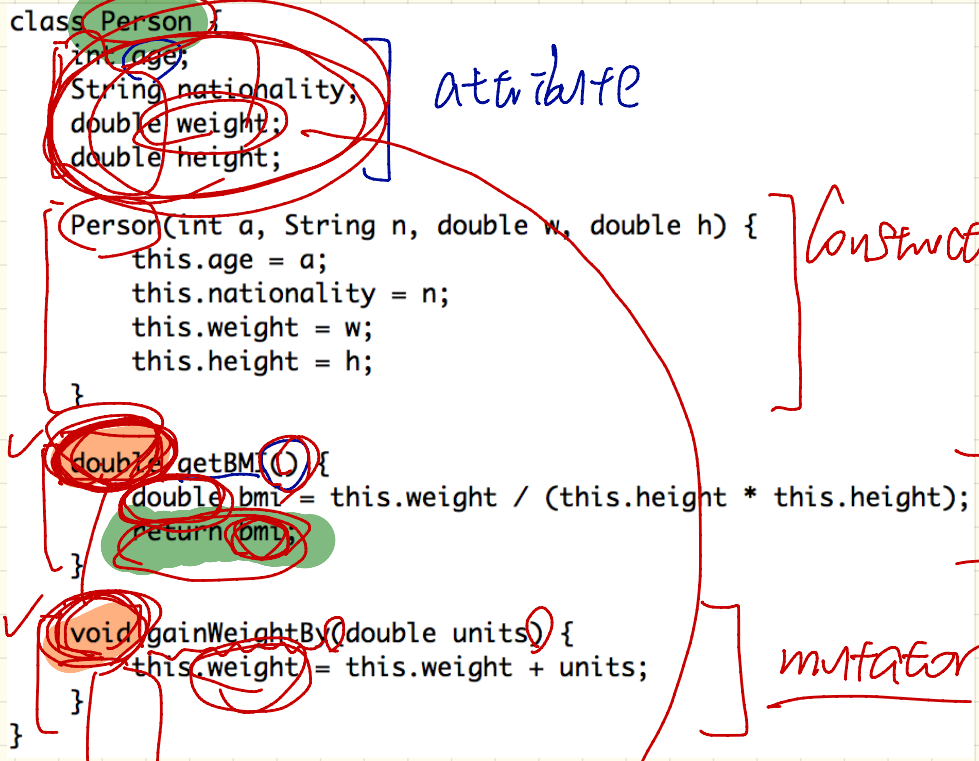
Monday, Feb. 5

Lecture 5


class Person {
int age;
String nationality;
double weight;
double height;
Person(int a, String n, double w, double h) {
this.age = a;
this.nationality = n;
this.weight = w;
this.height = h;
}
double getBMI() {
double bmi = this.weight / (this.height * this.height);
return bmi;
}
void gainWeightBy(double units) {
this.weight = this.weight + units;
}
}
attribute
Constructor
accessor
(return required)
mutator
return type
change attribute values

boolean pork(char c) {

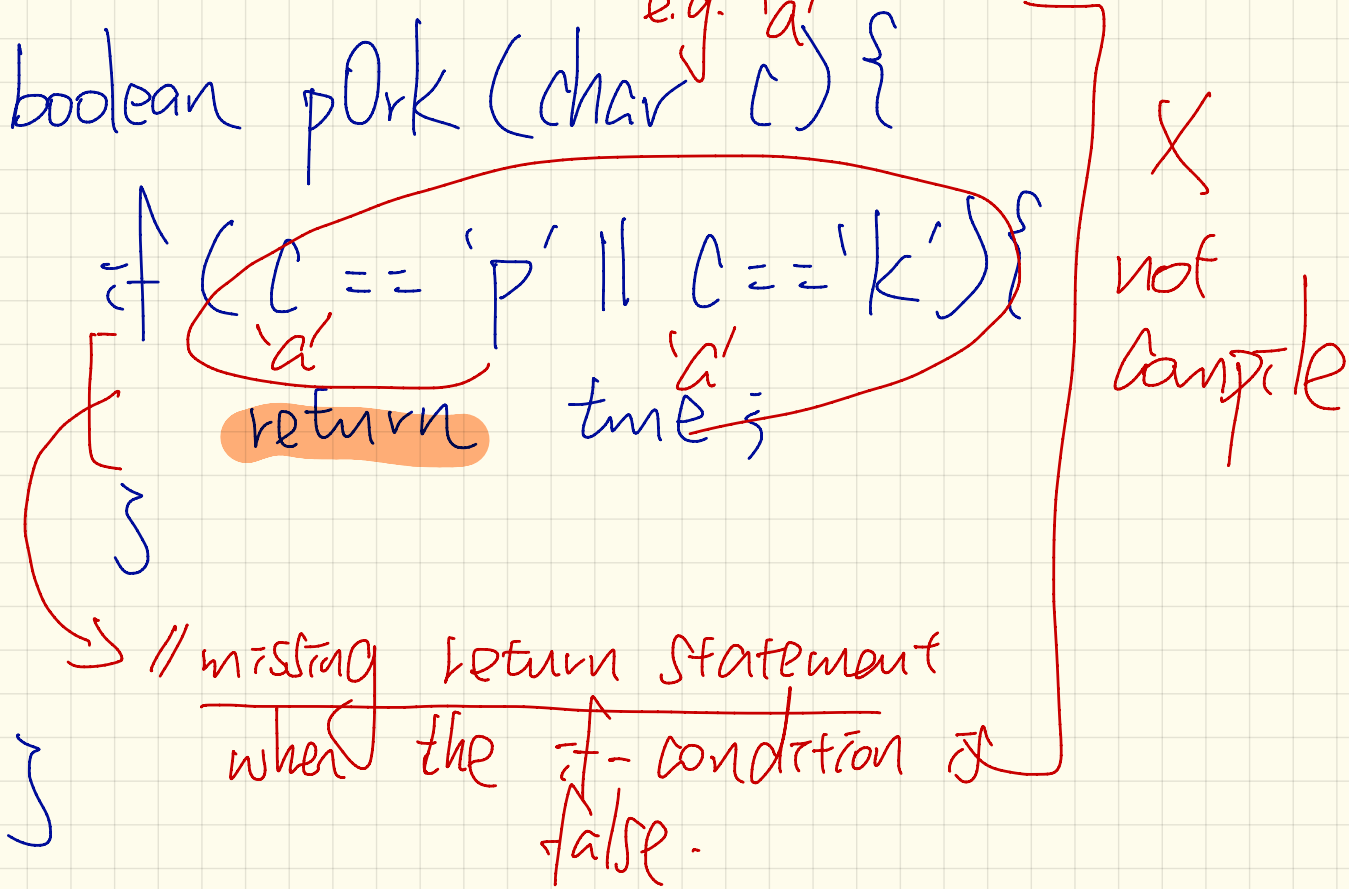
accesso

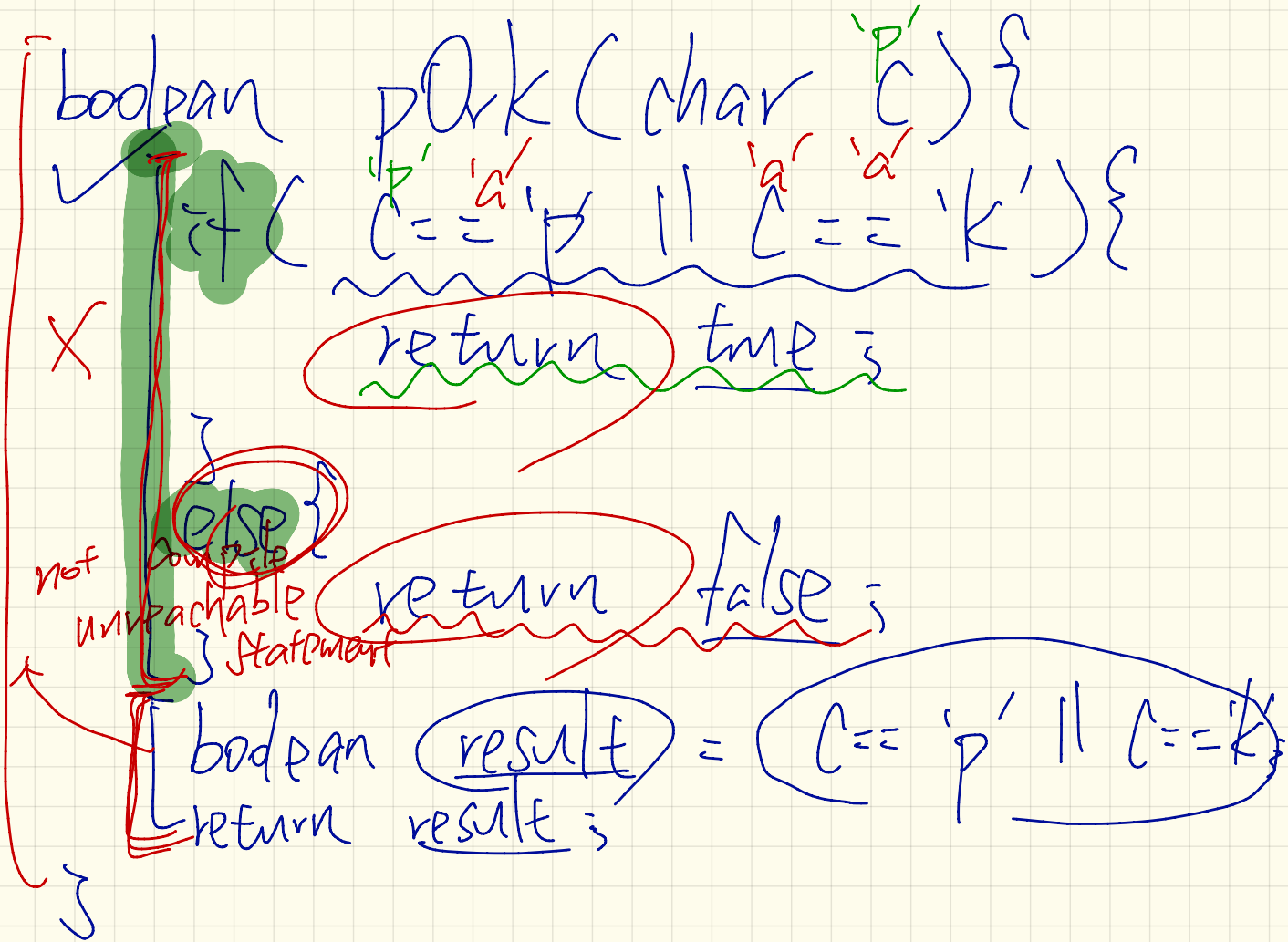
```
    boolean result = false;
    if( c == 'p' || c == 'k'){
        result = true;
    }
    return result;
}
```

X

not compile

; missing return statement


e.g. 'a'
boolean pOrk (char c) {
if (c == 'p' || c == 'k') {
'a'
'a'
return true;
}
// missing return statement
when the if-condition is false.
}
X
not compile


boolean pOrk(char c) {
if (c == 'p' || c == 'k') {
return true;
}
else {
return false;
}
boolean result = (c == 'p' || c == 'k');
return result;
}
not unreachable statement

```
class Person {
    int age;
    String nationality;
    double weight;
    double height;

    Person(int a, String n, double w, double h) {
        this.age = a;
        this.nationality = n;
        this.weight = w;
        this.height = h;
    }

    double getBMI() {
        double bmi = this.weight / (this.height * this.height);
        return bmi;
    }

    void gainWeightBy(double units) {
        this.weight = this.weight + units;
    }
}
```

this.age = 50

"British"

jim

context object (this)

Person

a	50
n	"British"
w	80
h	1.8

this.age

this.n

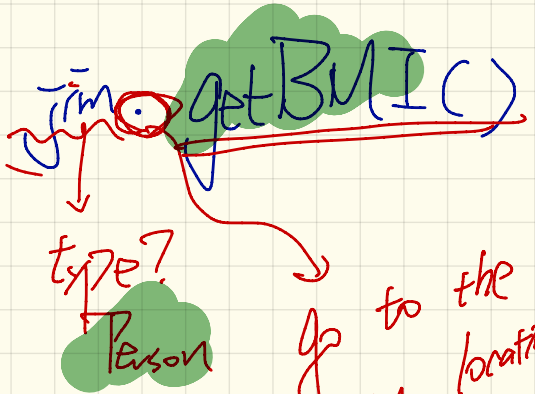
type of jim is Person

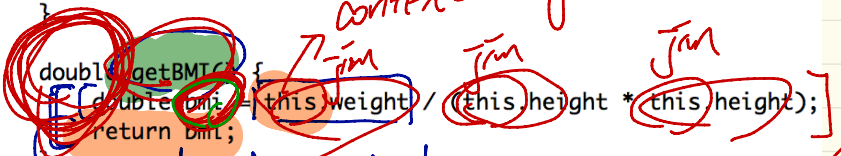
Person jim = new Person(50, "British", 80, 1.8);

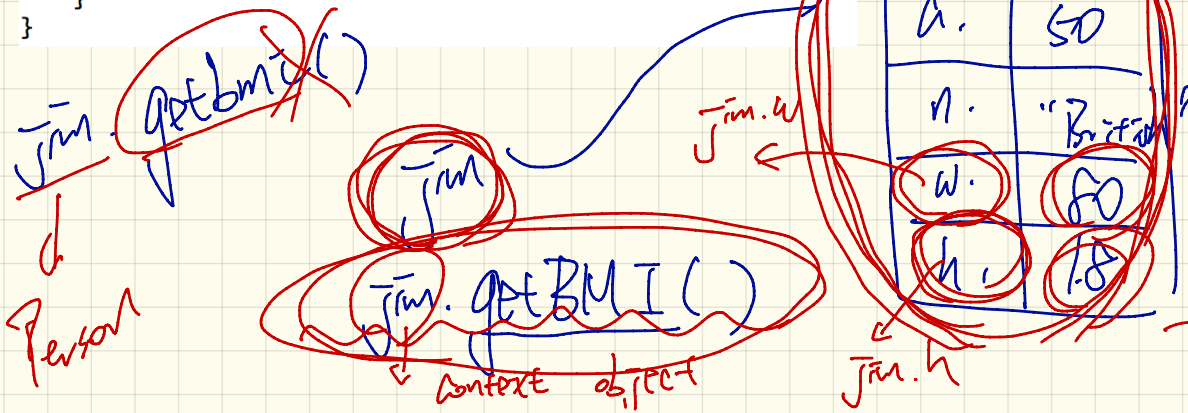
class name

variable → store address of Jim Person object

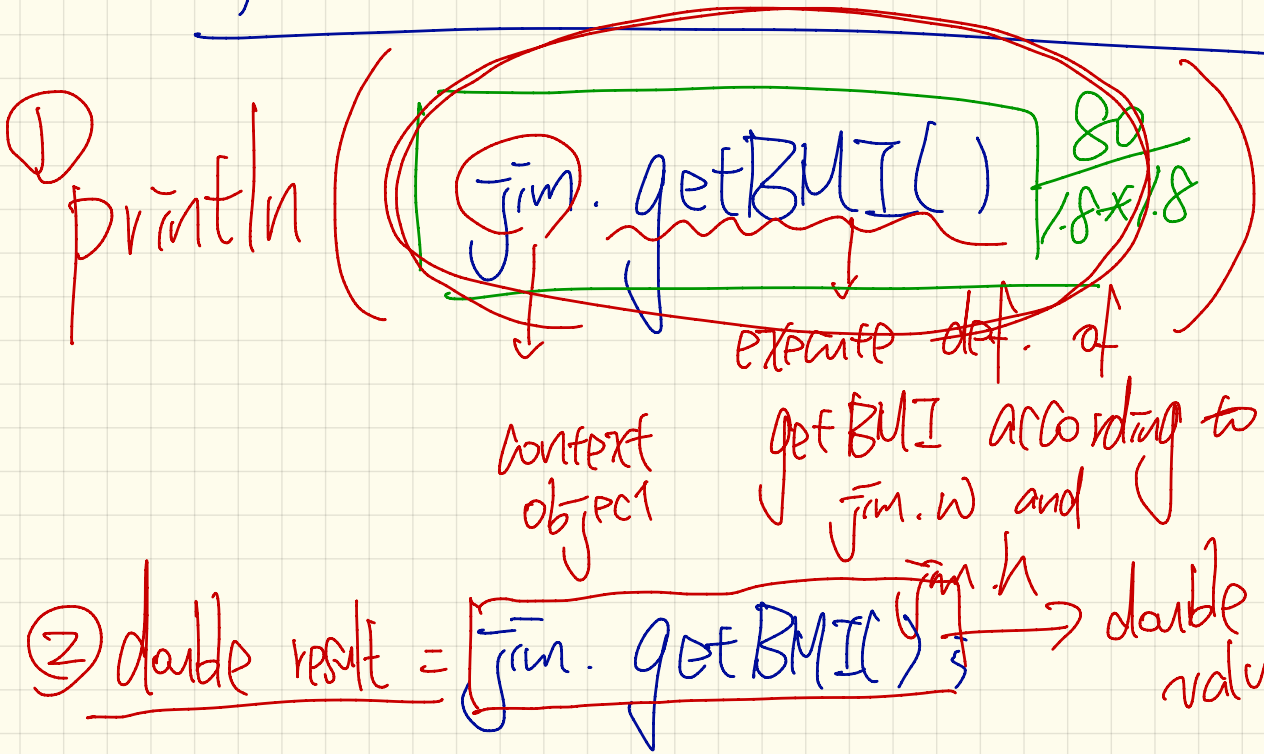
```
class Person {
    int age;
    String nationality;
    double weight;
    double height;

    Person(int a, String n, double w, double h) {
        this.age = a;
        this.nationality = n;
        this.weight = w;
        this.height = h;
    }

    double getBMI() {
        double bmi = this.weight / (this.height * this.height);
        return bmi;
    }
    void gainWeightBy(double units) {
        this.weight = this.weight + units;
    }
}
```



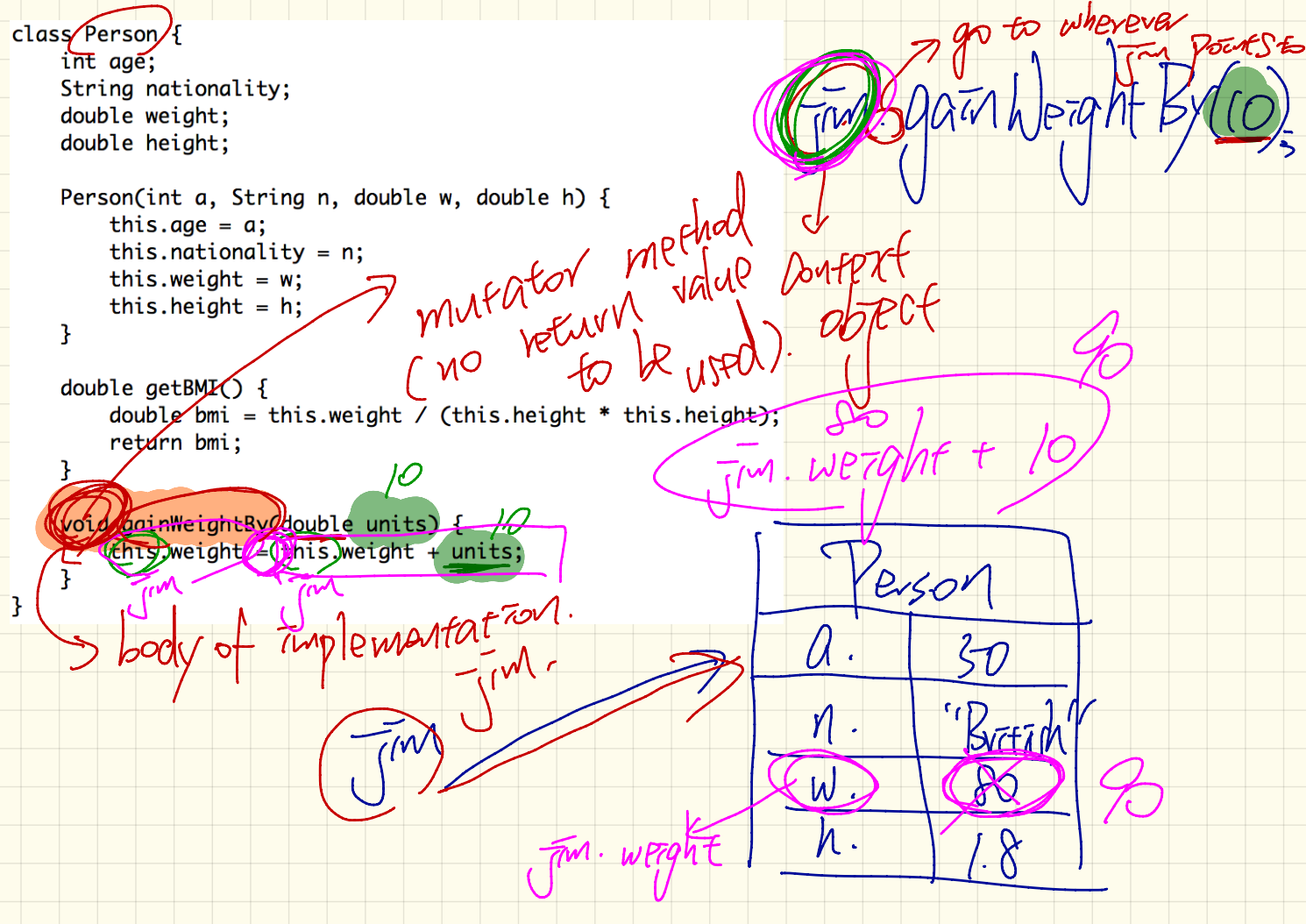




Store/Catch return value from an accessor method call



```
class Person {
    int age;
    String nationality;
    double weight;
    double height;

    Person(int a, String n, double w, double h) {
        this.age = a;
        this.nationality = n;
        this.weight = w;
        this.height = h;
    }

    double getBMI() {
        double bmi = this.weight / (this.height * this.height);
        return bmi;
    }

    void gainWeightBy(double units) {
        this.weight = this.weight + units;
    }
}
```



accessor method call

```
println( jim.getBMI() );
double result = jim.getBMI();
```

mutator method call

```
println( jim.gainWeightBy(10) );
```

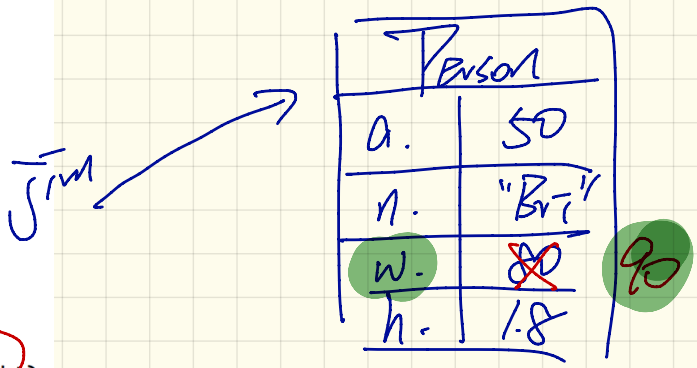
does not compile

```
double result = jim.gainWeightBy(10);
```

void

void

```
class Person {
    int age;
    String nationality;
    double weight;
    double height;

    Person(int a, String n, double w, double h) {
        this.age = a;
        this.nationality = n;
        this.weight = w;
        this.height = h;
    }

    double getBMI() {
        double bmi = this.weight / (this.height * this.height);
        return bmi;
    }

    void gainWeightBy(double units) {
        this.weight = this.weight + units;
    }
}
```

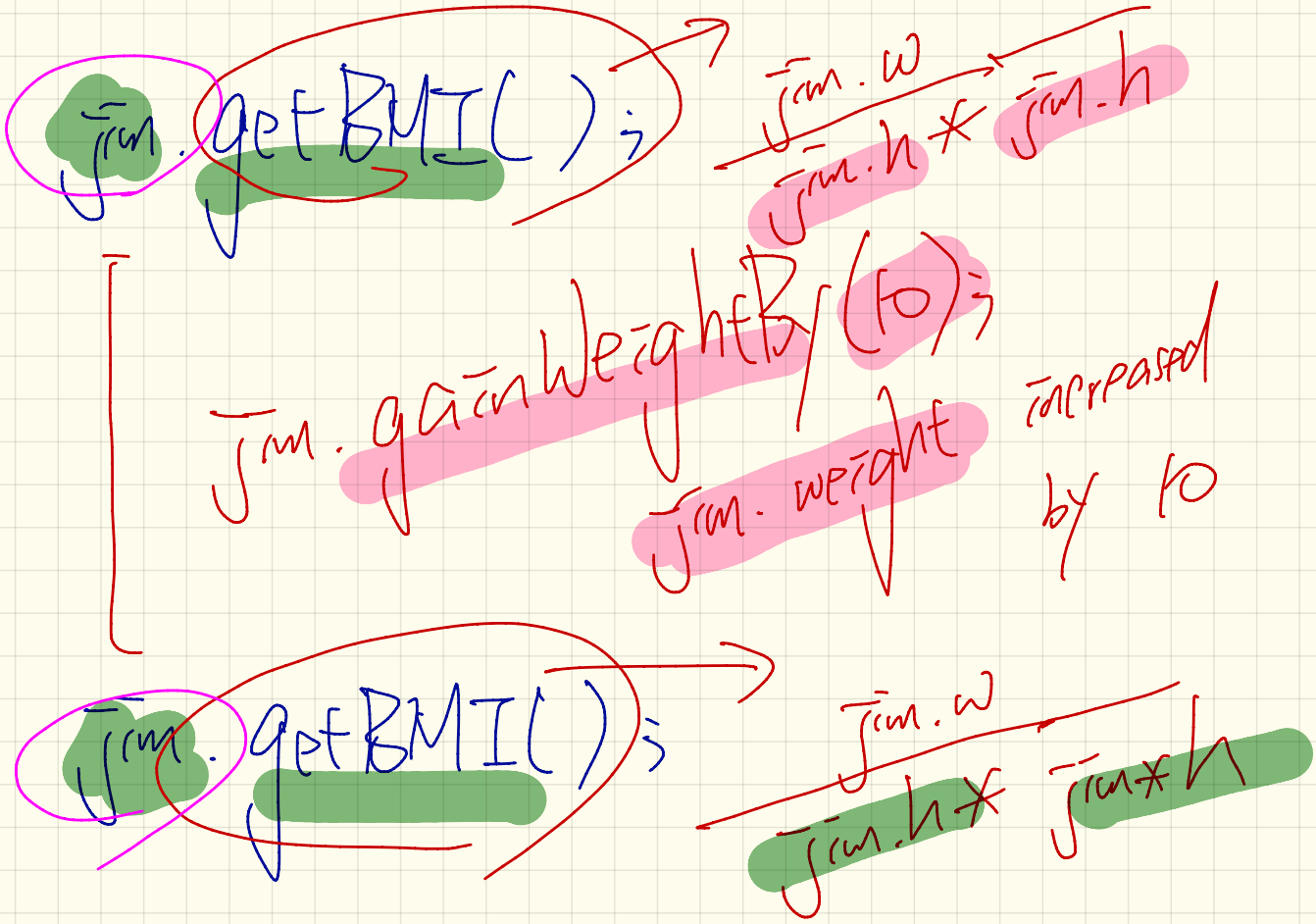


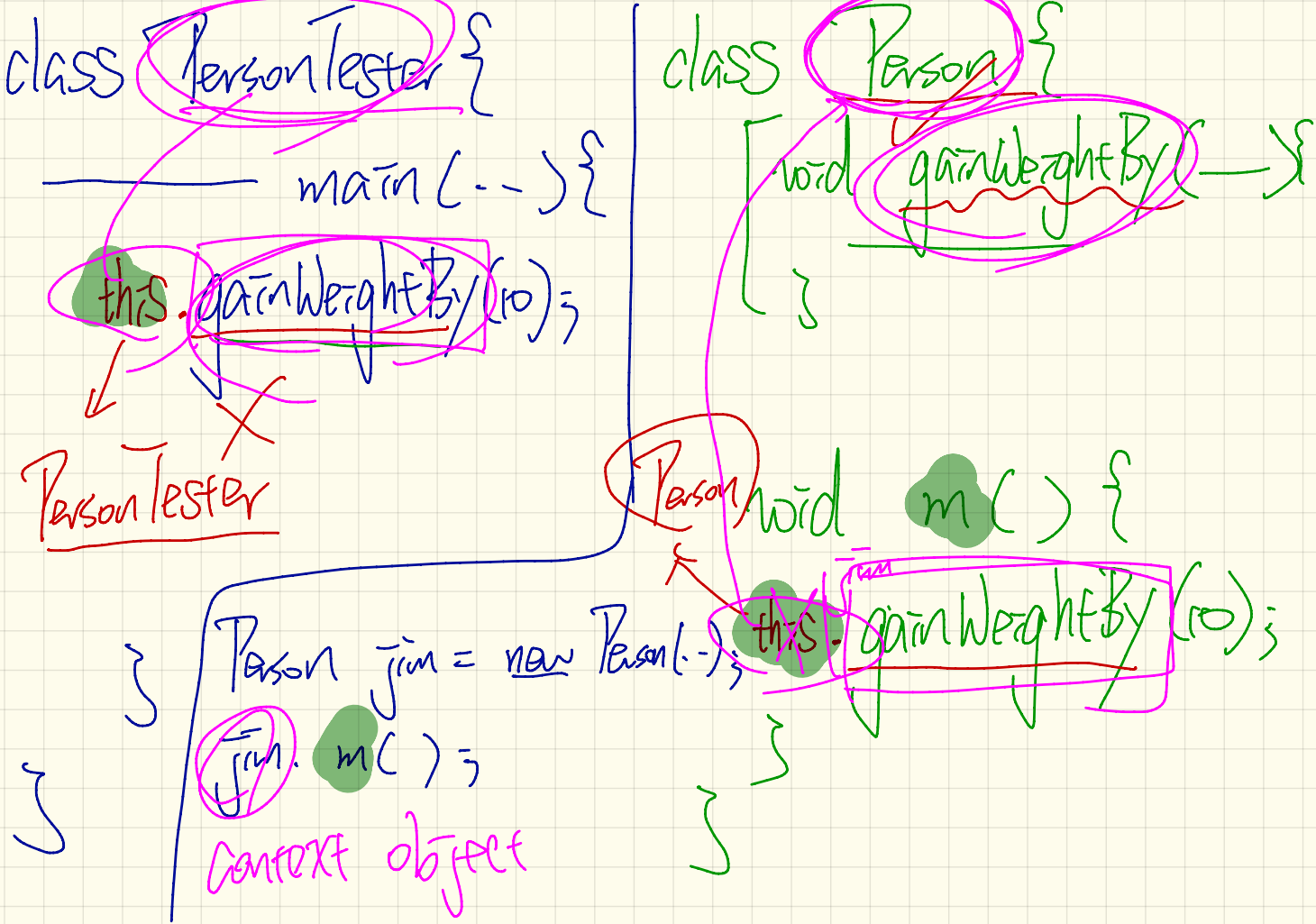
① Person jim = new Person(50, "Bri", 80, 1.8);

② println(jim.getBMI()); 80/1.8²

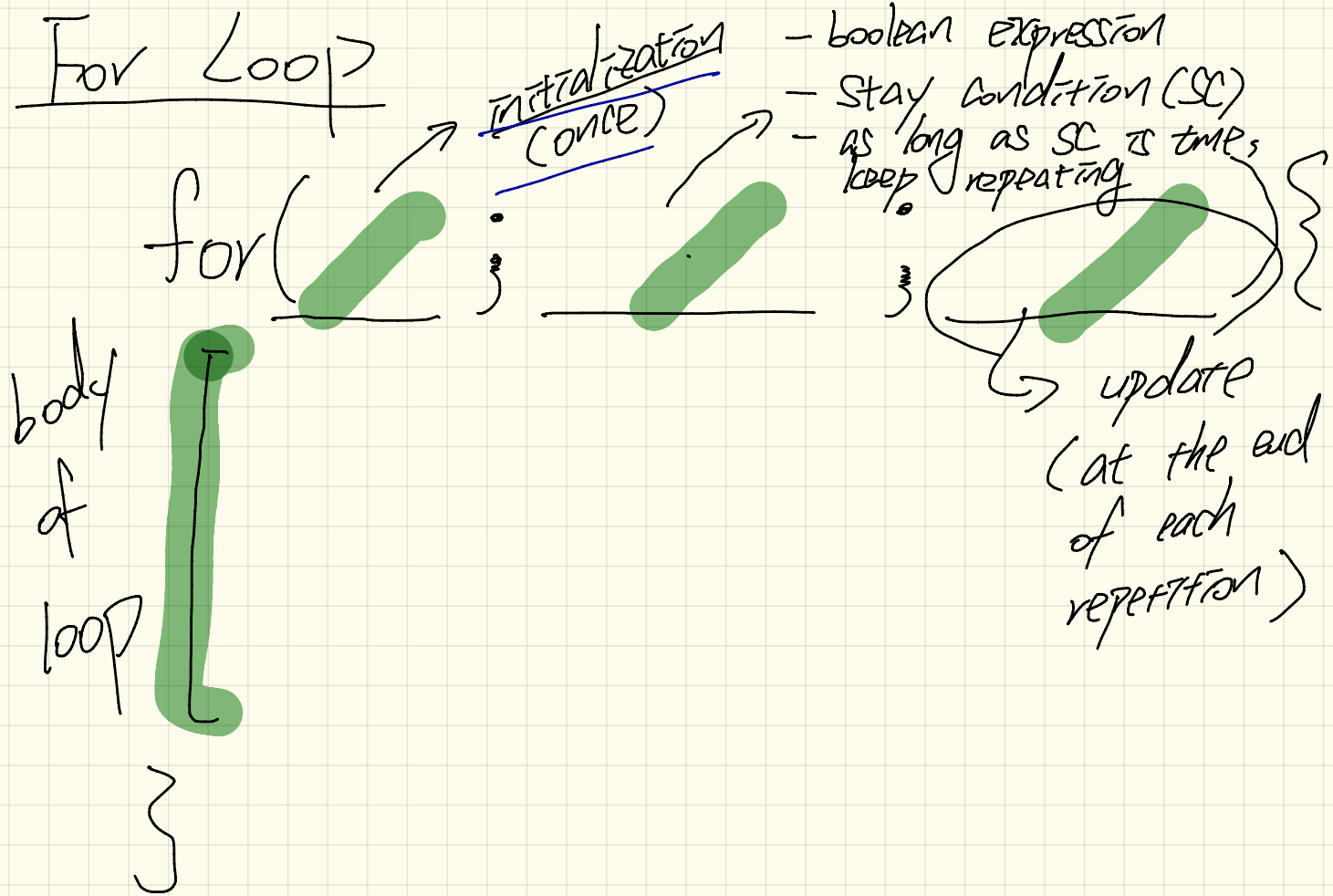
③ jim.gainWeightBy(10);

④ println(jim.getBMI()); 90/1.8²


jim.getBMI();
jim.w
jim.h * jim.h
jim.gainWeightBy(10);
jim.weight
increased by 10
jim.getBMI();
jim.w
jim.h * jim.h


class PersonTester {
main(--){
this.gainWeightBy(10);
PersonTester
Person jim = new Person(--);
jim.m();
context object
class Person {
void gainWeightBy(--){
}
Person void m(){
this.gainWeightBy(10);

For Loop

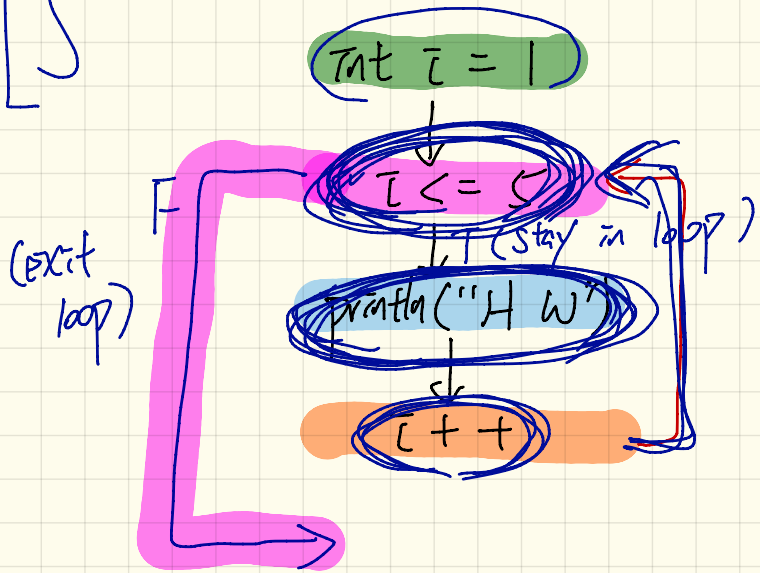

initialization
(once)
- boolean expression
- Stay condition (SC)
- as long as SC is true, loop repeating
for(; ;)
update
(at the end of each repetition)
body of loop
{
}

```
for (int i = 1; i <= 5; i++) {
    println("Hello World");
}
```

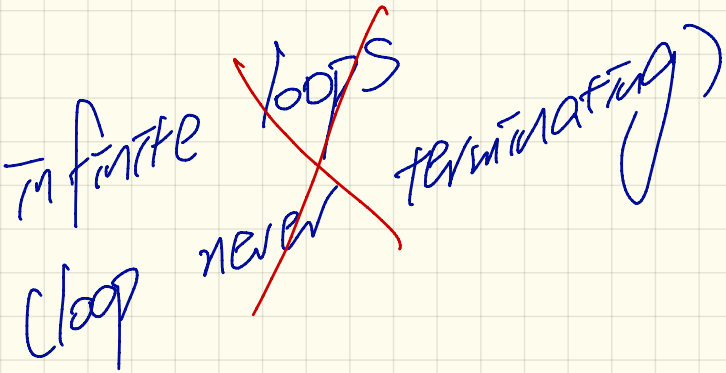
loop counter

stay condition

update



i	i<=5	action
1	T	print i++
2	T	print i++
3	T	print i++
4	T	print i++
5	T	print i++
6	F	


infinite loops
never terminating
(loop

①
```
for ( int i = 1 ; i <= 5 ; i++ ) {
    println ("H W");
}
```

②
```
int i = 1;
for ( ; i <= 5 ; i++ ) {
    println ("H W");
}
```

③

```
int i = 1;
for ( ; i <= 5; ) {
    println("Hi");
    i++;
}
```

① ≡ ② ≡ ③

④

```
int i = 1;
for( ; i <= 5; ){
    i++;
    println("Hello");
}
```

④ ≠ ③

```
for(int i=0; i<100; i++){
    println("Welcome");
}
```

Q1. How many times println("Welcome"); will be executed?

Q2. How many times i<100 will be checked?