OIM3640 - Problem Solving and Software Design

2022 Fall

Session 08 (9/22)

Today's Agenda

1. Welcome/News/Announcements
2. Review
3. Exercise Feedback
4. Lecture - **Iterations**
5. Q & A

Welcome/News/Announcements

1. **Quiz 0**
2. **Assignment 1** is posted on Canvas
    i. Please **accept** it now
    ii. Read instructions, including Code grading rubric.
3. Communications:
    i. **Email** - please specify your course number and section number in subject
    ii. Use **Slack/GitHub** when asking **code-related** questions
    iii. Please check and manage **Issues** on your GitHub repo regularly
4. Questions?

What we have learned so far...

- Variables, Expressions, Statements
- **Types**: int, float, string, boolean, `None`, other data structures
- Functions
- `turtle` module
- Conditional Statements
  - `if...elif...else`
  - recursion (**not recommended** for beginners)

Quick Quiz

1. Do the **last two lines** below have the **same** effect?

```
def polygon(t, n, length):
    ...

polygon(leo, 7, 70)
polygon(leo, length=70, n=7)
```

2. If `age` is 21, what will be printed?

```
if age >= 6:
    print("teenager")
elif age >= 18:
    print("adult")
else:
    print("kid")
```

Python Tips

1. `__main__` check

```
if __name__ == '__main__':
# Running as the main program ...
# but does not execute if loaded with import ...
```

   - Why: You should put this in all your Python scripts (YouTube)

2. To **swap** two variables:

   i. Use a temporary variable (in other programming languages)

   ii. A more **Pythonic** way: `x, y = y, x`

3. Use **comparison chain**:

```
if 18.5 <= bmi <= 30:
```

Practice

1. ⭐ **Codingbat** (OIM3640/codingbat)
2. Python Challange
3. More learning resources

Exercise Feedback

- Please check your OIM3640/**Issues** on GitHub.
  - **Reply** if we need to continue the conversation.
  - **Reply (with evidence) and Close** if you think it is fixed.
  - If you have any question regarding
    - Class **demo code**, create new issue in OIM3640/oim3640
    - Class **material**, create new issue in OIM3640/resources
- Recommendations:
  - File/folder names - use **lowercase** and **underscore**, i.e., *session04/type_demo.py*
  - Please add **docstrings** for the functions your create!
  - Please **TEST**!!
  - Please use separate files for different questions.

After (semi-)finishing exercises

- Check out my solutions in **OIM3640/oim3640**
- **Don't directly wipe out your current code**.
- **Make changes** based on it.
- Remember ***More on How to Succeed in this Course***?

Issues #0

```
def check_fermat(a, b, c, n):
    if n > 2 and a**n + b**n == c**n:
        print("Holy smokes, Fermat was wrong!")
    else:
        print("No, that doesn't work.")


def check_number():
    a = int(input("Select value of a: "))
    ...
    return check_fermat(a, b, c, n)
```

- No need to return in second function

Issues #1

```
def get_bmi_category(BMI):
    if BMI < 18.5:
        print("Underweight")
    elif BMI < 24.9:
        print("Normal weight")
    elif BMI < 29.9:
        print("Overweight")
    elif BMI > 30:
        print("Obesity")
```

- What if `BMI` is `29.95`?

Issues #2

```
def calculate_bmi (height, weight):
    """Calculate bmi through input of your height and weight."""
    height = float(input("Your height in meters: "))
    weight = float(input("Your weight in kilograms: "))
    print("Your bmi is: ", round(weight / (height * height), 2))
```

- `height` and `weight` are given!

Issues #3

```
def get_bmi_catagory(bmi):
    bmi = float(bmi)
    if bmi <= 18.5:
        print('Underweight')
    elif bmi in range[18.5,24.9]:
        print('Normal weight')
    elif bmi in range[25, 29.9]:
        print('Overweight')
    else:
        print('Obesity')
```

- Cannot use `range` this way.

Issues #4

```
...
if bmi <= 18.5:
    print("Underweight")
else:
    if bmi > 18.5 and bmi <= 24.9:
        print("Normal Weight")
    else:
        if bmi > 24.9 and bmi <= 29.9:
            print("Overweight")
        else:
            print("Obesity")
```

- Using too many layers is not recommended.

Issues #5

```
def get_bmi_category():
    weight = float(input("Input weight"))
    height = float(input("Input height in inches"))
    if calculate_bmi(weight, height) < 18.5:
        print("Underweight")
    elif calculate_bmi(weight, height) >= 18.5 and calculate_bmi(weight, height) < 25:
        print("Normal weight")
    elif calculate_bmi(weight, height) >= 25 and calculate_bmi(weight, height) < 30:
        print("Overweight")
    elif calculate_bmi(weight, height) >= 30:
        print("Obese")
```

- Function `calculate_bmi(weight, height)` might be executed four times in worst case - which is not ideal if the function costs a lot of time/computing.

Issues #6

Why is the code below not a good design?

```
def calculate_bmi(weight, height):
    bmi = 703 * (weight/height**2)
    if bmi <= 18.5:
        return "You are underweight"
    elif bmi > 18.5 and bmi <= 25:
        return "You are normal weight"
    elif bmi > 25 and bmi <= 29.9:
        return "You are overweight"
    else:
        return "You are obese!"

def start_bmi(weight, height):
    response = calculate_bmi(weight, height)
    print(response)
```

- The second function is actually (almost) just repeating the first function.
- Only need to return `bmi` value in first function.

Session 08

- Iterations