

1. give a complete description of AirPurifier class

- what should be instance variables? → ตัวแปรที่ทุกตัวมี
- what should be instance methods? → methods ที่ใช้บ่อย
- what should be class variables?
- what should be class methods?

2. for each method proposed above, how do you plan to implement it?

- give an overview of the computation
 - what's needed as inputs?
 - what's to be returned?

ออกแบบ

Instance variables :

- model name (string)
- serial number (string)
- current state (boolean); ดูว่าตอนนี้เปิดหรือปิด
- power (boolean); ดูว่าเสียบปลั๊กไฟแล้ว

Instance methods :

- # () ไม่รับ Input อย่างแรก
- # () รับ Input
- # return มี return

Public :

- about_me () ; return ชื่อ model และ serial
- check_plug (); เสียบปลั๊กไฟหรือไม่เสียบ
- toggle_switch (); กดปุ่มเปิดปิดเครื่อง

Protected :

- power_set (bool) ; ให้เปิดหรือปิด power
- do_nothing (); ไม่ทำอะไร
- open (); เปิดเครื่อง
- close (); ปิดเครื่อง

abstract :

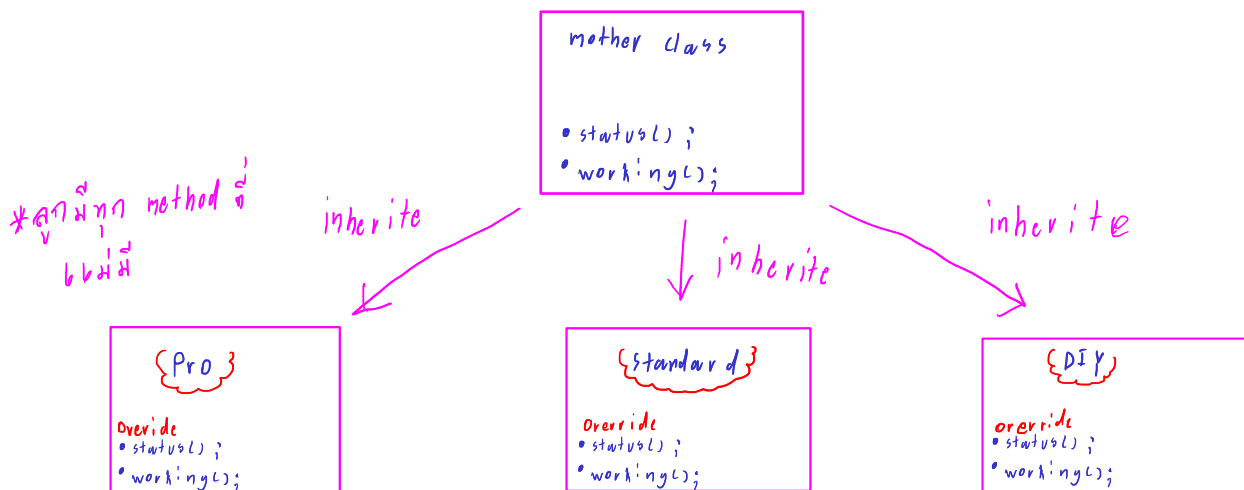
- status (); ดูว่าเครื่องทำงานหรือไม่
- working (); ดูว่าเครื่องทำงานหรือไม่

```

private String model_name, serial_number;
5 usages
protected boolean current_state;
3 usages
protected boolean power;

public AirPurifier(String model, String serial_number) ...
public String about_me() ...
public void check_plug() ...
public void toggle_switch() ...
protected void power_set(boolean power) ...
protected void do_nothing() ...
protected void open() ...
protected void close() ...
protected void clear_terminal(){System.out.print("\033[H\033[2J");}
abstract protected void status();
abstract protected void working();
  
```

set ชื่อตัวแปร และ set serial



DIY

ใช้ ตัวแปรจากแบบ

Override :

status(); ไม่กดปุ่มเปิด

working(); กดปุ่ม 2 ครั้ง เปิดทั้งปี

Standard

Instance variables (ตัวแปรจากแบบ):

dust value (int); ค่าฝุ่นที่วัด

Instance methods ;

fan speed (int); ความเร็วพัดลม

เปิดพัดลม

feel lot of dust (); การสัมผัส sensor ฝุ่น
ในชั้นใต้ หุ่น ฝุ่นเยอะ

Override :

status(); กดปุ่มกด serial number และ หน้าจอ

working(); กดปุ่ม set ค่าฝุ่น

```
private int dust_value = 0;
private String fan_speed(int dust_value) ...
private void feel_lot_of_dust() ...
public Standard_Version(String Model, String Serial_Number) ...
@Override
protected void status() ...
@Override
protected void working() ...
```

Pro

Instance variables (ตัวแปรจากแบบ):

- mode have (string[]); เก็บโหมดทั้งหมดที่มี
- mode (string); เก็บโหมดปัจจุบัน
- dust value (int); ค่าฝุ่นปัจจุบัน

Instance method :

- fan speed (string); ใช้ดูว่า mode ปัจจุบัน
เป็นโหมดไหนแล้ว
ควร ปรับพัดลมเท่าไร
- fan speed (int); ปรับความเร็วพัดลม ตาม
ฝุ่นที่มี (ใช้ใน mode auto)
- set mode (); เปลี่ยนโหมด
- feel lot of dust (); sensor ฝุ่น ที่ทำงาน
ได้ mode auto

```
private String mode_have[] = {"Silent", "Auto", "MAX_SPEED"};
private String mode = mode_have[1];

private int dust_value;
public Pro_version(String model, String serial_number) ...

@Override
protected void working() ...

@Override
protected void status() ...

private String fan_speed(String mode, int dust_value = 0)
{
    if(mode.equals(mode_have[0])) return;
}

private String fan_speed(String mode) ...
private String fan_speed(int dust_value) //use in auto mode ...
private void set_mode() ...
private void feel_lot_of_dust() ...
```

ที่บ้านมี class variable , class method

mother class

มีทุกอันแบบฉบับ

class variable :

- count (int) ; นับจำนวน obj ที่สร้าง
ไปกี่ตัว

```
private static int count = 0;
```

class method :

- count method() ; บั๊ว count
- return count() ; count getter

```
private static void count_method()  
public static int return_count()
```