There is **no real transaction happening**.

Instead:

- `minBuyPrice` =
  👉 *"If I had bought on the best possible day so far, what price would that be?"*

So yes — **you haven't committed to buying yet**.
You're just *remembering the cheapest opportunity* in the past.

Think of it like:

> "I'm window-shopping, keeping track of the cheapest price I've ever seen."

Only when we calculate profit, we *pretend*:

> "What if I had bought at that cheapest price and sold today?"

---

## Let's do a precise dry run, line by line

### Input

```ini
prices = [7, 1, 5, 3, 6, 4]
```

---

## Initialization (before loop)

```cpp
minBuyPrice = 7
maxProfit = 0
```

📌 Meaning:

- Cheapest price **seen so far** = 7
- We haven't made any profit yet
- No buy, no sell — just observation

---

## Day 0 (price = 7)

Already used for initialization
Nothing else happens

---

## Day 1 (price = 1)

```cpp
profitIfSoldToday = 1 - 7 = -6
maxProfit = max(0, -6) = 0
minBuyPrice = min(7, 1) = 1
```

🧠 Interpretation:

- Selling today makes no sense (loss)
- **We found a cheaper price**
- So we update:

> “If I ever buy, THIS is a better day to buy”

❗ Still no buying.
Just remembering: **best buying option so far = 1**

---

## Day 2 (price = 5)

```cpp
profitIfSoldToday = 5 - 1 = 4
maxProfit = max(0, 4) = 4
minBuyPrice = min(1, 5) = 1
```

🧠 Interpretation:

- If I **had bought at 1** and sell today → profit = 4
- That’s the best profit so far
- Cheapest price remains 1

📌 Logical transaction (imaginary):

> Buy at 1 → Sell at 5 → Profit = 4

Still **not locked in**, just recorded.

---

## Day 3 (price = 3)

```cpp
profitIfSoldToday = 3 - 1 = 2
maxProfit = max(4, 2) = 4
minBuyPrice = min(1, 3) = 1
```

🧠 Interpretation:

- Selling today is worse than selling at 5
- Ignore

- Best buy price is still 1

---

## Day 4 (price = 6)

```cpp
profitIfSoldToday = 6 - 1 = 5
maxProfit = max(4, 5) = 5
minBuyPrice = min(1, 6) = 1
```

🧠 Interpretation:

- This is the **best selling day so far**
- Imaginary transaction:

    Buy at 1 → Sell at 6 → Profit = 5

---

## Day 5 (price = 4)

```cpp
profitIfSoldToday = 4 - 1 = 3
maxProfit = max(5, 3) = 5
minBuyPrice = min(1, 4) = 1
```

🧠 Interpretation:

- Worse than before
- Ignore

---

## Final Answer

```cpp
return maxProfit = 5;
```

---

## Now let's address your CORE confusion 🔥

> "It's like I still didn't buy yet right? Because minBuyPrice is updating"

### ✅ YES. EXACTLY.

You **never actually buy**.

Instead:

- **`minBuyPrice`** keeps improving until the **best opportunity** is found