Writing programming interview questions hasn't made me rich yet ... so I might give up and start trading Apple stocks all day instead.

First, I wanna know how much money I *could have* made yesterday if I'd been trading Apple stocks all day.

So I grabbed Apple's stock prices from yesterday and put them in a list called `stock_prices`, where:

- The **indices** are the time (in minutes) past trade opening time, which was 9:30am local time.
- The **values** are the price (in US dollars) of one share of Apple stock at that time.

So if the stock cost $500 at 10:30am, that means `stock_prices[60] = 500`.

Write an efficient function that takes `stock_prices` and returns **the best profit I could have made from one purchase and one sale of one share of Apple stock yesterday.**

For example:

```
                                                        Python 3.6 ▼
stock_prices = [10, 7, 5, 8, 11, 9]


get_max_profit(stock_prices)
# Returns 6 (buying for $5 and selling for $11)
```

No "shorting"—you need to buy before you can sell. Also, you can't buy *and* sell in the same time step—at least 1 minute has to pass.

Want more coding interview help?