OOP:

①

Flow of control

```c
#include <stdio.h>

// function declaration
void printSquare(int);
int getSquare(int);
int get10(void);

int main(void) {
    // variable declaration
    int num;

    // function call
    num = get10();
    printSquare(num);

    return 0;
}

// function definition
void printSquare(int num) {
    int square = getSquare(num);
    printf("Square: %d\n", square);
}

int get10(void) {
    return 10;
}

int getSquare(int num) {
    return num * num;
}
```

End of Program

dyclassroom.co
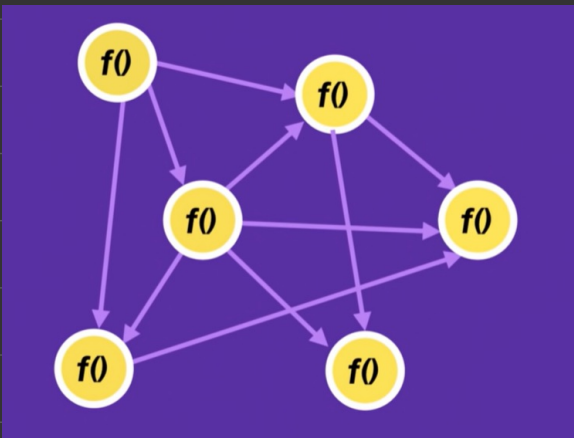
So many function calls

Imagine if there are 100s of functions in a pgm ⇒ then

overall code will be __insecure__, __unstructured__ & difficult to change

We will prefer to save data related to one entity together.

∴ we can think of some <u>User Defined datatype.</u>
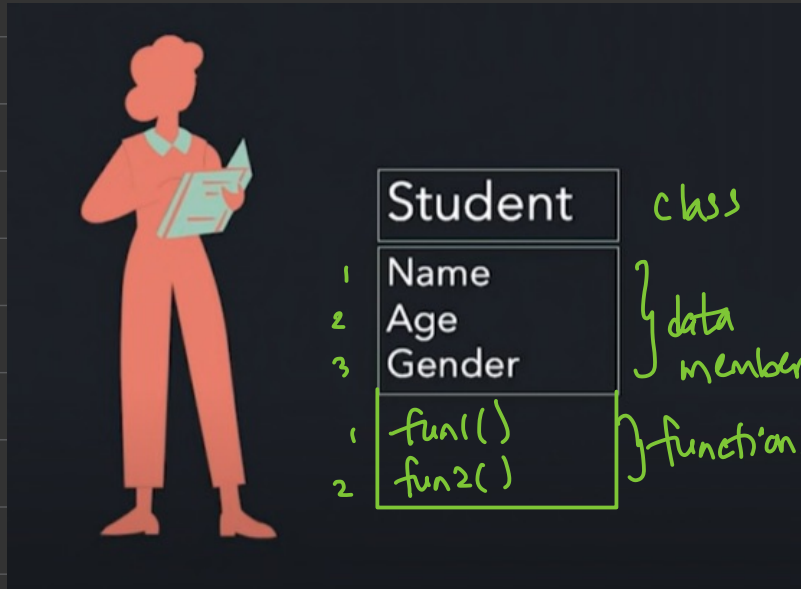
Why OOPs invented?

1960 → S/w Crisis

OOP is a programming paradigm / methodology which is more resilient,
looks more simple for outside world,
is more flexible for future changes
& can be modified as we move ahead with pgm.

How is OOP implemented?
→ Using Classes & Objects

What is a class?

→ A __class__ is a __User Defined datatype__ used to wrap __Some data members__ & __functions/methods__.

To access properties of Student class, you need to create a variable of type class.
These variables are called __Objects__.

```java
Rectangle.java > ...
1    public class Rectangle {          ← Class Name
2
3        double length;                ← Data attributes (or Variables)
4        double breadth;
5
6        void calculateArea(){          ← Member Functions (or Methods)
7            double area = length * breadth;
8            System.out.println("The area of Rectangle is: " +area);
9        }
10   }
```
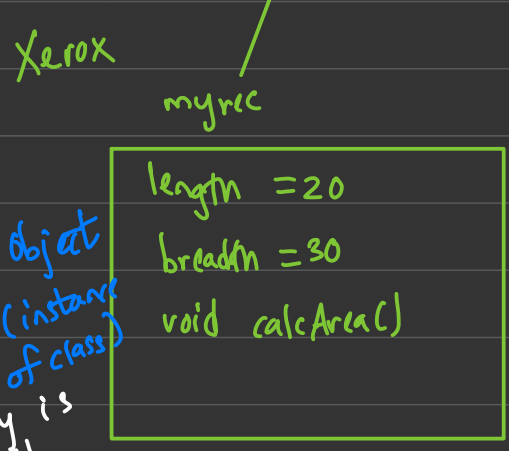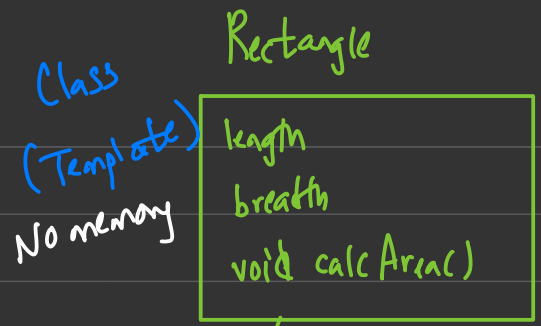
```java
Demo.java > ...
1    public class Demo{
     Run | Debug
2        public static void main(String args[]) {
3            Rectangle myrec = new Rectangle(); //first object created
4            myrec.length = 20;                 ← Object of class Rectangle
5            myrec.breadth = 30;
6            myrec.calculateArea();
7
8        }                    ← Using object to access variables of
9    }                          methods of class Rectangle
10
```

```java
// Rectangle.java
public class Rectangle {        // Class Name

    double length;              // Data attributes (or Variables)
    double breadth;

    void calculateArea() {      // Member Functions (or Methods)
        double area = length * breadth;
        System.out.println("The area of Rectangle is: " +area);
    }
}
```

```java
// Demo.java
public class Demo{

    public static void main(String args[]) {
        Rectangle myrec = new Rectangle(); //first object created
        myrec.length = 20;          // Object of class Rectangle
        myrec.breadth = 30;
        myrec.calculateArea();      // Using object to access variables of
                                    // methods of class Rectangle

    }
}
```

o/p: The area of Rectangle is 600.0

Handwritten notes (right side):

Rectangle

Class (Template) — No memory

- length
- breadth
- void calc Area()

Xerox

myrec

Object (instance of class) — Memory is allocated
- length = 20
- breadth = 30
- void calcArea()

```java
// Main.java  -- NS 16 May 2023
{

    public static void main(String[] args) {
        Rectangle myrec=new Rectangle();
        // System.out.println("Length and breadth of myrec are "+myrec.length+" "+myrec.breadth);
        myrec.length=20;
        myrec.breadth=30;
        System.out.println("Length and breadth of myrec are "+myrec.length+" "+myrec.breadth);
        myrec.calculateArea();

        Rectangle o1=new Rectangle();
        // System.out.println("Length and breadth of myrec are "+myrec.length+" "+myrec.breadth);
        o1.length=19;
        o1.breadth=1.219;
        System.out.println("Length and breadth of myrec are "+o1.length+" "+o1.breadth);
        o1.calculateArea();
    }
}
```

TERMINAL

```
Length and breadth of myrec are 20.0 30.0
The area of Rectangle is 600.0
Length and breadth of myrec are 19.0 1.219
The area of Rectangle is 23.161
(base) ingledarshan@192 NS 16 May 2023 %
```

Handwritten notes (right side):

Rectangle
- length
- breadth
- calc Area

Xerox

myrec
- length = 20
- breadth = 30
- calcArea()

o1
- length = 19
- breadth = 1.219
- calcArea