

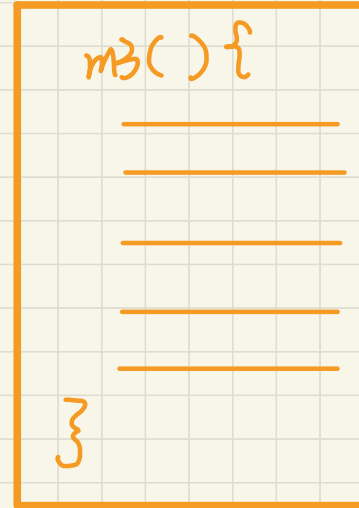
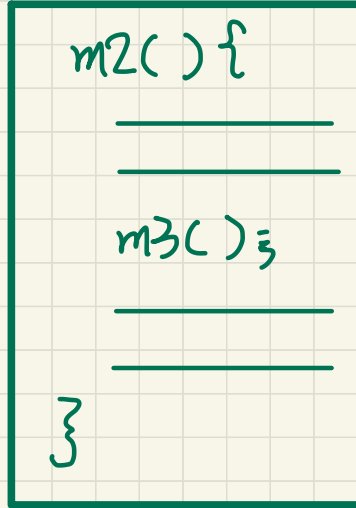
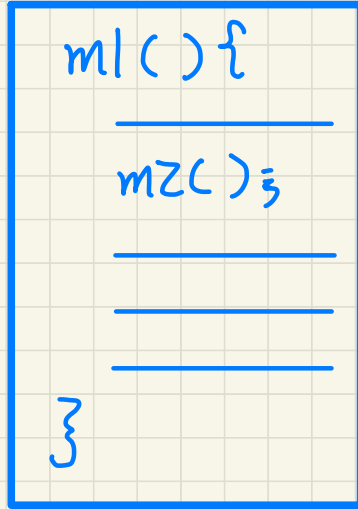
Caller vs. Callee

- **caller** is the **client** using the service provided by another method.
- **callee** is the **supplier** providing the service to another method.

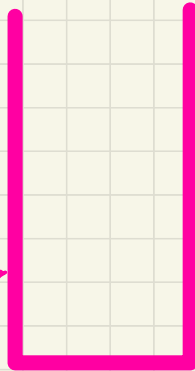
```
class C1 {  
    void m1() {  
        C2 o = new C2();  
        o.m2(); /* static type of o is C2 */  
    }  
}
```

Q: Can a method be a **caller** and a **callee** simultaneously?

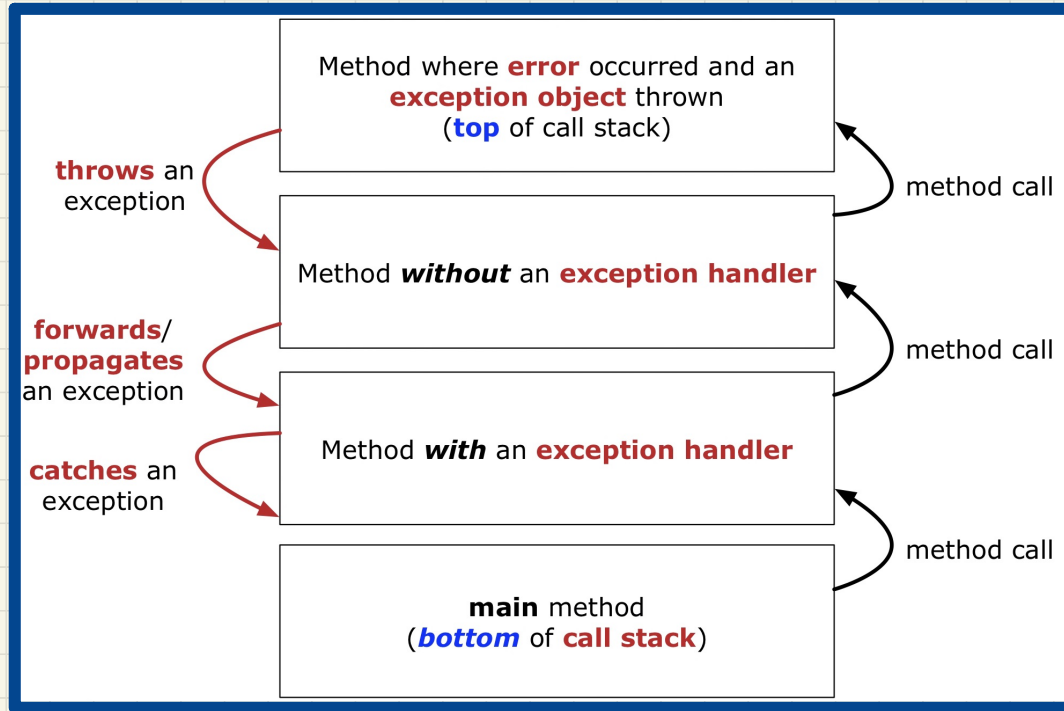
Visualizing a Call Chain using a Stack



call stack



What to Do When an Exception is Thrown: Call Stack



Catch-or-Specify Requirement

The “Catch” Solution: A `try` statement that *catches* and *handles* the *exception* (without propagating that exception to the method's *caller*).

```
main(...) {  
    Circle c = new Circle();  
    try {  
        c.setRadius(-10);  
    }  
    catch (NegativeRadiusException e) {  
        ...  
    }  
}
```

The “Specify” Solution: A method that specifies as part of its *header* that it may (or may not) *throw* the *exception* (which will be thrown to the method's *caller* for handling).

```
class Bank {  
    Account[] accounts; /* attribute */  
    void withdraw (double amount)  
        throws InvalidTransactionException {  
        ...  
        accounts[i].withdraw(amount);  
        ...  
    }  
}
```

Example: To Handle or Not To Handle?

context	caller	callee

```
class A {  
    ma(int i) {  
        if(i < 0) { /* Error */ }  
        else { /* Do something. */ }  
    } }  

```

```
class B {  
    mb(int i) {  
        A oa = new A();  
        oa.ma(i); /* Error occurs if i < 0 */  
    } }  

```

```
class Tester {  
    public static void main(String[] args) {  
        Scanner input = new Scanner(System.in);  
        int i = input.nextInt();  
        B ob = new B();  
        ob.mb(i); /* Where can the error be handled? */  
    } }  

```

```
class NegValException extends Exception {  
    NegValException(String s) { super(s); }  
}
```

Version 1:

Handle it in B.mb

Version 2:

Pass it from B.mb and handle it in Tester.main

Version 3:

Pass it from B.mb, then from Tester.main, then throw it to the console.

call
stack

