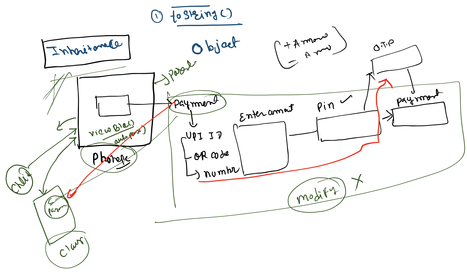
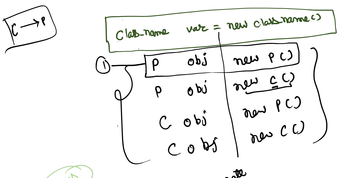
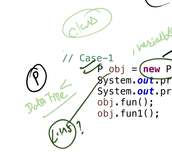
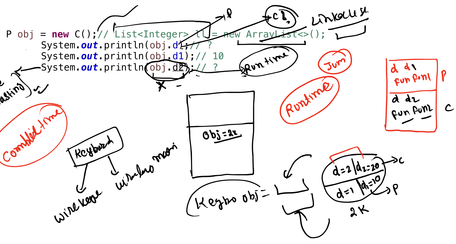


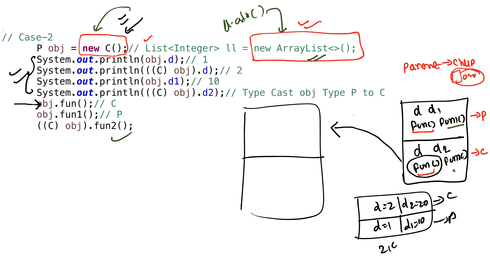



```
// Case-1
    P obj = new P();
    System.out.println(obj.d);// 1
    System.out.println(obj.d1);// 10
    obj.fun();
    obj.fun1();
```

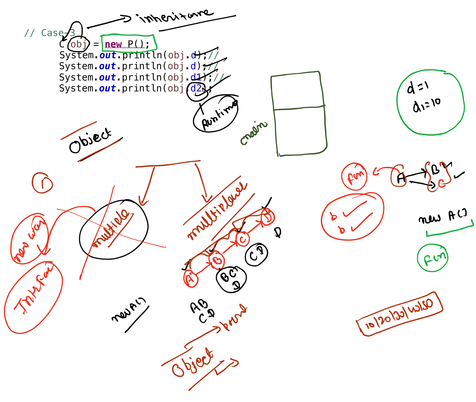





```
// Case-2
    P obj = new C();// List<Integer> ll = new ArrayList<>();
    System.out.println(obj.d);// 1
    System.out.println(((C) obj).d);// 2
    System.out.println(obj.d1);// 10
    System.out.println(((C) obj).d2);// Type Cast obj Type P to C
    obj.fun();// C
    obj.fun1();// P
    ((C) obj).fun2();
```

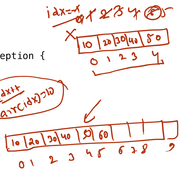



```
// Case-3
    C obj = new P();
    System.out.println(obj.d);//
    System.out.println(obj.d);//
    System.out.println(obj.d1);//
    System.out.println(obj.d2);//
```



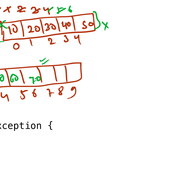

```
@Override
    public void push(int item) {
    }

    public static void main(String[] args) throws Exception {
        Stack st = new DynamicStack();// push
        // DynamicStack st1 = new DynamicStack();
        st.push(10);
        st.push(20);
        st.push(30);
        st.push(40);
        st.push(50);
        st.push(60);
        st.push(70);
    }
```




```
public void push(int item) {
        if (isFull()) {
            int[] new_arr = new int[2 * arr.length];
            for (int i = 0; i < arr.length; i++) {
                new_arr[i] = arr[i];
            }
            arr = new_arr;
        }
        arr[idx++] = item;
    }

    public static void main(String[] args) throws Exception {
        Stack st = new DynamicStack();// push
        // DynamicStack st1 = new DynamicStack();
        st.push(10);
        st.push(20);
        st.push(30);
        st.push(40);
        st.push(50);
        st.push(60);
        st.push(70);
    }
```




```
public void Enqueue(int item) {
        // TODO Auto-generated method stub
        if (isFull()) {

        }
        super.Enqueue(item);
    }

    public static void main(String[] args) {
        DynamicQueue dq = new DynamicQueue();
        dq.Enqueue(10);
        dq.Enqueue(20);
        dq.Enqueue(30);
        dq.Enqueue(40);
        System.out.println(dq.Dequeue());
        System.out.println(dq.Dequeue());
        dq.Enqueue(50);
        dq.Enqueue(60);
        dq.Enqueue(70);
        dq.Enqueue(80);
        dq.Display();
    }
```