, 2 → 1, 2, 3
- 2, 1, 3 → 1, 2, 3
- 2, 3, 1 → 1, 2,3
- 3, 1, 2 → 1, 2, 3
- 3, 2, 1 → 1, 2, 3
- 1, 1, 2, → 1, 1, 2
- 1, 1, 1 → Wym? Wrong data

```
∨ TERMINAL                   >_ Python  + ∨  ☐  🗑  ⋯

● ximyer@Yeraldins-MacBook-Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A-TEC/week3/a.py
  Number 1: 1
  Number 2: 2
  Number 3 : 3
  1
  2
  3
● ximyer@Yeraldins-MacBook-Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A-TEC/week3/a.py
  Number 1: 1
  Number 2: 3
  Number 3 : 2
  1
  2
  3
● ximyer@Yeraldins-MacBook-Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A-TEC/week3/a.py
  Number 1: 2
  Number 2: 1
  Number 3 : 3
  1
  2
  3
● ximyer@Yeraldins-MacBook-Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A-TEC/week3/a.py
  Number 1: 2
  Number 2: 3
  Number 3 : 1
  1
  2
  3
● ximyer@Yeraldins-MacBook-Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A-TEC/week3/a.py
  Number 1: 3
  Number 2: 1
  Number 3 : 2
  1
  2
  3
● ximyer@Yeraldins-MacBook-Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A-TEC/week3/a.py
  Number 1: 3
  Number 2: 2
  Number 3 : 1
  1
  2
  3
```

Body Mass Index

Input

- Weight

- Height
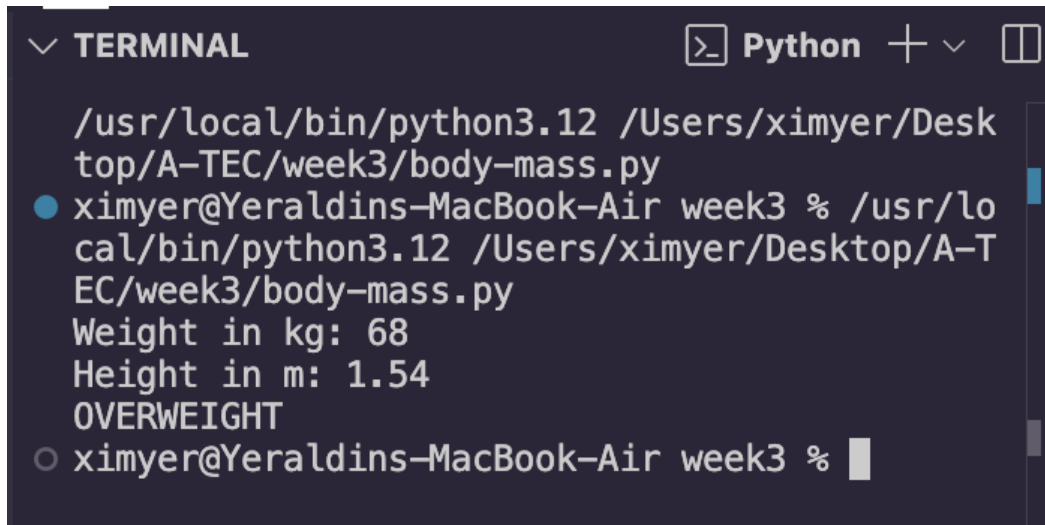
Output

- Description of IMC

Process

1. Ask for weight and save it in a variable.

2. Ask for height and save it in a variable.

3. Use the formula: index = weight / height**2

4. Based on the result of index, write the possible scenarios.

5. IF:

   a. index < 20.   'LOW WEIGHT'

   b. 20 <= index < 25   'NORMAL'

   c. 25 <= index < 30   'OVERWEIGHT'

   d. 30 <= index < 40   'OBESITY'

   e. 40 >= index   'MORBID OBESITY'

6. Print the valid scenario.

Test Cases / Terminal :

- 44, 1.56 → LOW WEIGHT

- 85, 2.0 → NORMAL

- 68, 1.54 → OVERWEIGHT

- 50, 1.10 → MORBID OBESITY

- aa, aa → Wym? Wrong data

Terminal

```
∨ TERMINAL                          >_ Python  + ∨  ⬚

  /usr/local/bin/python3.12 /Users/ximyer/Desk
  top/A-TEC/week3/body-mass.py
● ximyer@Yeraldins-MacBook-Air week3 % /usr/lo
  cal/bin/python3.12 /Users/ximyer/Desktop/A-T
  EC/week3/body-mass.py
  Weight in kg: 68
  Height in m: 1.54
  OVERWEIGHT
○ ximyer@Yeraldins-MacBook-Air week3 % ▮
```

Leap Year

Input:

- A number (year)

Output:

- True or False if it's divisible /4.

Process:

1. Input a year and save it in a variable 'year'.
2. IF:

    a. year is divisible over 4, but not 100 is TRUE

    b. else print FALSE

Test Cases / Terminal :

- 2004: LEAP

- 2005: NO

- 2006: NO

- 2007: NO

- 2008: YES

- aslidjalksd, Wym? Wrong data

```
/usr/local/bin/python3.12 /Users/ximyer/Desk
top/A-TEC/week3/leap-year.py
● ximyer@Yeraldins-MacBook-Air week3 % /usr/lo
  cal/bin/python3.12 /Users/ximyer/Desktop/A-T
  EC/week3/leap-year.py
  Year: 2004
  True
● ximyer@Yeraldins-MacBook-Air week3 % /usr/lo
  cal/bin/python3.12 /Users/ximyer/Desktop/A-T
  EC/week3/leap-year.py
  Year: 2005
  False
● ximyer@Yeraldins-MacBook-Air week3 % /usr/lo
  cal/bin/python3.12 /Users/ximyer/Desktop/A-T
  EC/week3/leap-year.py
  Year: 2006
  False
● ximyer@Yeraldins-MacBook-Air week3 % /usr/lo
  cal/bin/python3.12 /Users/ximyer/Desktop/A-T
  EC/week3/leap-year.py
  Year: 2007
  False
● ximyer@Yeraldins-MacBook-Air week3 % /usr/lo
  cal/bin/python3.12 /Users/ximyer/Desktop/A-T
  EC/week3/leap-year.py
  Year: 2008
  True
○ ximyer@Yeraldins-MacBook-Air week3 % █
```

Next Day

Input:

- Year

- Month

- Day

Output:

- Next Day (based on the date input)

Process:

1. Input year as an int and save it in a variable

2. Input month as an int and save it too

3. Input day and save it too

4. IF:

5. month == 1 and month == 3 and month == 5 and month == 7 and month == 9 and month == 11:

    a. day = 31

Test Cases / Terminal :

- 2024, 10, 31 → 2024, 11, 1

- 1959, 6, 15 → 1959, 6, 16

- 1987, 4, 30 → 1987, 5, 1

- 1872, 4, 16 → 1872, 4, 17

- 2004, 2, 28 → 2004, 2, 29

- 2005, 2, 28 → 2005, 3, 1

- 1999, 12, 31 → 2000, 1, 1

- idjasd, sldjasd, asdjasd → Wym? Wrong data

```
/usr/local/bin/python3.12 /Users/ximyer
/Desktop/A-TEC/week3/next-day.py
ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/next-day.py
Year: 2024
Month: 10
Day: 31
2024
11
1
ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/next-day.py
Year: 1959
Month: 6
Day: 15
1959
6
16
ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/next-day.py
Year: 1987
Month: 4
Day: 30
1987
5
1
ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/next-day.py
Year: 1872
Month: 4
Day: 16
1872
4
17
ximyer@Yeraldins-MacBook-Air week3 %
```

```
TERMINAL                    >_ Python + ∨  ⬚  🗑  …

/usr/local/bin/python3.12 /Users/ximyer
/Desktop/A-TEC/week3/next-day.py
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/next-day.py
Year: 2004
Month: 2
Day: 28
2004
2
29
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/next-day.py
Year: 2005
Month: 2
Day: 28
2005
3
1
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/next-day.py
Year: 1999
Month: 12
Day: 31
2000
1
1
○ ximyer@Yeraldins-MacBook-Air week3 % █
```

Rock, Paper, Scissors

Input:

- Ana's turn and save it as a variable.

- Jane's turn and save it as a variable

Output:

- Who wins

- Who loses

- It's a tie?

Process:

1. Input Ana's game and save it in a variable

2. Input Jane's game and save it in a variable.

3. IF:

    a. ana = a and jane = p → ana wins

    b. ana = a and jane = t → ana wins

    c. ana = p and jane = a → ana wins

    d. ana = p and jane = t → ana loses

    e. ana = t and jane = a → ana loses

    f. ana = t and jane = p → ana wins

Test Cases / Terminal:

- ana = a and jane = p → ana wins

- ana = a and jane = t → ana wins

- ana = p and jane = a → ana wins

- ana = p and jane = t → ana loses

- ana = t and jane = a → ana loses

- ana = t and jane = p → ana wins

```
/usr/local/bin/python3.12 /Users/ximyer
/Desktop/A-TEC/week3/a.py
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Your game, Ana? a
Your game, Juan? a
Tie
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Your game, Ana? t
Your game, Juan? p
Ana Wins!
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Your game, Ana? p
Your game, Juan? t
Juan Wins!
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Your game, Ana? p
Your game, Juan? p
Tie
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Your game, Ana? t
Your game, Juan? a
Juan Wins!
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Your game, Ana? a
Your game, Juan? t
Ana Wins!
○ ximyer@Yeraldins-MacBook-Air week3 %
```

Position of a point with respect to the circumference

Input:

- Radius of the circle
- Coordinate points of the center of the circle
- Coordinates of a point

Output:

- A string that represents the position of the point with respect to the circumference.

Process:

1. Input the numbers I'll need as *floats* and save them in variables.

2. `distance = √((x2 - x1)² + (y2 - y1)²)`

3. IF:

    a. distance < radius → Print IN

    b. distance == radius → Print ON

    c. distance < radius → Print OUTSIDE

Test Case / Terminal

```
/usr/local/bin/python3.12 /Users/ximyer
/Desktop/A-TEC/week3/radius.py
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/radius.py
Radius: 7
Coordinate x of the center of the circu
mference: 0
Coordinate y of the center of the circu
mference: 0
X-coordinate of the point: 1
Y-coordinate of the point (floating)8
OUTSIDE
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/radius.py
Radius: 5
Coordinate x of the center of the circu
mference: 0
Coordinate y of the center of the circu
mference: 1
X-coordinate of the point: 4
Y-coordinate of the point (floating)2
INSIDE
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/radius.py
Radius: 7
Coordinate x of the center of the circu
mference: 0
Coordinate y of the center of the circu
mference: 0
X-coordinate of the point: 0
Y-coordinate of the point (floating)7
ON
○ ximyer@Yeraldins-MacBook-Air week3 %
```

From Centimeters to Kilometers, Meters and Centimeters

Input:

- Distance in cm

Output:

- Equivalent in km

- Equivalent in m

- Equivalent in cm

Process:

1. Input the distance in cm as a float and save in in a variable.

2.

Test Cases / Terminal :

```
/usr/local/bin/python3.12 /Users/ximyer
/Desktop/A-TEC/week3/a.py
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Distance in cm: 100
1 m
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Distance in cm: 240005
2 km
400 m
5 cm
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Distance in cm: 67
67 cm
● ximyer@Yeraldins-MacBook-Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A-TEC/week3/a.py
Distance in cm: 300004
3 km
4 cm
○ ximyer@Yeraldins-MacBook-Air week3 %
```

Compare Dates

Input:

- Day 1
- Month 1
- Day 2
- Month 2

Output:

- Which date comes first?

Process:

1. Input the days, and months respectively and save them in variables
2. IF:
   a. `month1 < month2` :
      i. Print DATE 1 COMES FIRST
   b. `month2 < month1:`
      i. Print DATE 2 COMES FIRST
   c. `month1 == month2` :
      i. IF
         1. `day1 < day2:`
         2. PRINT DATE 1 COMES FIRST
      ii. else
      iii. Print DATE 2 COMES FIRST.

Test Cases / Terminal :

- 30,4 and 16,4 → Date 2 comes first

- 15, 6 and 28, 6 → Date 1 comes first

- 29,3 and 7,7 → Date 1 comes first

- 1,11, and 1,1 → Date 2 comes first

```
/usr/local/bin/python3.12 /Users/ximyer
/Desktop/A—TEC/week3/dates.py
● ximyer@Yeraldins—MacBook—Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A—TEC/week3/dates.py
Day 1: 30
Month 1: 4
Day 2 16
Month 2: 4
Date 2 occurs fist
● ximyer@Yeraldins—MacBook—Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A—TEC/week3/dates.py
Day 1: 15
Month 1: 6
Day 2 30
Month 2: 6
Date 1 occurs first
● ximyer@Yeraldins—MacBook—Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A—TEC/week3/dates.py
Day 1: 29
Month 1: 3
Day 2 7
Month 2: 7
Date 1 occurs first.
● ximyer@Yeraldins—MacBook—Air week3 % /u
sr/local/bin/python3.12 /Users/ximyer/D
esktop/A—TEC/week3/dates.py
Day 1: 1
Month 1: 11
Day 2 1
Month 2: 1
Date 2 occurs first
○ ximyer@Yeraldins—MacBook—Air week3 % ▮
```

Quadrant

Input:

- An integer between 0 and 360

Output:

- The location in the quadrant (if any)

Process:

1. Ask for the integer and save it in a variable.

2. IF:

   a. integer == 0 or integer == 90 or integer == 180 or integer == 270 or integer == 360 → print AXIS

   b. integer < 0 and integer > 360 → print EXCEEDS

   c. 1 < = integer < = 89 → print QUADRANT 1

   d. 91 < = integer < = 179 → print QUADRANT 2

   e. 181 < = integer < = 269 → print QUADRANT 3

   f. 271 < = integer < = 359 → print QUADRANT 4

Test Cases / Terminal :

```
∨ TERMINAL            >_ Python  + ∨  ⊓  🗑  ···

/usr/local/bin/python3.12 /Users/ximyer
/Desktop/A—TEC/week3/quadrant.py
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: —43
  Exceeds
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: 0
  Axis
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: 16
  Quadrant 1
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: 90
  Axis
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: 123
  Quadrant 2
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: 180
  Axis
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: 256
  Quadrant 3
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: 270
  Axis
● ximyer@Yeraldins—MacBook—Air week3 % /u
  sr/local/bin/python3.12 /Users/ximyer/D
  esktop/A—TEC/week3/quadrant.py
  Degrees: 306
  Quadrant 4
○ ximyer@Yeraldins—MacBook—Air week3 % █
```

Quadratic:

Input:

- Integer 1
- Integer 2
- Integer 3

Output

- The value of x1 and x2
- No Solution
- Complex roots

Process:

1. If a == 0 and b == 0:

    a. print → NO SOLUTION

2. If a = 0 and b != 0:

    a. x = (- c / b)

x1 = -b + (math.sqrt((b**2) - (4*a*c)) / (2*a)

x2 = -b - (math.sqrt((b**2) - (4*a*c)) / (2*a)

Test Cases / Terminal :

- a=1, b=2, c=3 → complex roots

- a= 0, b=1, c = 2, x= -2.0

- a=1, b=5, c=6 → x1 = 2.0, x2 = 3.0

```
⌄ TERMINAL                    >_ Python + ⌄  ⊡  🗑  ⋯

   /usr/local/bin/python3.12 /Users/ximyer
   /Desktop/A—TEC/week3/quadratic.py
 ● ximyer@Yeraldins—MacBook—Air week3 % /u
   sr/local/bin/python3.12 /Users/ximyer/D
   esktop/A—TEC/week3/quadratic.py
   a: 1
   b: 2
   c: 3
   Complex Roots
 ● ximyer@Yeraldins—MacBook—Air week3 % /u
   sr/local/bin/python3.12 /Users/ximyer/D
   esktop/A—TEC/week3/quadratic.py
   a: 0
   b: 1
   c: 2
   —2.0
 ● ximyer@Yeraldins—MacBook—Air week3 % /u
   sr/local/bin/python3.12 /Users/ximyer/D
   esktop/A—TEC/week3/quadratic.py
   a: 1
   b: 5
   c: 6
   —2.0
   —5.5
 ○ ximyer@Yeraldins—MacBook—Air week3 % ▮
```