



# Comments in Dart

620710288 กนกนาฏ ชิขุนทด

## Comment คืออะไร?

**Comment** คือ ข้อความหรือคำอธิบายที่ถูกเขียน  
ใช้เพื่ออธิบายโค้ด เพื่อให้ผู้อื่นสามารถเข้าใจได้ง่าย<sup>ๆ</sup>  
โดยทั่วไป comment จะไม่ถูกนำเข้าสู่การทำงาน  
ของโปรแกรม

# Types of Comments

ชนิดของ Comments จะมีอยู่ 3 ประเภทหลัก ๆ คือ

- 1) Single-Line Comments: สำหรับการแสดงความคิดเห็นในโค้ดบรรทัดเดียว
- 2) Multi-Line Comments: สำหรับการแสดงความคิดเห็นในโค้ดหลายบรรทัด
- 3) Documentation Comments: สำหรับสร้างเอกสารหรือคำอธิบายที่อยู่ในโค้ดเอง

# Single-Line Comments

ใช้เพื่อแสดงความคิดเห็นในบรรทัดจนกว่าตัวแบ่งบรรทัดจะเกิดขึ้น

Syntax ที่ใช้คือ เครื่องหมายทับคู่ (//)

ตัวอย่าง:

```
int main()
{
    double area = 3.14 * 4 * 4;
    // It prints the area (จะพิมพ์พื้นที่)
    // of a circle of radius = 4 (ค่ารัศมีวงกลมมีค่า =4)
    print(area);

    return 0;
}
```

**Output:**

50.24



# Multi-Line Comments

ใช้เพื่อแสดงความคิดเห็นในส่วนของโค้ดทั้งหมด Syntax ที่ใช้คือ /\* เพื่อเริ่มต้น และปิดท้ายด้วย \*/ เพื่อสิ้นสุดความคิดเห็น ตัวอย่าง:

```
int main()
{
    var lst = [1, 2, 3];
    /*
    It prints
    the whole list
    at once (จะพิมพ์รายการทั้งหมดในครั้งเดียว)
    */
    print(lst);
    return 0; }
```

**Output:**

[1, 2, 3]

# Documentation Comments

มักจะรวมข้อมูลเกี่ยวกับ:

- การใช้งานของฟังก์ชันหรือคลาส
- พารามิเตอร์และค่าที่รับเข้า
- ค่าที่ส่งออก
- ตัวอย่างการใช้งาน
- คำแนะนำเกี่ยวกับการใช้งาน

Syntax ที่ใช้คือ `///` หรือ `**/...*/` แต่ `///` จะทำให้เข้าใจง่ายกว่า



## ตัวอย่าง:

```
bool checkEven(n){  
    /// Returns true (คืนค่า true)  
    /// if n is even (ถ้า n เป็นเลขคู่)  
    if(n%2==0)  
        return true;  
    /// Returns false if n is odd (คืนค่า false ถ้า n เป็นเลขคี่)  
    else  
        return false; }  
  
int main()  
{  
    int n = 43;  
    print(checkEven(n));  
    return 0;  
}
```

Output:

[false]



# ตัวอย่าง Comments ในภาษาอื่น ๆ

## Single-Line Comments

- C

```
#include <stdio.h>
Int main()
{
    // print Hello world to the screen
    print("Hello world");
    Return 0;
}
```

Output:

Hello world





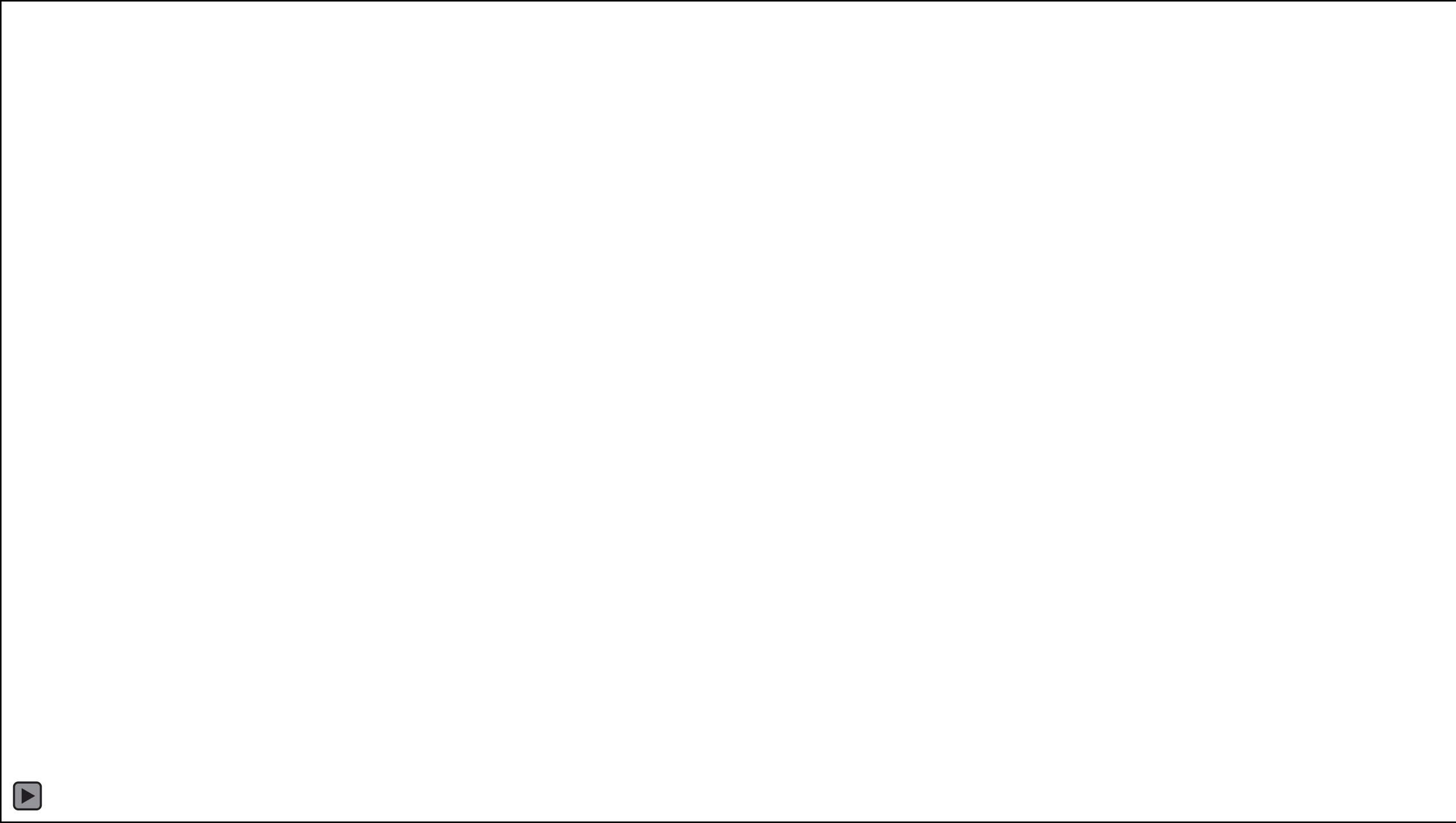
## Single-Line Comments

- Java

```
public class JavaCommentsExample {  
  
    public static void main(String[] args) {  
        System.out.println("This line will run.");  
        // System.out.println("This line will not run.");  
    }  
}
```

Output:

This line will run.



# ตัวอย่าง Comments ในภาษาอื่น ๆ

## Multi-Line Comments

- C

```
/* This program takes age input from the user  
It stores it in the age variable  
And, Print the value using printf()*/
```

```
#include <stdio.h>  
  
int main()  
{  
    int age;  
    printf("Enter the age: ");  
    scanf("%d", &age);  
    printf("Age= %d", age);  
    return 0;  
}
```

Output:

```
Enter the age: 24  
Age= 24
```



# Multi-Line Comments

- Java

```
public class JavaCommentsExample {  
    /* This block comment spans only one line. */  
    public static void main(String[] args) {  
        System.out.println("This line will run.");  
  
        /* This block comment spans multiple lines.  
         * System.out.println("This line will not run.");  
         * System.out.println("This line will not run.");  
         * None of the code in this Java block comment will run.  
    }  
}
```



## Multi-Line Comments

- Python ใช้ได้ทั้งสองแบบ คือ #, "", หรือ """

```
# This is a long comment  
# and it extends  
# to multiple lines
```

```
"" This is also a  
perfect example of  
multi-line comments ""
```



# ตัวอย่าง Comments ในภาษาอื่น ๆ

## Documentation Comments

- Java

```
/** This JavaDoc comment should describe the class. */
```

```
public class JavaCommentsExample {
```

```
/** This JavaDoc comment should describe the method. */
```

```
    public static void main(String[] args) {
```

```
        System.out.println("This line will run."); // Java comment
```

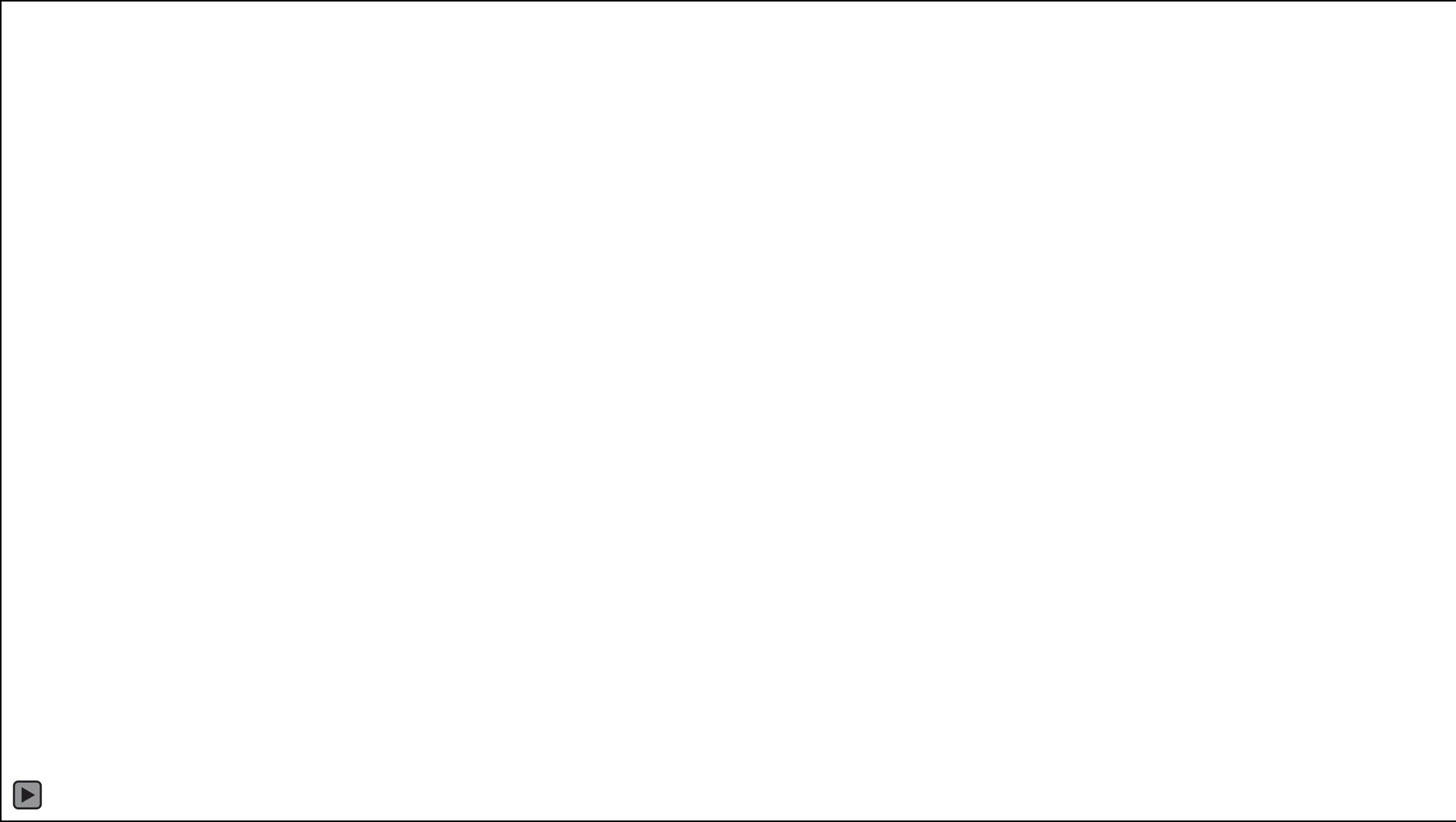
```
        /* System.out.println("This is in a Java block comment."); */
```

```
}
```

```
}
```

Output:

This line will run.



# Thank You